

# Ad Inserter

## Hotels

[www.example.com](http://www.example.com)

Special rates until the end of the month. No booking fees, book your room now!



## Dublin Hotels

[www.example.com](http://www.example.com)

Browse hundreds of hotels in Dublin, sort by price, location and user reviews.



## Hotels in Ireland

[www.example.com](http://www.example.com)

Compare prices of 1000s of hotels all over ireland!



AdChoices

### Ad Inserter Stats

Version:	2.3.15
Last updated:	24 hours ago
Active installations:	100,000+
Requires WordPress:	4.0
Tested up to:	4.9.7
Requires PHP Version:	5.6

### Ratings



5 stars		646
4 stars		11
3 stars		1
2 stars		0
1 star		4

# User Manual

# Table of Contents

Introduction.....	6
Features.....	7
Installation.....	9
Uninstall.....	9
Quick Start.....	10
Settings.....	13
Block name.....	14
Block code.....	15
Quickly disable insertion.....	15
Simple editor for mobile devices.....	15
PHP code processing.....	15
Code editing.....	16
Code generator.....	16
Banners.....	16
Placeholders.....	18
AdSense.....	19
Rotation Editor.....	20
Visual HTML editor.....	22
Clipboard.....	22
Import/Export (Pro only).....	24
Insertion Options.....	25
Automatic Insertion.....	25
Disable insertion.....	27
Custom Hooks.....	27
Insertion Before or After HTML element.....	28
Alignment and Style.....	30
Wrapping code.....	30
Default Alignment.....	31
Left Alignment.....	31
Right Alignment.....	32
Center Alignment.....	32
Float Left.....	32
Float Right.....	33
No Wrapping.....	33
Custom CSS.....	33
Sticky Ads (Pro only).....	35
Horizontal position.....	36
Vertical position.....	39
Animation.....	42
Code Preview.....	45
Page types.....	47
Post/Page Exceptions.....	49
Paragraphs.....	51
Counting.....	51
Clearance.....	52

Ad rotation.....	53
Ad counting.....	55
Buttons.....	56
Button Misc.....	56
Insertion.....	56
Scheduling.....	57
Display.....	57
Word Count.....	57
Filter.....	57
Ad Blocking (Pro only).....	58
General.....	59
Button Lists - Black and White Lists.....	59
Categories.....	60
Tags.....	60
Taxonomies (also Users and Post Types).....	60
Post IDs.....	61
Urls.....	61
Url parameters and Cookies.....	61
How to show ads based on visitor's consent?.....	61
Referers.....	62
IP addresses (Pro only).....	62
Countries (Pro only).....	62
Button Devices – Device detection.....	63
Special Insertions.....	65
Button Manual - Manual Insertion.....	66
Disable insertion.....	66
Widgets.....	67
Shortcodes.....	67
PHP function call.....	67
Special Shortcodes.....	68
AdSense Ads.....	69
New to AdSense?.....	69
Ad unit.....	70
Text & display ads.....	70
Responsive ads.....	71
Ad code.....	71
AdSense code generator.....	73
AdSense Integration.....	75
In-feed ads.....	80
In-article ads.....	82
Auto ads.....	83
Connect your site.....	84
Ads not displayed – troubleshooting.....	86
Ad Impression and Click Tracking (Pro only).....	90
Impressions.....	90
Clicks.....	90
Standard Click Detection.....	90

Advanced Click Detection.....	91
Pageview Tracking.....	92
Ad Tracking.....	93
Internal Tracking Statistics.....	93
A/B testing.....	95
Ad Blocking Statistics.....	97
Ad Blocking Detection.....	98
Ad Blocking Statistics (Pro only).....	98
Ad Blocking Detection Settings.....	98
Popup Message.....	100
Redirection.....	102
Ad Replacement (Pro Only).....	103
Do nothing.....	103
Replace.....	103
Show.....	103
Hide.....	103
Content protection (Pro only).....	104
Hide Content.....	104
Delete Content.....	104
Change Style.....	104
Replace Text.....	105
Content Selectors.....	105
Troubleshooting Content Protection.....	105
Debugging Ad Blocking Detection.....	105
Ad Blocking Status.....	106
Simulate Ad Blocking.....	106
Practical example of content protection.....	107
Ad Inserter General Plugin Settings.....	112
General.....	112
Viewports.....	115
Header and Footer code.....	115
Page-specific Header and Footer code.....	116
Raw HTTP Header.....	116
Geolocation (Pro only).....	118
GEO targeting Groups.....	119
Exceptions (Pro only).....	119
Multisite (Pro only).....	120
Debugging.....	120
Debugging.....	121
Debugging Tools widget .....	121
Troubleshooting.....	127
Plugin upload fails.....	127
Ads not displayed.....	127
Page blocked.....	127
Incompatible file loaded.....	128
Buttons missing or page elements distorted.....	129
document.write errors.....	129

Tracking, ad blocking detection or other client-side code is not working.....	129
Ad Blockers.....	129
Are you sure you want to do this?.....	130
Heavy website load.....	130
License Expired.....	131
License Overused.....	131
Caching.....	132
W3 Total Cache.....	133
Ad rotation.....	134
Device detection.....	135
IP address and country detection (Pro only).....	136
Common Settings.....	137
Ad before the first paragraph on all posts.....	137
Ad before the last paragraph on all posts.....	137
Contextual Amazon Native Shopping Ads.....	138
Ad wrapped with text of the third paragraph on all posts.....	139
Ad above the first and third post excerpts on the homepage.....	139
Ad between the second, third and fourth comment on posts.....	139
Ad before the second paragraph on all posts, homepage ads in the first three post.....	139
Centered ad in the middle of post paragraphs.....	140
Ad above post excerpts on the Insurance category page.....	140
Ad above the post excerpts on the Cars tag page.....	141
Settings for ads on AMP pages.....	142
Generating AdSense code for AMP pages.....	143
Support.....	145

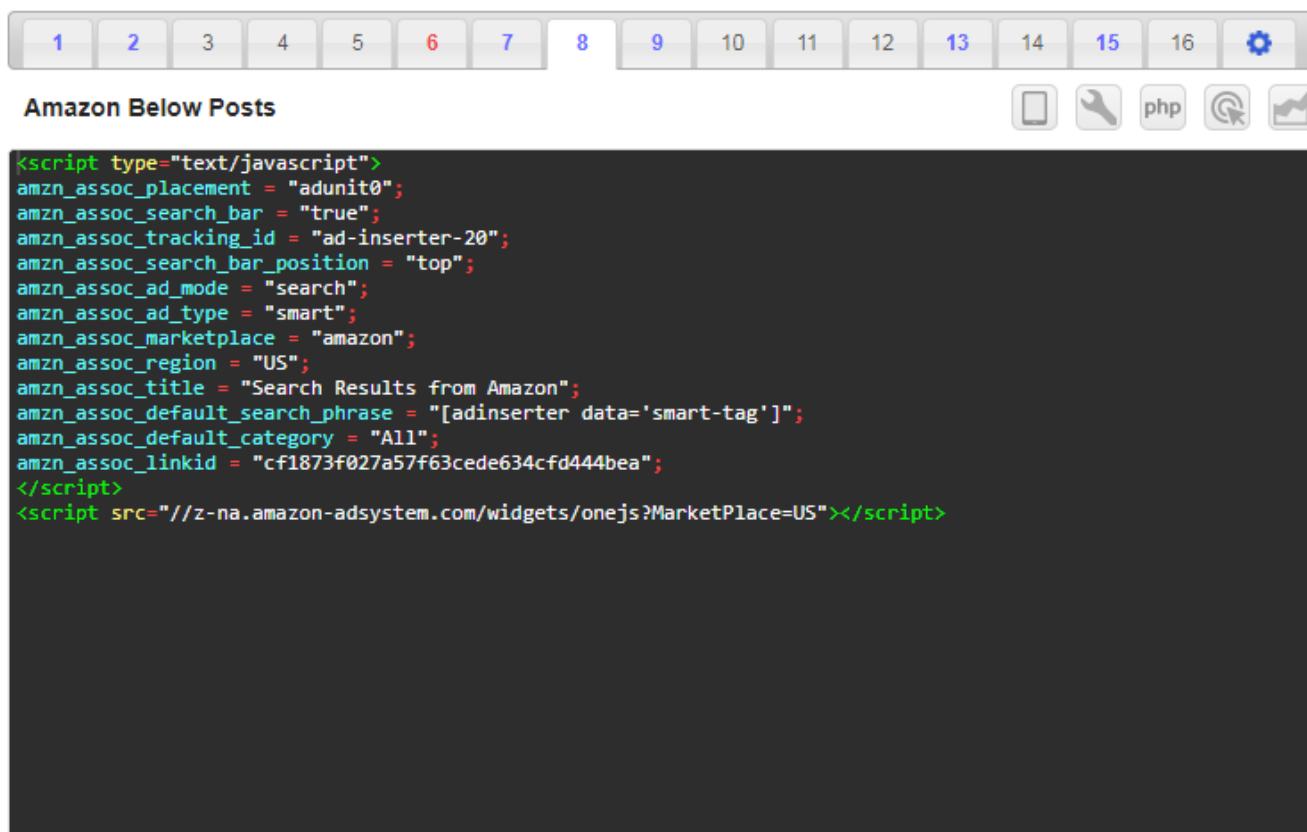
## Introduction

---

This document describes free **Ad Inserter** and **Ad Inserter Pro** which has additional features. Unless marked with **Pro only** everything described here is valid also for free Ad Inserter.

Ad Inserter is a simple yet powerful Wordpress ad management plugin to **insert any ad or code into WordPress**. Perfect for all kinds of ads including AdSense, Amazon and Media net. It can be also used to insert opt-in forms, videos, texts, etc. Simply enter any ad or HTML/Javascript/PHP code and select where and how you want to display it. All the settings are available on a single page so management is very simple.

Free Ad Inserter contains most of the functionality of the Pro version. The main difference is the number of code blocks supported, sticky ads with optional animation and close button, country-level GEO targeting, ad impression and click tracking, ad replacement or content protection when ad blocking is detected and support via email - see below for details. Before you purchase license for Ad Inserter Pro please check whether free **Ad Inserter** can be installed on your website, it works as expected, you are not experiencing issues and it fits your needs. If free Ad Inserter does not work, Ad Inserter Pro will also not work. If you are having issues with the free Ad Inserter you can use [Ad Inserter WP support forum](#) and ask for help or advice there.



The screenshot shows the Ad Inserter plugin interface. At the top, there is a navigation bar with numbered tabs from 1 to 16, and a gear icon for settings. Below the navigation bar, the title "Amazon Below Posts" is displayed. To the right of the title are several icons: a mobile device, a wrench, a "php" icon, a circular arrow, and a line graph. The main content area is a dark-themed code editor containing the following JavaScript code:

```
<script type="text/javascript">
amzn_assoc_placement = "adunit0";
amzn_assoc_search_bar = "true";
amzn_assoc_tracking_id = "ad-inserter-20";
amzn_assoc_search_bar_position = "top";
amzn_assoc_ad_mode = "search";
amzn_assoc_ad_type = "smart";
amzn_assoc_marketplace = "amazon";
amzn_assoc_region = "US";
amzn_assoc_title = "Search Results from Amazon";
amzn_assoc_default_search_phrase = "[adinsserter data='smart-tag']";
amzn_assoc_default_category = "All";
amzn_assoc_linkid = "cf1873f027a57f63cede634cfd444bea";
</script>
<script src="//z-na.amazon-adsystem.com/widgets/onejs?MarketPlace=US"></script>
```

## Features

Ad Inserter supports up to 16 code blocks. Code block is any code (for example AdSense ad) that has to be inserted (displayed) at some position. Each code block can be configured to insert code at almost any position on any WordPress page. Main features of Ad Inserter include:

- 16 code (ad) blocks (64 code blocks in Pro)
- AdSense integration
- Syntax highlighting editor
- Code preview with visual CSS editor
- Simple user interface - all settings on a single page
- Automatic Insertion: before or after post / content / paragraph / excerpt
- Automatic Insertion between posts
- Automatic insertion before, between and after comments
- Automatic insertion below `<body>` or above `</body>` tags
- Automatic insertion at custom hook positions
- Automatic insertion before or after any HTML element on page
- Insertion exceptions for posts and pages
- Manual Insertion: widgets, shortcodes, PHP function call
- Sticky ad positions with optional close button (Pro only)
- Sticky sidebar ads (stick to the screen or to the content, Pro only)
- Sticky ad trigger (page scroll in % or px, HTML element visible, Pro only)
- Sticky ad animations (fade, slide, turn, flip, zoom, Pro only)
- Sticky (fixed) widgets (sidebar does not move when the page scrolls)
- Custom block alignment and style
- Clearance options to avoid insertion near images or headers
- Insertion of custom fields
- PHP code processing
- Banner and placeholder code generator
- Server-side and client-side device detection (works with caching, 3 custom viewports, 6 in Pro)
- Black/White-list categories, tags, taxonomies, users, post IDs, urls, url parameters, cookies, referers
- Black/White-list IP addresses and countries (Pro only)
- Ad rotation (works also with caching)
- Timed ad rotation – rotate different ads in a single ad block without page reload
- Ad impression and click tracking (Pro only)
- A/B testing (Pro only)
- Support for lazy loading (load ads only when they become visible, Pro only)
- Support for ads on AMP pages
- Support for header and footer code
- Ad blocking detection - popup message, page redirection
- Replace ads and/or protect content when ad blocking is detected (Pro only)
- Ad blocking statistics (Pro only)
- Simple troubleshooting with many debugging functions
- Function to visualize inserted code blocks
- Function to visualize available positions for automatic insertion
- Function to visualize HTML tags
- Export and import of settings (Pro only)

- Multisite options to limit settings on the sites (Pro only)
- Country-level GEO targeting (works also with caching, Pro only)
- Scheduling with fallback option (Pro only)
- Clipboard support to easily copy code blocks or settings
- Support via email (Pro only)



## Installation

Free **Ad Inserter** can be installed from Wordpress (Plugins / Add New / search for **Ad Inserter**).

When you decide to upgrade to **Ad Inserter Pro** and purchase the license you will receive an email containing your download link and license key for the plugin. The link in the email is used only for initial download and installation of the plugin. All future updates during the free update period don't need manual downloading and can be done inside Wordpress like for other plugins. Please save this email for future reference. Of course, if you have lost your download link or license key you can [recover](#) it by entering email address that was used for payment.

Installing Ad Inserter Pro is simple and preserves settings from Ad Inserter. **If you are using free Ad Inserter simply uninstall it.** The Pro version will automatically import existing settings from the free version. After you receive the email with download link for the Ad Inserter Pro plugin, download it, go to WordPress Plugins, Add New, Upload Plugin, Choose file, click on Install Now, activate it and then click "Enter License Key" and enter license key you got in the email. If you need to later edit license key go to Ad Inserter plugin settings (tab ⚙️). If you have trouble uploading the plugin check Plugin upload fails section for explanation and solutions.

**WARNING You will not be able to activate Ad Inserter Pro unless you uninstall free Ad Inserter first.**

**WARNING: If you are using ad blocker make sure it is disabled when using Ad Inserter. If ad blocker is active you may experience issues and some functions may not work as described here.**

Some security plugins like WP Security or Wordfence may also detect Ad Inserter page as problematic and prevent it from loading. Please be assured that this is false positive. **Javascript errors may also prevent the settings page from properly loading.** Check Javascript console for errors or messages.

If you are using caching the inserted code may not appear immediately on the page. Make sure you have disabled caching when you are testing or debugging. Some caching plugins like WP Super Cache have an option to disable caching for known users.

If you are using caching some settings may not work as expected. For example, ad rotation, referer check, user check, country check (using **server-side detection** or **server-side dynamic blocks**) normally work only when the page is generated and Ad Inserter is called. In such cases please make sure you have disabled caching when you are using such settings. Some functions can work with caching but need some additional or different settings. Please check Caching for more.

Some code for ads may not display anything, either because of errors in the ad code or because of ad network issues. Before you report problem please check source code of the page and make sure the code is not inserted where it should be. The code may be inserted properly but you won't see anything. Try to add some text after the ad code to check if it appears at the expected ad position. Check also Debugging and Ads not displayed sections for more information.

## Uninstall

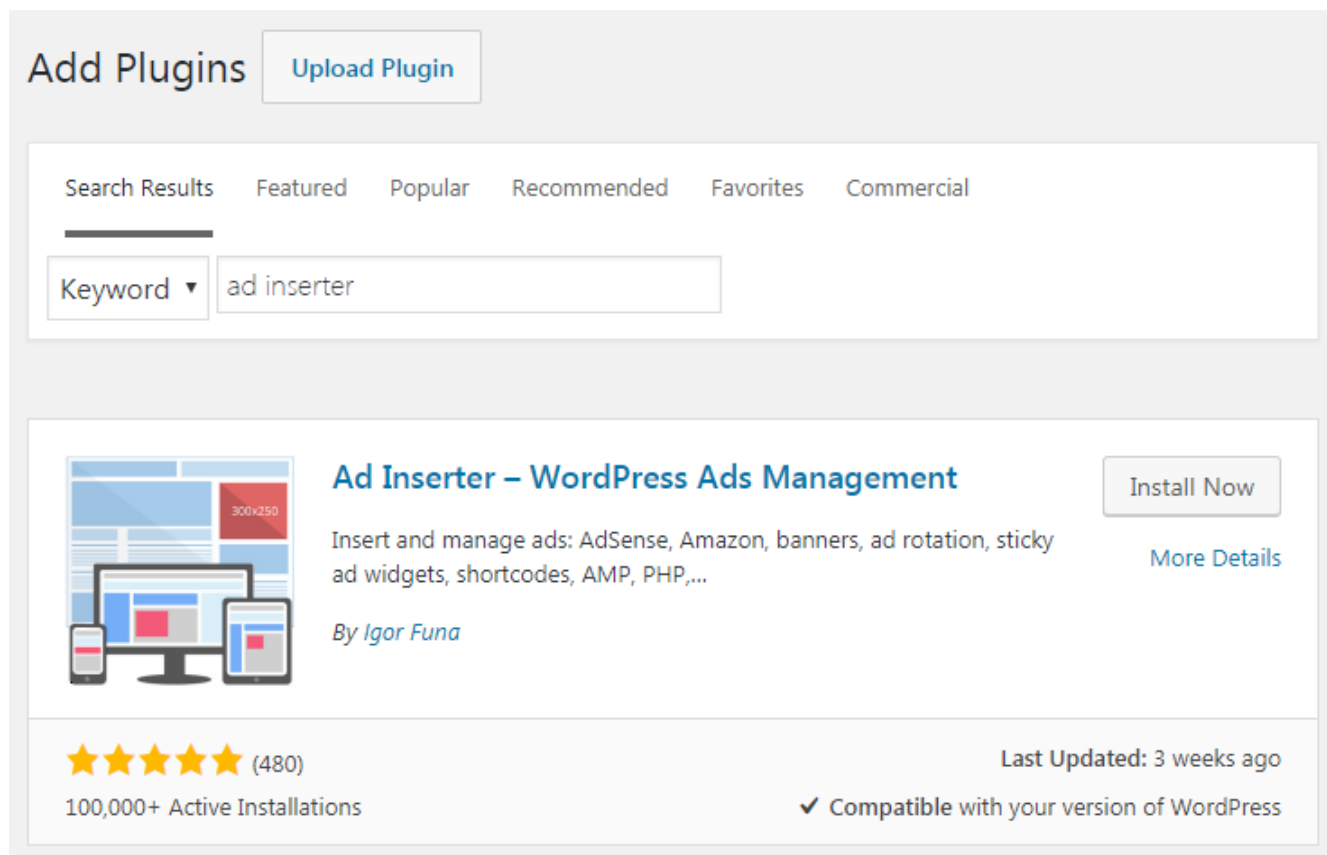
If you deactivate and delete Ad Inserter the settings will remain in the database. To completely remove the plugin and the settings do the following:

- Go to Ad Inserter general settings (tab ⚙️) and click on **Reset All Settings**
- Go to Ad Inserter general settings (tab ⚙️) and click on **Clear All Statistics Data** (Pro only)
- Deactivate Ad Inserter
- Delete Ad Inserter plugin

## Quick Start

---

To install Ad Inserter plugin log in to your WordPress dashboard, click on **Plugins** in the WordPress menu, click on **Add New** on top of the plugin page, search for **Ad Inserter**, select **Ad Inserter** from the results, click **Install** and then **Activate** the plugin.



The screenshot shows the WordPress 'Add Plugins' interface. At the top, there's a search bar with 'ad inserter' entered. Below the search bar, the 'Ad Inserter – WordPress Ads Management' plugin is featured. It includes a thumbnail image of a computer monitor and a smartphone displaying ads, the title 'Ad Inserter – WordPress Ads Management', a description 'Insert and manage ads: AdSense, Amazon, banners, ad rotation, sticky ad widgets, shortcodes, AMP, PHP,...', the author 'By Igor Funa', a '5 stars (480)' rating, '100,000+ Active Installations', and a 'Last Updated: 3 weeks ago' note. There are 'Install Now' and 'More Details' buttons.

Some typical settings are described in the section Common Settings. Please make sure you have also read Troubleshooting section.

Few very important things you need to know in order to insert code and display some ad:

- Code block (there are 16 of them, 64 in Pro) is ad code with settings which define where the ad will be inserted (displayed).
- Each code block can be renamed. Click on the name to edit it.
- Enter or paste the code into the code window or use code tools to generate or edit banner code. You can also use visual HTML editor to create or edit HTML code for your ad.

**WARNING:** If you are using **ad blocker** make sure it is disabled when using Ad Inserter. If ad blocker is active you may experience issues and some functions may not work as described here.



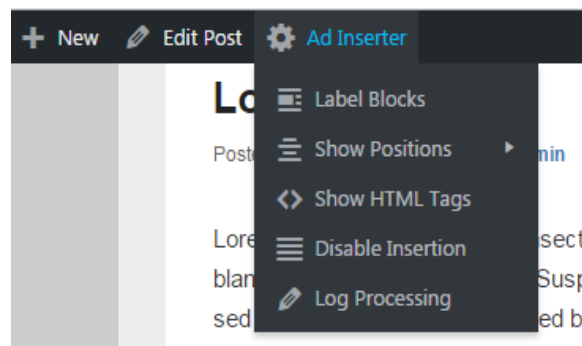
- **Enable and use at least one insertion option (Automatic Insertion or manual insertion Widget, Shortcode or PHP function call)**

Automatic Insertion:  Alignment and Style:

- **Enable insertion on at least one Wordpress page type** (Posts, Static Pages, Blog pages: Homepage, Category pages, Search Pages, Archive pages)
- For **Posts** and **Static Pages** leave default **Blank** selection value after the checkbox unless you really know what are you doing (using individual exceptions)

<input checked="" type="checkbox"/>	<input type="text" value=""/>	Posts	<input type="checkbox"/>	Homepage	<input type="checkbox"/>	Category pages
<input type="checkbox"/>	<input type="text" value=""/>	Static pages	<input type="checkbox"/>	Search pages	<input type="checkbox"/>	Tag / Archive pages

- If you don't see inserted code block turn on debugging functions: **Label blocks, Show positions** for automatic insertion (Ad Inserter menu item in the WordPress toolbar on the top of every post/page), Check Debugging and Ads not displayed sections for more.



- If you are using AdSense you may get blank (empty) ad blocks. This might be because there is some error in the code (wrong IDs), your AdSense account is not fully approved yet, your website was not accepted or your AdSense account is banned. Try [Media net ads](#) – they seem to be a good alternative for contextual ads.




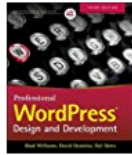
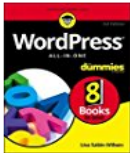



## Endorsed by Google

Google lists Ad Inserter plugin as possible [solution to place AdSense ad code into your WordPress site](#). If you're new to AdSense Ads, one of your first tasks is to connect your site to AdSense. This requires you to copy the code on your AdSense homepage and paste it into the HTML of your page, between the and tags. Advanced users can enable AdSense Integration: get AdSense ad codes from the plugin settings page.

## Endorsed by Amazon

Amazon suggests to use Ad Inserter to add Native Shopping Ads to Wordpress posts. Native Shopping Ads provide highly relevant and dynamic product recommendations in a stylishly designed and responsive ad unit that can be placed at the end of your content or within your content to create a more compelling visitor experience and shopping opportunity. Check [Wordpress Integration Guide for Native Shopping Ads](#). In the section Common Settings you can also find an example of contextual Amazon ads that show items related to the post content (using post tags).

All wordpress Go

 <p>Step-By-Step WordPress for Beginners: How to Build ...</p> <p>\$5.60 <del>\$7.00</del> ✓prime ★★★★☆ (67)</p>	 <p>WordPress for Beginners 2017: A Visual Step-by-St...</p> <p>\$3.79 ★★★★☆ (93)</p>	 <p>WordPress: The Missing Manual (Missing Manuals)</p> <p>\$20.22 <del>\$20.00</del> ✓prime ★★★★☆ (123)</p>	 <p>Professional WordPress: Design and Development</p> <p>\$33.55 <del>\$60.00</del> ✓prime ★★★★☆ (144)</p>
 <p>WordPress All-in-One For Dummies (For Dummies (...)</p> <p>\$21.93 <del>\$24.00</del> ✓prime ★★★★☆ (12)</p>	 <p>WordPress: 665 Free WordPress Plugins for Creat...</p> <p>\$6.99 ✓prime ★★★★☆ (46)</p>	 <p>WordPress To Go: How To Build A WordPress Website...</p> <p>\$9.99 ✓prime ★★★★☆ (430)</p>	 <p>Teach Yourself VISUALLY WordPress</p> <p>\$19.28 <del>\$20.00</del> ✓prime ★★★★☆ (44)</p>

## Settings

Ad Inserter can be configured to insert any code anywhere on the page. **Each ad code with its settings is called a code block.** Free Ad Inserter supports 16 code blocks, Ad Inserter Pro supports up to 96 code blocks (depending on the license type). The settings page is divided into tabs – 16 code blocks and general plugin settings. Black number means inactive code block (code is not inserted anywhere), red number means block is using automatic insertion, blue number means block is using manual insertion while violet number means block is using automatic and manual insertion.

The screenshot shows the settings interface for the 'Amazon Below Posts' code block. At the top, there is a navigation bar with 16 numbered tabs. Tab 13 is highlighted in red, indicating automatic insertion. Below the tabs, the title 'Amazon Below Posts' is displayed. To the right of the title are icons for mobile, settings, PHP, refresh, and a chart. The main area contains a dark-themed code editor with the following JavaScript code:

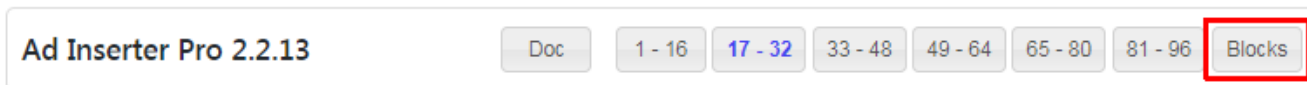
```
<script type="text/javascript">
amzn_assoc_placement = "adunit0";
amzn_assoc_search_bar = "true";
amzn_assoc_tracking_id = "ad-inserter-20";
amzn_assoc_search_bar_position = "top";
amzn_assoc_ad_mode = "search";
amzn_assoc_ad_type = "smart";
amzn_assoc_marketplace = "amazon";
amzn_assoc_region = "US";
amzn_assoc_title = "Search Results from Amazon";
amzn_assoc_default_search_phrase = "[adinsserter data='smart-tag']";
amzn_assoc_default_category = "All";
amzn_assoc_linkid = "cf1873f027a57f63cede634cfd444bea";
</script>
<script src="//z-na.amazon-adsystem.com/widgets/onejs?MarketPlace=US"></script>
```

Below the code editor are several control elements:

- A row of tabs: Lists, Manual, Devices, Misc, Preview. A 'Save Settings 1 - 16' button is on the right.
- 'Automatic Insertion: After Content' (dropdown menu).
- 'Alignment and Style: Default' (dropdown menu) with a 'Show' button.
- Checkboxes for page types: Posts (checked), Static pages, Homepage, Category pages, Search pages, Tag / Archive pages.

Each code block has a name, code and settings which are divided into sections. Some settings depend also on automatic insertion used. When you are finished with settings you need to save them by clicking on the button **Save Settings**.

To show a list of all blocks click on the **Blocks** button in the header row.



This list shows all active code blocks. To show all blocks click on the top right button. To filter blocks enter keywords in the search field. You can also use **php**, **shortcode** and **widget** keywords to filter blocks that have enabled respective functions.

Block	Name	Automatic Insertion	Alignment	P	S	W	Widget Positions
<input type="checkbox"/>	▶ 3 Amazon Sidebar	Disabled	Default	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	sidebar-a
<input type="checkbox"/>	▶ 6 16-Channel USB Logic Analyzer	Disabled	Default	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	sidebar-a
<input type="checkbox"/>	▶ 8 AdSense Bottom	After Post	Default	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	
<input type="checkbox"/>	▶ 9 Amazon / AdSense Sidebar	Disabled	Default	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	sidebar-a

Here you can also easily rearrange block order. Click on the **Rearrange block order** button to enable moving blocks.

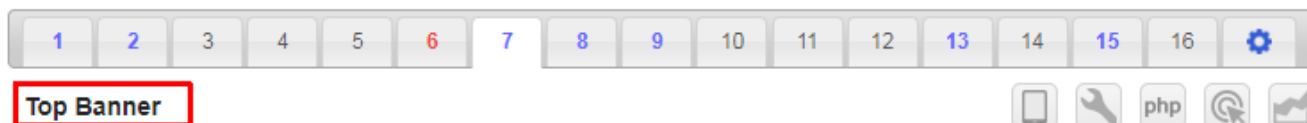


Now you can drag and move blocks to create new block order. The position where the block would be moved is highlighted. When you are happy with the order click on the **Save Changes** button. Click on the **Rearrange block order** button to cancel changes. When you save new block order this changes also Ad Inserter widget blocks in widget positions and statistics data accordingly.

Block	Name	Automatic Insertion	Alignment	P	S	W	Widget Positions
<input type="checkbox"/>	▶ 6 16-Channel USB Logic Analyzer	Disabled	Default	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	sidebar-a
<input type="checkbox"/>	▶ 3 Amazon Sidebar	Disabled	Default	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	sidebar-a
<input type="checkbox"/>	▶ 8 AdSense Bottom	After Post	Default	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	
<input type="checkbox"/>	▶ 9 Amazon / AdSense Sidebar	Disabled	Default	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	sidebar-a

## Block name

To rename code block click on the name for editing.



## Block code

Ad Inserter provides many simple functions to generate, edit or copy codes for ads. You can simply paste the code you have generated elsewhere, create code for banners or text ads or copy and edit code form other code blocks.

```
<a href="https://wordpress.org/plugins/ad-inserter/" target="_blank"></a>
```

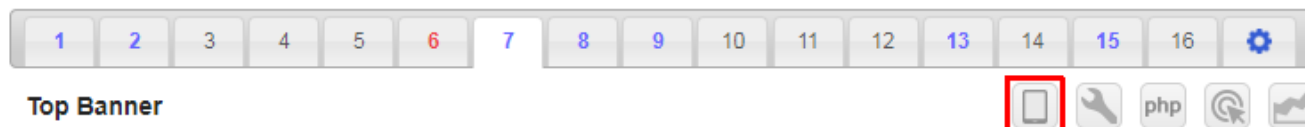
## Quickly disable insertion

To insert code block you need to use automatic or manual insertion. Once you configure settings including insertion the code will be inserted. If for some reason you need to temporarily disable (pause) insertion you can disable automatic insertion or uncheck manual insertion. To achieve the same effect without changing the settings you can simply disable (pause) insertion of each code block. To disable insertion of code block click on the **pause button** (it will be red when paused) and save settings.



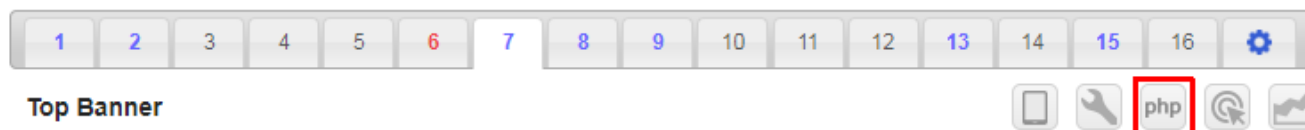
## Simple editor for mobile devices

On mobile devices some functions like text selection, copy and paste may not work with default syntax highlighting editor. In such case enable simple editor to switch to plain text editor.



## PHP code processing

If you have PHP code (surrounded with PHP tags `<?php ... ?>`) in code block, you need to enable PHP code processing.



## Code editing

You can simply edit the code in the code window but there are also additional tools to help you creating, editing and copying the code. To show the toolbar click on the **Code tools** button.



## Code generator

Code generator currently supports code for banners and AdSense. To open code generator click on the code generator button.



## Banners

Ad Inserter has a simple code generator for banners and placeholders. You can select banner image (or placeholder), optionally define link (web page address that will open when the banner will be clicked) and select whether to open link in a new tab.

A screenshot of the Ad Inserter banner code generator form. At the top, there are two tabs: 'Banner' and 'AdSense'. To the right of the tabs are two vertical arrows (up and down). Below the tabs are two input fields: 'Image' and 'Link'. Below the 'Link' field is a checkbox labeled 'Open link in a new tab'. At the bottom right, there are two buttons: 'Select Placeholder' and 'Select Image'.

**WARNING:** If you are using **ad blocker** make sure it is disabled when using Ad Inserter. If ad blocker is active you may experience issues and some functions may not work as described here.



If you already have a banner HTML code in the code window you can import it by clicking on the **import** button. A thumbnail will be displayed next to the banner fields.

## Placeholders

If you don't have banner image ready yet you can use placeholder images to create ad positions. Ad Inserter has built-in placeholder generator/editor to easily select dummy image of any size or color. Click on the **Select Placeholder** button to open placeholder editor.

### Placeholder 300x250

Size	<input type="text" value="Custom"/>	Width	<input type="text" value="300"/>	Height	<input type="text" value="250"/>	Background color	<input type="text" value="#cccccd"/>	<input type="button" value="Use"/>
		Text	<input type="text" value="300 x 250"/>			Text color	<input type="text" value="#969697"/>	<input type="button" value="Edit"/>
								<input type="button" value="Cancel"/>

Here you can create a universal placeholder image that can be used in place of ads when they are not available yet. Placeholder images created here will behave as any other image. You can also save them to local computer or server.



Choose between common ad sizes 300x250, 336x280, 728x90, 468x60, 250x250, 300x600 or define custom size. Default placeholders are gray with size as placeholder text but you can use any color or text you want. Click on **Edit** button to edit placeholder size, color or text. You can also create blank solid color rectangles by clearing placeholder text.

**Please note:** if you have active rotation editor the code window shows only the code for the currently selected option. Therefore, code generator will in such case import or generate code for the currently selected option.

Code generator for banners and AdSense generates the code only when you click on the button Generate code. It is a tool that can help you to create code for AdSense or banners with links. So if you are

using rotation editor and switch between options you need to (optionally) import and generate code for each rotation option.

Ad Inserter has a simple code generator for banners and placeholders. You can select banner image (or placeholder), optionally define link (web page address that will open when the banner will be clicked) and select whether to open link in a new tab.

Choose between predefined ad sizes or define custom size. Default placeholders are gray with size as placeholder text but you can use any color or text you want. Click on **Edit** button to edit placeholder size, color or text. You can also create blank solid color rectangles by clearing placeholder text.

Once you are happy with the placeholder click on the **Use** button to generate placeholder image url.

## AdSense

Ad Inserter has a simple code generator for AdSense ads. Here you can define data needed for the AdSense code. In most cases it will be much easier to simply paste AdSense code into the code window and then import the data from it. You can also get AdSense ad codes directly from the ad Inserter settings page. Check AdSense Ads section for details.

This tool is mainly intended to change the type of AdSense ads. But please note: not all conversions are possible: you can convert standard ads to link ads or vice versa, fixed width ads to responsive ads and vice versa and you can enable or disable ads on AMP pages. Once you have the **Publisher ID** and **Ad Slot ID** entered (or imported) you can change ad type, responsiveness and ad type for AMP pages.

Banner AdSense ↑ ↓

Publisher ID

Ad Slot ID

Ad Type   Responsive

AMP Ad

```
<script async src="//pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
<ins class="adsbygoogle"
  style="display:block"
  data-ad-client="ca-pub-6360287238763624"
  data-ad-slot="3033292924"
  data-ad-format="auto"></ins>
<script>
(adsbygoogle = window.adsbygoogle || []).push({});
</script>
```

Once you are happy with the settings click on the generate button and the the plugin will generate Javascript code for the AdSense ad in the code window of the current block.

Banner AdSense ↑ ↓

## Rotation Editor

To rotate different ad versions separate them with `[ADINSERTER ROTATE]` - Ad Inserter will randomly select one of the ad versions. For example, to rotate 3 different images you can use the following code:

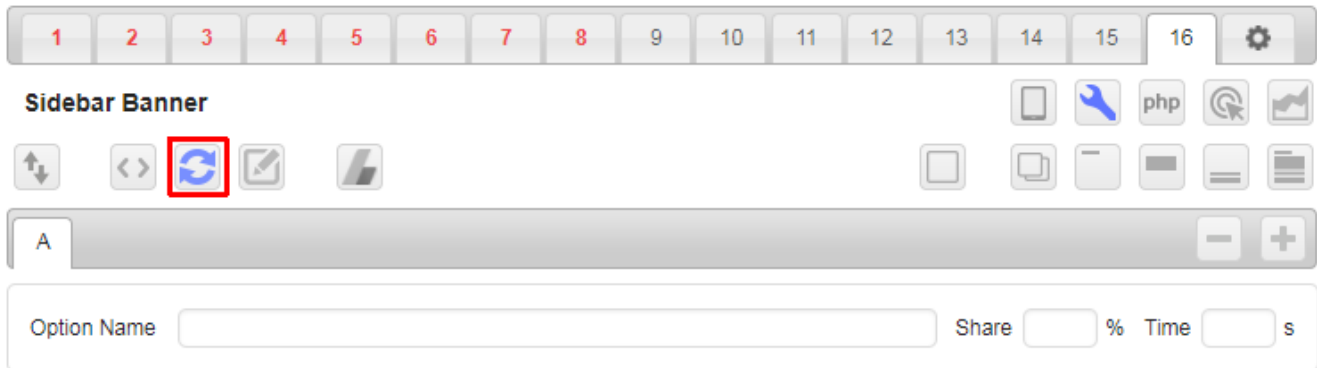
```

[ADINSERTER ROTATE]

[ADINSERTER ROTATE]

```

However, there is an easier way to create code for rotation. Use rotation editor available on the code tools toolbar.



Rotation editor will **AUTOMATICALLY import** rotation code from the code window (if present). In our case it would generate three options: A, B and C. You can use **+** and **-** buttons to add or remove options.

You can also specify **rotation share** (percentage) for each option. If you omit one share value it will be automatically calculated. Leave **Share** field empty for all options to have equal shares of all rotation options.

Ad Inserter supports also **timed rotation** – for each option you can define rotation time. When at least one option has defined time, the plugin will generate client-side code (using CSS / javascript code) that will show one option for the time specified. If an option has no time defined the rotation will stop there. If you set time to 0 then this option will be skipped. When you define also share values for timed rotation, share values will be used only for the initial display of the first chosen option.



When rotation editor is active the **code window shows ONLY code for the selected option**. For each option

you can copy/paste code, use code generator or visual HTML editor. Each option can also be named. This name is then used in statistics reports (Pro only) and is displayed when debugging function Label blocks is used.

After importing the code for the first option you can name each option. Please note that when switching rotation options and using code generator you need to import (and generate) code manually for each option. **Once you are happy with the rotation settings click on the rotation editor button and the the plugin will generate rotation code in the code window of the current block.** Now you can save settings.



**WARNING:** Do not save settings when the rotation editor is active – after you are finished with rotation settings click on the rotation button to generate rotation code with `[ADINSERTER ROTATE]` separator.

**WARNING:** If you are using caching ad rotation by default may not work as expected. It works only when the page is generated and Ad Inserter is called. In such cases please make sure you have disabled caching or use client-side **Dynamic blocks** - but you need to be aware of how it works. Please check section Caching - Ad rotation for details. Timed rotation always generates client-side code – rotation code runs in the browser.

Ad Inserter is perfect for displaying any kind of ads. It can also be used to display various versions of ads, for example [AdSense ads using channels](#) to test which format or color combination performs best – but please, in such case use **Dynamic blocks** set to **Server-side** or **Client-side insert** to prevent generating hidden ads.

You can also rotate ads configured in other blocks. In such case enable shortcuts for those blocks and use shortcodes for rotation. For example, this code will rotate blocks 10, 11 and 12.

```
[ADINSERTER code="10"]
```

```
[ADINSERTER ROTATE]
```

```
[ADINSERTER code="11"]
```

```
[ADINSERTER ROTATE]
```

```
[ADINSERTER code="12"]
```

In this case we have used shortcode `[ADINSERTER code="10"]` that will return only pure block code (without wrapping div with alignment CSS code). You can also use shortcode `[ADINSERTER block="10"]` that will return code with wrapping div and alignment CSS code. In such case you can rotate between different alignments – just make sure to set alignment of the rotation block to **No wrapping**.

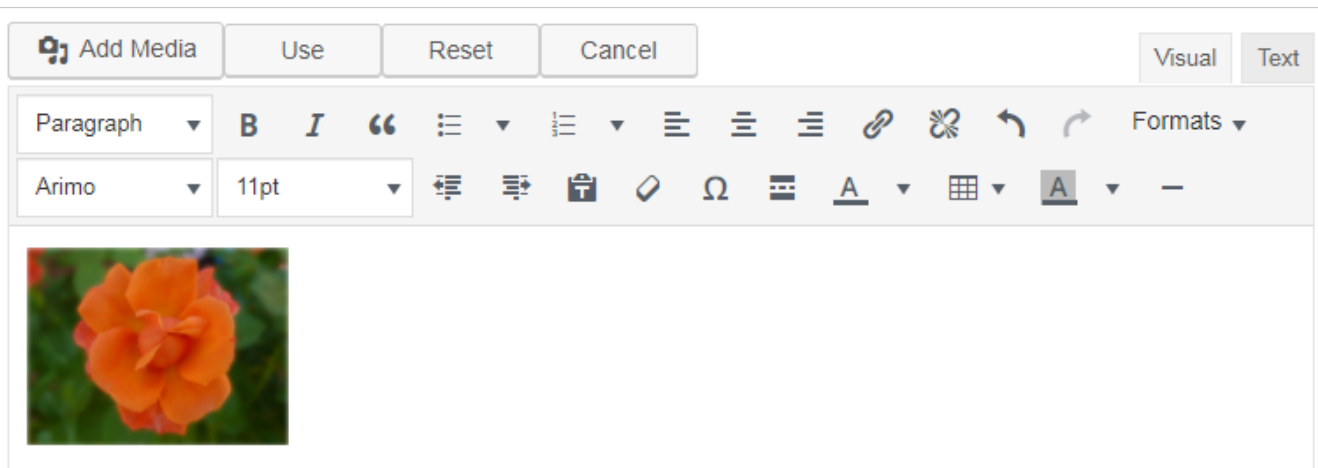
## Visual HTML editor

Ad Inserter has also standard WordPress visual HTML editor to create or edit image or text ads. The button for it is located on the code tools toolbar.



The editor opens in a new window where you can edit the code the same way as you edit WordPress posts and pages. You can insert images and switch between visual editor and source code. There are also three additional buttons:

- **Use** Copies current code to the code window and closes visual code editor
- **Reset** Resets the code in the editor to the code from the code window in current block settings
- **Cancel** Cancels editing and closes visual code editor



## Clipboard

Ad Inserter has internal clipboard which can be used to copy and paste block name, code and settings. The buttons are located on the code tools toolbar.



Copy button copies current block to the internal clipboard. Once the block is copied the button **copy** is highlighted.



You can then switch to any of the remaining blocks and paste either block name, code, insertion settings or everything: name, code and settings.



To clear block settings click on the **Clear block** button.



With buttons in the code tools toolbar you can copy settings from any of 16 displayed code blocks. But you can also copy settings from any other code block. Click on the **List** button in the header to show the list of all (or only active) blocks. Click small button left of block number to copy block settings to the internal clipboard. Then you can use paste functions to paste name, code, settings or complete block.

Block	Name	Automatic Insertion	Alignment	P	S	W	Widget Positions
<input type="checkbox"/>	3 Amazon Sidebar	Disabled	Default	●	●	●	sidebar-a
<input type="checkbox"/>	6 16-Channel USB Logic Analyzer	Disabled	Default	●	●	●	sidebar-a
<input type="checkbox"/>	8 AdSense Bottom	After Post	Default	●	●	●	
<input type="checkbox"/>	9 Amazon / AdSense Sidebar	Disabled	Default	●	●	●	sidebar-a





## Insertion Options

Each code block has 4 independent insertion options – automatic and three manual:

- Automatic Insertion
- Widget
- Shortcode
- PHP function call

Normally, for each code block you use only one insertion option. Of course, you can use all 4 options simultaneously taking into account that all insertion options use the same block settings (with some exceptions mentioned below). To insert code block (and display ad) at some position **you need to enable and use at least one insertion option and enable at least one page type**.

## Automatic Insertion

Automatic insertion means that the code block will be automatically inserted on all posts or pages according to the block settings. For each code block you can also define exceptions to prevent insertions on some posts or pages. There are at least two different ways to exclude posts or pages - check Post/Page Exceptions for more.



Automatic Insertion: Before Paragraph ▼ Alignment and Style: Float Right ▼ Show

Automatic insertion uses WordPress hooks – triggers that are called when WordPress (with theme and other plugins) creates pages. Some themes do not use standard Wordpress excerpts, posts or other page elements. In such cases some insertion positions will not be available. But you can still insert ads there with Ad Inserter by modifying theme files and adding calls to `adinsserter` PHP function. In such case do not modify theme files directly but create a [child theme](#) instead. The following positions are available for automatic insertion:

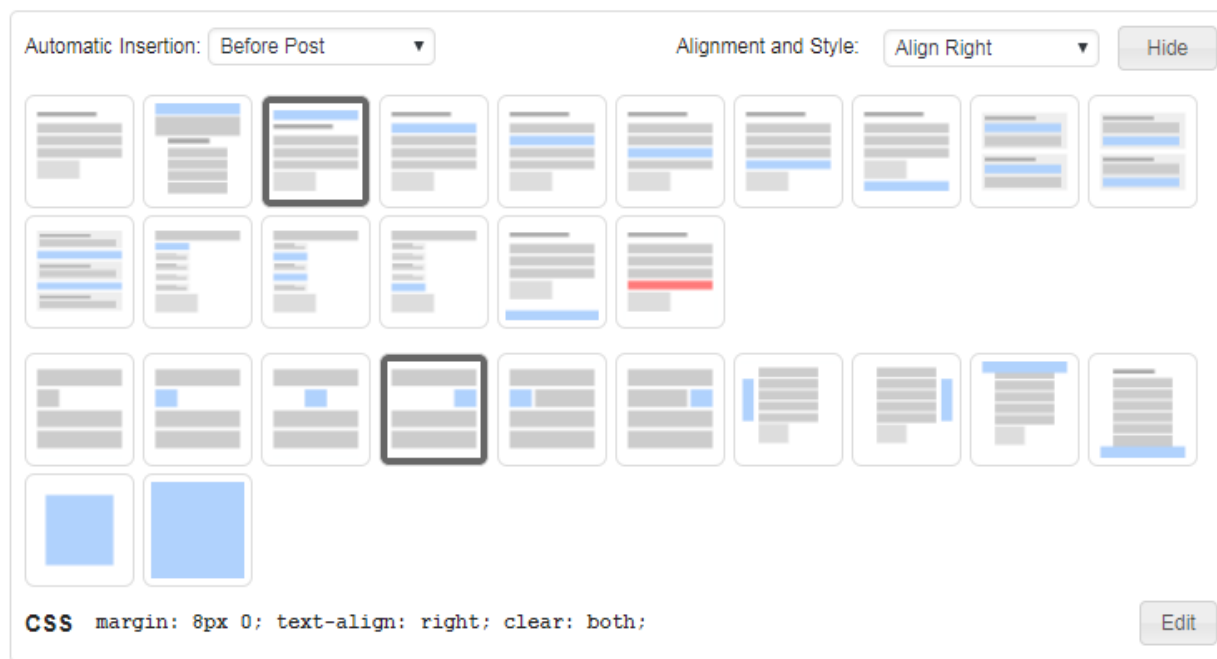
- Insert **Above header** (on all pages, after `<body>` tag, available only when **Output buffering** is enabled)
- Insert **Before Post** (before post or posts on blog pages)
- Insert **Before Content** (before post or page text)
- Insert **Before Paragraph** (on posts, pages and blog pages)
- Insert **After Paragraph** (on posts, pages and blog pages)
- Insert **After Content** (after post or page text)
- Insert **After Post** (after post or posts on blog pages)
- Insert **Before Excerpt** (on blog pages)
- Insert **After Excerpt** (on blog pages)
- Insert **Between posts** (on blog pages)
- Insert **Before Comments** (on posts)
- Insert **Between Comments** (on posts)
- Insert **After Comments** (on posts)
- Insert in page **Footer** (on all pages, before `</body>` tag)
- Insert **Before HTML element** (server-side insertion available only when **Output buffering** is enabled)
- Insert **After HTML element** (server-side insertion available only when **Output buffering** is enabled)

PLEASE NOTE: Insertion position **Above header** inserts code after `<body>` tag, while **Header** code in general plugin setting (tab ⚙) inserts the code (scripts) in the `<head>` section of the HTML page. Insertion position **Above header** needs special code processing and is **available only** when you enable **Output buffering** (tab ⚙ / General), however, this may not work with all installations.

For single posts or pages insertion position **Before Post** usually means position above the post/page title, for blog pages **Before Post** position means position above all the posts on the blog page. For single posts or pages insertion position **After Post** means position below the post/page after all the content, for blog pages **After Post** position means position below all the posts on the blog page.

Before and after **Excerpt** positions are available on blog pages only if your theme uses standard Wordpress excerpts. Otherwise you'll have to modify theme files and use PHP function [adinsserter](#) or custom hook in order to insert ads between excerpts.

Between posts work with most themes that use standard WP blog pages. However, some themes call the `_post` hook before the content is actually written. In such case this position would not work as expected.



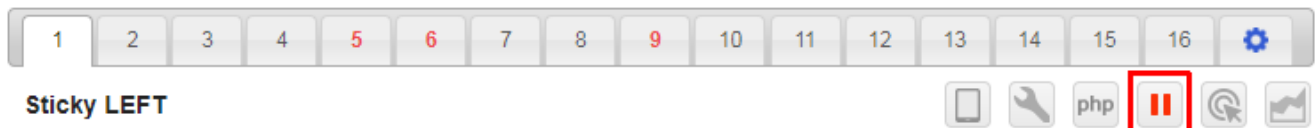
Order of insertion positions in a typical post is the following:

- **[Above Header]**
- Theme header
- **[Before Post]**
- Post Title
- **[Before Content]**
- Paragraph 1
- Paragraph ...
- Paragraph n - 1
- Paragraph n
- **[After Content]**
- Output of some other plugins
- **[Before Comments]**
- Comments
- **[After Comments]**
- Output of some other plugins
- **[After Post]**
- Theme footer
- **[Footer]**

Of course, the final order of items depends also on other plugins. Ad Inserter is by default called as one of the last plugins. You can change Plugin priority on the settings page (tab ⚙️). Use **Show positions** function to see available positions for automatic insertion (Ad Inserter menu item in the WordPress toolbar on the top of every post/page) check Debugging section for details.

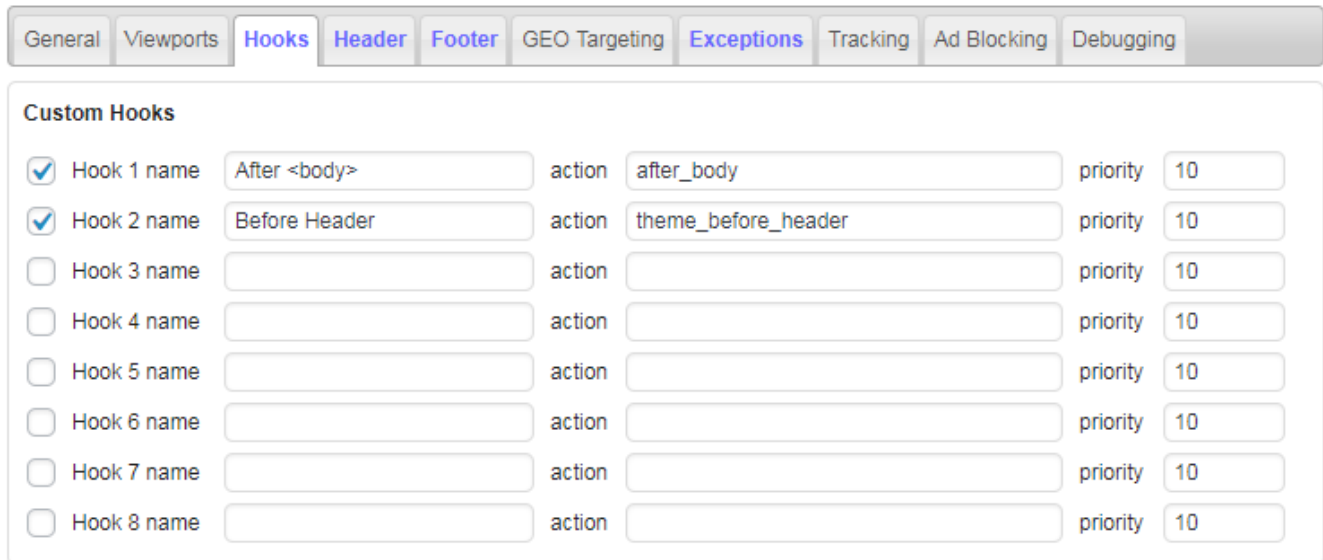
## Disable insertion

To insert code block you need to use automatic or manual insertion. Once you configure settings including insertion, the code will be inserted. If for some reason you need to temporarily disable (pause) insertion you can disable automatic insertion or uncheck manual insertion. To achieve the same effect without changing the settings you can simply disable (pause) insertion of each code block. To disable insertion of code block click on the **pause button** (it will be red when paused) and save settings.



## Custom Hooks

Ad Inserter supports also automatic insertion at custom positions where hooks are used using the `do_action ()` WordPress function. The number of custom hooks depends on the license type (free Ad Inserter supports 2 custom hooks). To define hooks go to tab ⚙️ / tab **Hooks**. Example of hooks settings:



Hook name is a name that is used in the Automatic insertion selection, action is the name that is used in the `do_action ()` WordPress function in the theme and priority is the hook priority (lower number means earlier processing). You can use ANY hook that is implemented by your theme or added by plugins. After you save settings you get new positions for automatic insertion.

Hook is a standard WordPress PHP function call to allow themes and other plugins to insert custom code. Instead of inserting `adinsserter` PHP function call you can install custom hook at any theme position:

```
<?php do_action ('custom_action_name'); ?>
```

The difference between `adinsserter` PHP function call and custom hook is that `adinsserter` PHP function will insert only the defined code block, while custom hook position is available as position for automatic insertion and can be used for any code block.

## Insertion Before or After HTML element

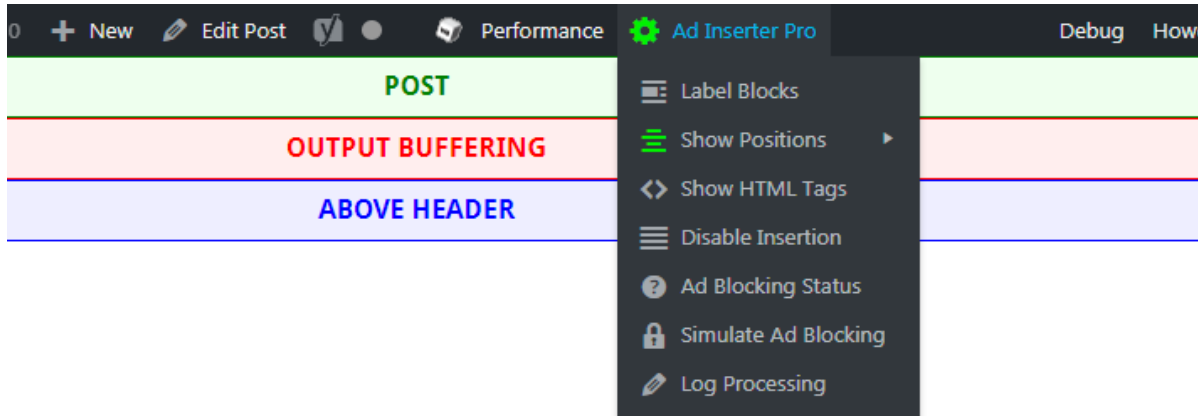
Insertion positions **Before** and **After HTML element** can be used to insert any code anywhere on the page. Ad Inserter supports client-side and server-side insertion. Client-side insertion uses Javascript code to insert code block during or after the page is loaded in the browser while server-side insertion means insertion when the plugin is called and the page is created. However, server-side insertion requires enabled output buffering (tab / tab General).

**HTML element(s):** here you list CSS selector(s) of the HTML element(s) where you want to insert the code block (`#id`, `.class`).

**Insertion: Client-side** insertion inserts the code immediately when the code at Javascript code positions runs. Wanted HTML element must be already available (above) on the page. **Client-side when DOM ready** (default) inserts the code after the page is loaded and all elements are ready. **Server-side** insertion is the best approach, however, it requires **output buffering** which is by default disabled as it may not work with all themes. To enable it go to general plugin settings (tab / tab General) and set **Output buffering** to enabled. After you save settings you will be able to select server-side for insertion before/after HTML element.

**Javascript code position:** Page position where the Javascript code for **client-side** insertion will be inserted. Should be after the HTML element if not waiting for DOM ready.

**Output buffering** means buffering the page while it is created by WordPress using your theme and plugins. After the buffering ends the buffered page content is processed and code blocks before/after HTML elements inserted. To check if your theme supports output buffering, go to general plugin settings and enable it, then go to homepage and enable **Show positions** debugging function (Ad Inserter menu in top admin bar). If you see **OUTPUT BUFFERING** (and **ABOVE HEADER**) bar then output buffering works as expected.

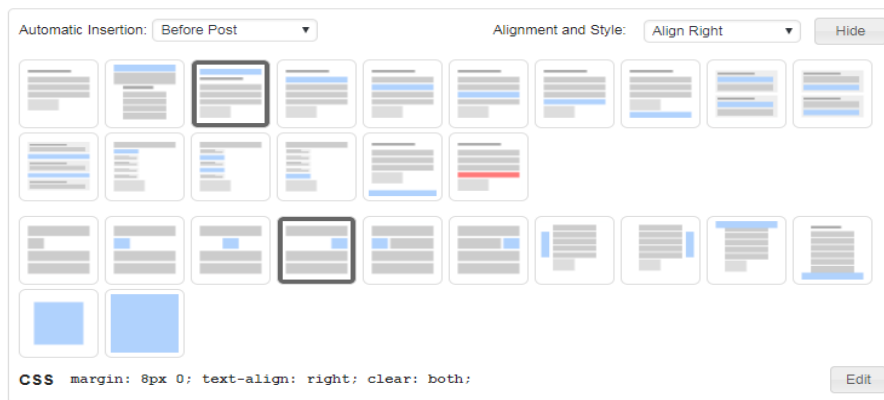


## Alignment and Style

Each code block or ad can be aligned according to your needs. Normally, the code block is wrapped with a `div` with appropriate CSS style. Of course, you can customize or define your own CSS code for the wrapping div. The following alignments are available:

- **Default** (simple div with thin margin around the block)
- **Align Left**
- **Center**
- **Align Right**
- **Float Left** (ad on left with wrapped text on right)
- **Float Right** (ad on right with wrapped text on left)
- **Sticky Left** (ad on the left, doesn't move when the page is scrolled, Pro only)
- **Sticky Right** (ad on the right, doesn't move when the page is scrolled, Pro only)
- **Sticky Top** (ad on the top, doesn't move when the page is scrolled, Pro only)
- **Sticky Bottom** (ad on the bottom, doesn't move when the page is scrolled, Pro only)
- **Custom CSS** (You can define custom CSS code for wrapping div)
- **No Wrapping** (leaves ad code as it is, otherwise it is wrapped by a div)

Click on the **Show** button next to **Alignment and Style** selection to see the graphical representation of the automatic insertion options and alignment styles. You can also test different alignments and styles - click **Preview** button to open the preview window. For details check Code Preview section and description below.



## Wrapping code

Each code (ad) block is normally wrapped by a div element. Classes and inline style can be configured on the General tab of plugin settings. For example, the default wrapping code for block 1 looks like this:

```
<div class="code-block code-block-1" style="margin: 8px 0; clear: both;">  
  AD_CODE  
</div>
```

The wrapping code contains the CSS style for thin margin around the code block (and optionally alignment) and Ad Inserter code block classes. You can change or add your own CSS code using Custom CSS style (to edit the CSS code simply click Show button to reveal the code and then click on it). But you can also apply the same style to all the code blocks using Ad Inserter CSS class (default `code-block`) and CSS code in the CSS file of your theme. Default class name can be changed in Ad Inserter plugin settings (Tab ⚙️ / tab General). You can also have empty **Block Class Name** and no classes will be added the the wrapping div. **Default, Left, Center**

and **Right** alignment styles contain also CSS code `clear: both;` which clears any floating elements above.

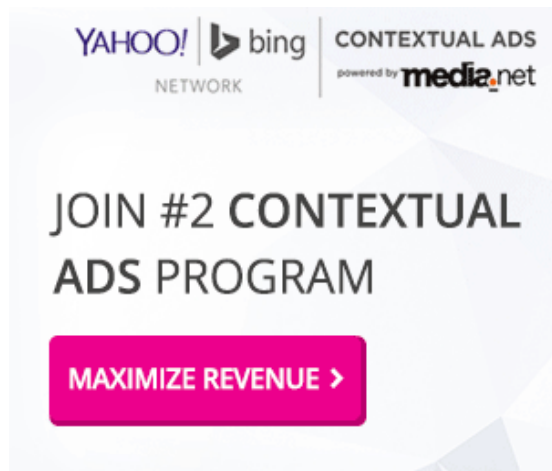
## Default Alignment

Default alignment simply wraps the code with a div with thin margin around the block. The block will be aligned according to the settings of the page. You should use default alignment unless you need to force specific alignment or wrap the ad with page text.



## Left Alignment

Left alignment wraps the code with a `div` with thin margin around the ad block and forces the ad to align left. This means that it aligns the ad along the left side of the page or containing element. You can also test and modify the code in **Preview** window. In Preview window you can also use alternative CSS code for left alignment. Double-click on the icon for Left alignment and the code will toggle. The current code version is displayed next to Alignment and Style selection.



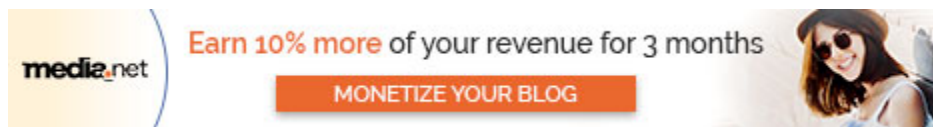
## Right Alignment

Right alignment wraps the code with a `div` with thin margin around the ad block and forces the ad to align right. This means that it aligns the ad along the right side of the page or containing element. You can also test and modify the code in **Preview** window. In Preview window you can also use alternative CSS code for right alignment. Double-click on the icon for Right alignment and the code will toggle. The current code version is displayed next to Alignment and Style selection.



## Center Alignment

Center alignment wraps the code with a `div` with thin margin around the ad block and centers the ad between left and right. This means that it aligns the ad to be equally spaced from the left and the right side of the page or containing element. You can also test and modify the code in **Preview** window. In Preview window you can also use alternative CSS code for centering the ad. Double-click on the icon for Center alignment and the code will toggle. The current code version is displayed next to Alignment and Style selection.



## Float Left

Float Left alignment wraps the code with a `div` with thin margin around the ad block and allows the text to continue right of the ad. This means that it wraps the ad on the right side with text. You can also test and modify the code in **Preview** window.



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni



dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?

## Float Right

Float Right alignment wraps the code with a `div` with thin margin around the ad block and allows the text to continue left of the ad. This means that it wraps the ad on the left side with text. You can also test and modify the code in **Preview** window.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?



## No Wrapping

No Wrapping style does not wrap the code with a `div` - it inserts the code as it is. The alignment will depend on the page settings. Use No Wrapping style when you don't want to add the wrapping `div`. **Please note that ad tracking, client-side device detection and client-side GEO targeting can't work without a wrapping `div`.**

## Custom CSS

Sometimes the predefined CSS styles need to be modified in order to match ad styles with the theme. You can start with any predefined style and modify it or add any CSS code. To edit CSS code simply click Show button to reveal the code and then click on it for editing. You can also visually edit CSS code and watch it real-time how the ad will look like. Save the settings and click **Preview** button to open the preview window.

Below the settings there is a preview of the code between two dummy paragraphs. Here you can test various block alignments, visually edit margin and padding values of the wrapping `div` or write CSS code directly and watch live preview. **Highlight** button highlights background, wrapping `div` margin and code area, while Reset button restores all the values to those of the current block. You can resize the window (and refresh the page to reload ads) to check display with different screen widths. Once you are satisfied with alignment click on the Use button and the settings will be copied to the active block where you can save them.

# Preview

## Block 1

### AdSense 728x90

HighlightUseResetCancelCSS EditBlock Alignment and Style: 

This is a preview of the saved code between two dummy paragraphs. Here you can test various block alignments, visually edit margin and padding values of the wrapping div or write CSS code directly and watch live preview. Highlight button highlights background, wrapping div margin and code area, while Reset button restores all the values to those of the current block.

You can resize the window (and refresh the page to reload ads) to check display with different screen widths. Once you are satisfied with alignment click on the Use button and the settings will be copied to the active block.

## Sticky Ads (Pro only)

Ad Inserter Pro supports also various types of sticky ads – ads that stick to the screen or content and stay fixed when the page is scrolled. It is also possible to configure sidebar ads that scroll with the content. There are also additional options like close button, delayed display and animations.

To configure sticky ad select **Sticky** as Alignment and Style. This will open settings for sticky ads:

All sticky ads will be displayed according to the settings, however, the ad code should normally be inserted in the page footer. Therefore, you should select **Footer** Automatic Insertion unless instructed otherwise. If your theme doesn't have **Footer** insertion position (and the sticky ads are not displayed) you can try to use any other suitable position (**Before post** or **After post**).

Sticky settings are divided into three sections:

- Horizontal position
- Vertical position
- Animation

While you can configure all the settings for sticky ads here, it is much easier to configure and test settings in the **Preview** window where you can also test all the alignments and animations. Click on the **Preview** button to open Preview window.

## Horizontal position

Horizontal position defines how sticky ad is positioned horizontally. Available options are:

- Stick to the left
- Stick to the content left
- Center
- Stick to the content right
- Stick to the right



### Stick to the left

This alignment displays ad next to the left screen edge - it overlaps any content there. Next to the selection you can optionally enter horizontal margin in pixels.



#### Preview

**Block 1**  
Sticky LEFT

Code block  
160px × 600px

background

Alignment and Style: Sticky

Close button: Top right

Horizontal position: Stick to the left

Vertical position: Stick to the top

Horizontal margin: px

Vertical margin: px

Animation: Turn

CSS position: fixed; z-index: 9995; top: 100px; left: 0px;

This is a preview of the code for sticky ads. Here you can test various horizontal and vertical alignments, close button locations, visually edit margin values or write CSS code directly and watch live preview. Highlight button highlights background, margin and code area, while Reset button restores all the values to those of the current block.

Ad Inserter can be configured to insert any code anywhere on the page. Each code with its settings is called a code block. Free Ad Inserter supports 16 code blocks, Ad Inserter Pro supports up to 96 code blocks (depending on the license type). The settings page is divided into tabs - 16 code blocks and general plugin settings. Black number means inactive code block (code is not inserted anywhere), red number means block is using automatic insertion, blue number means block is using manual insertion while violet number means block is using automatic and manual insertion.

Few very important things you need to know in order to insert code and display some ad: Enable and use at least one display option (Automatic Display, Widget, Shortcode, PHP function call). Enable display on at least one Wordpress page type (Posts, Static pages, Homepage, Category pages, Search Pages, Archive pages). Single pages (posts and static pages) have also additional setting for individual exceptions. Use default blank value unless you are using individual post/page exceptions.

## Stick to the content left

This alignment displays ad next to the left edge of page content (as sidebar) - it also overlaps any content there. Next to the selection you can optionally enter horizontal margin in pixels. Page content width is automatically calculated and it is assumed that the content is centered horizontally.

**Preview**

**Block 1**  
**Sticky LEFT**

Code block  
160px x 600px

background

Alignment and Style: Sticky

Close button: Top right

Horizontal position: Stick to the content left

Vertical position: Stick to the top

Horizontal margin: px

Vertical margin: px

Animation: Turn

```
CSS position: fixed; z-index: 9995; top: 100px; left: auto; margin-right: 10px; width: fit-content; display: none;
```

This is a preview of the code for sticky ads. Here you can test various horizontal and vertical alignments, close button locations, visually edit margin values or write CSS code directly and watch live preview. Highlight button highlights background, margin and code area, while Reset button restores all the values to those of the current block.

Ad Inserter can be configured to insert any code anywhere on the page. Each code with its settings is called a code block. Free Ad Inserter supports 16 code blocks, Ad Inserter Pro supports up to 96 code blocks (depending on the license type). The settings page is divided into tabs - 16 code blocks and general plugin settings. Black number means inactive code block (code is not inserted anywhere), red number means block is using automatic insertion, blue number means block is using manual insertion while violet number means block is using automatic and manual insertion.

Few very important things you need to know in order to insert code and display some ad: Enable and use at least one display option (Automatic Display, Widget, Shortcode, PHP function call). Enable display on at least one Wordpress page type (Posts, Static pages, Homepage, Category pages, Search Pages, Archive pages). Single pages (posts and static pages) have also additional setting for individual exceptions. Use default blank value unless you are using individual post/page exceptions.

## Center

This alignment displays ad centered horizontally - it overlaps any content there.

**Preview**

**Block 2**  
**Sticky TOP**

Code block  
720px x 90px

background

Alignment and Style: Sticky

Close button: None

Horizontal position: Center

Vertical position: Stick to the top

Horizontal margin: px

Vertical margin: px


Animation: Slide

```
CSS position: fixed; z-index: 9995; top: 0px; text-align: center; left: 50%; transform: translate(-50%);
```

This is a preview of the code for sticky ads. Here you can test various horizontal and vertical alignments, close button locations, visually edit margin values or write CSS code directly and watch live preview. Highlight button highlights background, margin and code area, while Reset button restores all the values to those of the current block.

## Stick to the content right

This alignment displays ad next to the right edge of page content (as sidebar) - it also overlaps any content there. Next to the selection you can optionally enter horizontal margin in pixels. Page content width is automatically calculated and it is assumed that the content is centered horizontally.



The screenshot shows the Ad Inserter configuration panel for a sticky ad. The 'Horizontal position' is set to 'Stick to the content right'. The 'Vertical position' is set to 'Stick to the top'. The 'Horizontal margin' is set to 10px. The 'Vertical margin' is set to 0px. The 'Animation' is set to 'Slide and Fade'. The CSS code is: `position: fixed; z-index: 9995; top: 100px; right: auto; margin-left: 10px; width: fit-content; display: none;`. The preview area shows a code block with dimensions 160px x 600px. A large red 'R' is overlaid on the right side of the interface.

## Stick to the right

This alignment displays ad next to the right screen edge - it overlaps any content there. Next to the selection you can optionally enter horizontal margin in pixels.

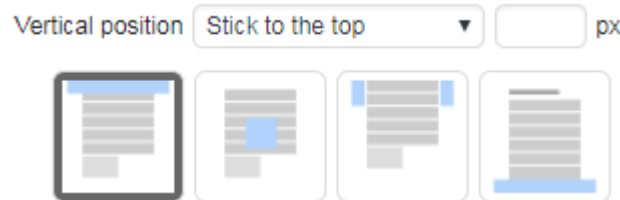


The screenshot shows the Ad Inserter configuration panel for a sticky ad. The 'Horizontal position' is set to 'Stick to the right'. The 'Vertical position' is set to 'Stick to the top'. The 'Horizontal margin' is set to 0px. The 'Vertical margin' is set to 0px. The 'Animation' is set to 'Slide and Fade'. The CSS code is: `position: fixed; z-index: 9995; top: 100px; right: 0px;`. The preview area shows a code block with dimensions 160px x 600px. A large red 'R' is overlaid on the right side of the interface.

## Vertical position

Vertical position defines how sticky ad is positioned vertically. Available options are:

- Stick to the top
- Center
- Scroll with the content
- Stick to the bottom



### Stick to the top

This alignment displays ad next to the top screen edge - it overlaps any content there. Next to the selection you can optionally enter vertical margin in pixels. The ad stays fixed there - it doesn't move when the page is scrolled.

**TOP**

#### Preview

**Block 2**  
**Sticky TOP**

Code block  
728px × 90px

background

Alignment and Style: Sticky Close button: None

Horizontal position: Center

Horizontal margin:  px

Vertical position: Stick to the top

Vertical margin:  px

Animation: Slide Animate

```
CSS position: fixed; z-index: 9995; top: 0px; text-align: center; left: 50%; transform: translate(-50%);
```

This is a preview of the code for sticky ads. Here you can test various horizontal and vertical alignments, close button locations, visually edit margin values or write CSS code directly and watch live preview. Highlight button highlights background, margin and code area, while Reset button restores all the values to those of the current block.

Ad Inserter can be configured to insert any code anywhere on the page. Each code with it's settings is called a code block. Free Ad Inserter supports 16 code blocks, Ad Inserter Pro supports up to 96 code blocks (depending on the license type). The settings page is divided into tabs - 16 code blocks and general plugin settings. Black number means inactive code block (code is not inserted anywhere), red number means block is using automatic insertion, blue number means block is using manual insertion while violet number means block is using automatic and manual insertion.

## Center

This alignment displays ad centered vertically - it overlaps any content there.

The screenshot shows the Ad Inserter settings for a sticky ad. The 'Alignment and Style' dropdown is set to 'Sticky'. The 'Close button' is set to 'Top right'. The 'Horizontal position' is set to 'Center' and the 'Vertical position' is also set to 'Center'. The 'Horizontal margin' is set to 0 px and the 'Vertical margin' is set to 600 px. The 'Animation' is set to 'Flip'. The CSS code is: `position: fixed; z-index: 9995; top: 50%; text-align: center; left: 50%; transform: translate(-50%, -50%);`. Below the settings, there is a preview of the code for sticky ads and a large red 'C' watermark.

## Scroll with the content

This alignment scrolls ad with the page. It stays fixed relative to the page (you can optionally define vertical margin in pixels) and when the page is scrolled the ad is scrolled with it. Please note that this vertical position requires code insertion above any other content just below the `<body>` tag – automatic insertion **Above header**. This position for automatic insertion is available when you enable **Output buffering** (tab ⚙️ / tab General). Check Automatic Insertion for details.

The screenshot shows the Ad Inserter settings for a scroll-with-content ad. The 'Alignment and Style' dropdown is set to 'Sticky'. The 'Close button' is set to 'None'. The 'Horizontal position' is set to 'Stick to the content right' and the 'Vertical position' is set to 'Scroll with the content'. The 'Horizontal margin' is set to 0 px and the 'Vertical margin' is set to 140 px. The 'Animation' is set to 'Flip'. The CSS code is: `position: fixed; z-index: 9995; position: absolute; margin-bottom: auto; width: fit-content; top: 140px; display: none; right: auto; margin-left: 10px; width: fit-content; display: none;`. Below the settings, there is a preview of the code for sticky ads and a large red 'R' watermark.




## Stick to the bottom

This alignment displays ad next to the bottom screen edge - it overlaps any content there. Next to the selection you can optionally enter vertical margin in pixels. The ad stays fixed there - it doesn't move when the page is scrolled.

The screenshot shows the 'Preview' window for 'Block 4' with the alignment set to 'Sticky BOTTOM'. The preview area shows a 'Code block' of 728px x 90px on a 'background'. The 'Alignment and Style' dropdown is set to 'Sticky'. The 'Close button' is set to 'None'. The 'Horizontal position' is set to 'Center' and the 'Vertical position' is set to 'Stick to the bottom'. The 'Horizontal margin' is 0 px and the 'Vertical margin' is -3 px. The 'Animation' is set to 'None'. The CSS code is: `position: fixed; z-index: 9995; bottom: -3px; text-align: center; left: 50%; transform: translate(-50%);`. Below the CSS code, there is a text block explaining the preview and the plugin's configuration options. At the bottom of the preview window, there is a blue bar with the text 'BOTTOM' in yellow.

## PLEASE NOTE

**Stick to the content** horizontal position displays ads next to the left or right edge of page content (as sidebar). In most cases the plugin will automatically calculate page content width (assuming that the content is centered horizontally) and position ads automatically. However, some themes may require to define main content element manually. For this go to tab  / tab General and define **Main content element**. You can enter either CSS selector (`#id` or `.class`) of the main (outermost) page element or content width in pixels (numerical value only).

The easiest way to configure and test alignment settings is in the **Preview** window where you can test all the alignments and animations. Click on the **Preview** button to open Preview window.

## Animation

Animation	None ▼	Close button	None ▼				
Trigger	Page loaded ▼	Offset	<input type="text"/> px	Delay	<input type="text"/> ms	Trigger once	<input type="checkbox"/>

Sticky ads can be displayed when the page is scrolled to some position, after some delay when the page is loaded. You can also define effect to show or hide the ad.

The following animations (effects) are available:

- Fade
- Slide
- Slide and Fade
- Turn
- Flip
- Zoom In
- Zoom out

For each horizontal and vertical alignment the plugin will select optimal animation direction. Some animations are not available with all positions. In such cases the plugin will choose the best alternative.

The easiest way to test animations is in the **Preview** window where you can also test alignments. Click on the **Preview** button to open Preview window.

Each animation has the following additional parameters:

- Animation trigger
- Trigger offset
- Trigger delay
- Trigger only once

### Animation trigger

The following triggers are available:

- **Page loaded** - The animation starts when the page is loaded.
- **Page scrolled (%)** - The animation starts when the page is scrolled for the defined percentage (specify only numerical value from 1 - 100).
- **Page scrolled (px)** - The animation starts when the page is scrolled for the defined number of pixels (specify only numerical value).
- **Element visible** - The animation starts when specific HTML element becomes visible. Specify CSS selector for the element (**#id** or **.class**).

### Trigger offset

This is additional offset in pixels that is taken into account when the animation is triggered (positive and negative values are possible).

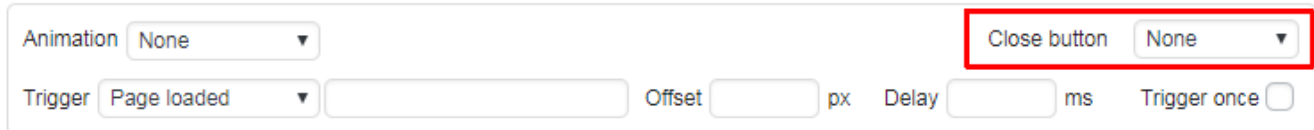
### Trigger delay

This defines delay of animation (in milliseconds) after the trigger event. To simply show sticky ad with a delay set animation to **Fade** and define delay.

### Trigger only once

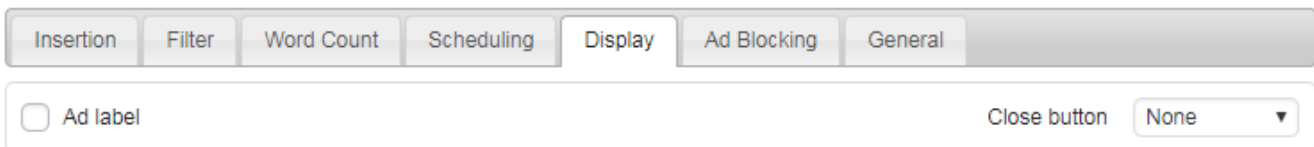
Normally the sticky ad will be displayed after the trigger condition occurs and will hide when the condition is not met. Here you can define to trigger animation only once – show the ad and don't hide it when the page is scrolled back.

### Close button



The screenshot shows a configuration panel with several settings. The 'Close button' dropdown menu is highlighted with a red border. The other settings are: Animation: None; Trigger: Page loaded; Offset: [empty] px; Delay: [empty] ms; Trigger once: [checkbox].

Each ad can have a close button. You can choose one of the four corners where the button will be displayed. The ad will be not be displayed until the page is reloaded. This setting is a mirror of the **Close button** setting on the **Misc / Display** tab. It is available also here for convenience.



The screenshot shows the 'Display' tab selected in the AdInserter Pro interface. The 'Ad label' checkbox is unchecked. The 'Close button' dropdown menu is set to 'None'.

The easiest way to test animations is in the **Preview** window where you can also test alignments. Click on the **Preview** button to open Preview window.

# Preview

**Block 1**  
**Sticky LEFT**

Code block  
160px × 600px

background

Highlight

Use

Reset

Cancel

Alignment and Style Sticky

Close button Top right

Horizontal position Stick to the left

Horizontal margin px

Vertical position Stick to the bottom

Vertical margin px

Animation Turn Animate

**CSS** `position: fixed; z-index: 9995; bottom: 100px; left: 0px;` Edit

This is a preview of the code for sticky ads. Here you can test various horizontal and vertical alignments, close button locations, visually edit margin values or write CSS code directly and watch live preview. Highlight button highlights background, margin and code area, while Reset button restores all the values to those of the current block.

## Code Preview

Ad Inserter has a very useful function that can help you to check if the ad code is working and to see how it will look like when it will be inserted. Click on the **Preview** button to open Preview window.

### WARNING

Some ad blockers may block code on the Ad Inserter preview window. If you see warning PAGE BLOCKED or you don't see your code and the widow elements are distorted, **make sure you have disabled ad blockers**. On the top of the preview window there is visual CSS editor, four buttons, CSS code of the wrapping div (which can be edited manually) and **Block Alignment and Style** selection.

720 px

# Preview

## Block 1

### AdSense 728x90

margin 20 px  
padding 10 px

Code block  
704px × 60px

wrapping div 20 px background

Highlight  
Use  
Reset  
Cancel

CSS margin: 20px 0px; padding: 10px 0; Edit

Block Alignment and Style: Custom CSS

This is a preview of the saved code between two dummy paragraphs. Here you can test various block alignments, visually edit margin and padding values of the wrapping div or write CSS code directly and watch live preview. Highlight button highlights background, wrapping div margin and code area, while Reset button restores all the values to those of the current block.

## Vat Warehousing service

Saving VAT with our Storage Area MGB Logistics Distribution



You can resize the window (and refresh the page to reload ads) to check display with different screen widths. Once you are satisfied with alignment click on the Use button and the settings will be copied to the active block.

Below the settings there is a **preview of the saved code** between two dummy paragraphs. Here you can test various block alignments, visually edit margin and padding values of the wrapping div or write CSS code directly and watch live preview. Highlight button highlights background, wrapping div margin and code area, while Reset button restores all the values to those of the current block. You can resize the window (and refresh the page to reload ads) to check display with different screen widths. Once you are satisfied with alignment click on the Use button and the settings will be copied to the active block.

720 px

# Preview

## Block 1

AdSense 728x90

margin 20 px

padding 10 px

Code block

704px × 60px

wrapping div 20 px

background

Highlight

Use

Reset

Cancel

CSS margin: 20px 0px; padding: 10px 0; Edit

Block Alignment and Style: Custom CSS

This is a preview of the saved code between two dummy paragraphs. Here you can test various block alignments, visually edit margin and padding values of the wrapping div or write CSS code directly and watch live preview. Highlight button highlights background, wrapping div margin and code area, while Reset button restores all the values to those of the current block.



You can resize the window (and refresh the page to reload ads) to check display with different screen widths. Once you are satisfied with alignment click on the Use button and the settings will be copied to the active block.

Please note that the **code displayed here is the code that is saved** for this block, while block name, alignment and style are taken from the current block settings (may not be saved). No Wrapping style inserts the code as it is so margin and padding can't be set. However, you can use own HTML code for the block. Preview window uses also Header and Footer code.

## Page types

For all insertion positions you should also define Wordpress page types where the code blocks can be inserted (and ads displayed).

There are basically two page types in WordPress: single pages (posts and static pages) and blog pages. Single page displays single post or static page content while blog pages list few posts usually with short post excerpts. Typical blog pages are category pages which list posts in a category.

**PLEASE NOTE:** Regardless of other settings **you need to enable insertion on AT LEAST ONE PAGE TYPE.**

Wordpress has basically two page types:**blog pages** and **single pages**. You can enable insertion for each page type.



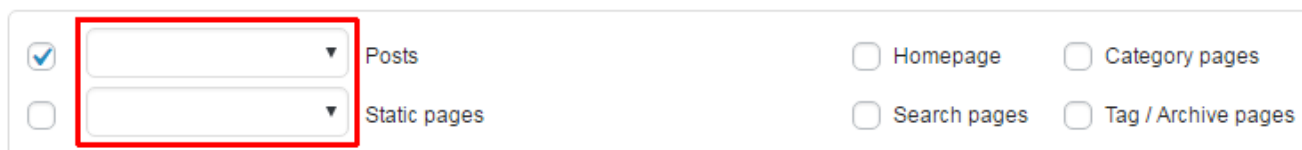
The screenshot shows the 'Page Types' section of the Ad Inserter settings. It features two rows of settings. The first row is for 'Posts', with a checked checkbox and a dropdown menu. The second row is for 'Static pages', with an unchecked checkbox and a dropdown menu. To the right of these rows are four checkboxes: 'Homepage', 'Category pages', 'Search pages', and 'Tag / Archive pages', all of which are currently unchecked.

Themes or plugins may add also other page types. Automatic insertion on such pages is not possible unless theme or plugin uses standard WordPress hooks on which Ad Inserter relies.

Single pages are posts and static pages. They show one article or page. Ad Inserter labels:

- Posts
- Static Pages

Single pages have also additional setting for exceptions. If you are not using post/page individual exceptions (on post/page editor pages) then use default BLANK value. Even if you use lists to filter urls, IDs, categories or tags you should use BLANK (meaning no individual posts or page exceptions) as these settings affect only individual exceptions on post/page editor pages.



This screenshot is identical to the one above, but with a red rectangular box highlighting the two dropdown menus for 'Posts' and 'Static pages'.

Blog pages are pages that show few latest posts with excerpts (or full posts). Such blog pages are category pages, tag pages, archive pages and also homepage if set to show latest posts. If your theme has custom homepage or the homepage is set as static page then homepage is not blog page and automatic insertion will not be possible as on typical blog pages. Ad Inserter labels:

- Homepage
- Category pages
- Search Pages
- Archive pages

Because each page type has different page layout, all positions for automatic insertion are not available on all page types. The word content used in the context of automatic insertion means the content of post or static page (paragraphs that make the post or static page). Some themes use content hooks also on blog pages. In such cases you can also use content positions on blog pages. Use debugging functions to show available positions.

Available only on single pages:

- Before/after paragraph
- Before/after content

Available only on blog pages:

- Before/after excerpt (only if WP excerpts are used)
- Between posts

Available on all page types:

- Before header (available only when **Output buffering** is enabled)
- Footer

Available on all page types (if supported by the theme):

- Before/after post

Insertion (located in the Misc section / tab Insertion) is possible also for:

- AMP pages
- Ajax requests (e.g. for infinite scroll)
- 404 page (Error 404: Page not found)
- RSS Feed

#### **Please note**

- For shortcodes and PHP function calls it is possible to ignore enabled page types and use them on any page. See **Button Manual - Manual Insertion** for details.
- Many themes use custom homepage layout and do not use standard WP hooks so some positions for automatic insertion are not available. Check also Custom Hooks.
- Use **Show positions** debugging function to see available positions for automatic insertion check Debugging section for details.
- In most cases you should enable each code block **ONLY either on single pages or on blog pages**. DO NOT enable both unless you are absolutely sure this is what you need and it works fine with your theme. Only position Before/After Post is normally available on all page types.



## Post/Page Exceptions

You can also disable insertion of ads on certain posts or pages. For each code block on posts or static pages you first define default insertion settings for posts/pages page type:

Individually disabled ▼ Posts

Homepage
  Category pages

Individually disabled ▼ Static pages

Search pages
  Tag / Archive pages

- enabled **BLANK** – no individual exceptions, insert on all posts/pages and ignore individual exceptions if they exist
- enabled **Individually disabled** – insert on all posts/pages except where insertion is individually disabled for post or page
- enabled **Individually enabled** - insert only on individual posts/pages where insertion is enabled

Then you can define individual post/page exceptions **on the post/page editor page** (check **Ad Inserter Individual Exceptions** meta box below each post). Exceptions work only on page/post content (positions Before Content, Before Paragraph, After Paragraph, After Content).

**PLEASE NOTE:** Shortcodes **IGNORE** post/static page exception settings! However, you can insert shortcodes that ignore enabled page types with `[adinsserter block="BLOCK_NUMBER" ignore="page-type"]`.

For one or few exceptions it is easier to first enable insertion on Posts/Pages page types using BLANK selection (no individual exceptions) and then either white or black-list single url or few space-separated urls (click on the **Lists** button).

Ad Inserter Exceptions ▲				
Block	Name	Automatic Insertion	Default insertion for posts	For this Post
1	<a href="#">AdSense</a>	Before Content	Disabled	<input type="checkbox"/> Enabled
2	<a href="#">Block 2</a>	Disabled	Enabled	<input type="checkbox"/> Disabled
3	<a href="#">Block 3</a>	Disabled	Enabled on all posts	
4	<a href="#">Block 4</a>	After Paragraph	Enabled on all posts	
5	<a href="#">Block 5</a>	Disabled	Enabled on all posts	
6	<a href="#">Block 6</a>	Disabled	Enabled on all posts	
7	<a href="#">Block 7</a>	Disabled	Enabled on all posts	
8	<a href="#">Block 8</a>	Before Content	Enabled on all posts	
9	<a href="#">Block 9</a>	Disabled	Enabled on all posts	
10	<a href="#">Block 10</a>	Disabled	Enabled on all posts	
11	<a href="#">Facebook</a>	Disabled	Enabled on all posts	
12	<a href="#">Block 12</a>	Disabled	Enabled on all posts	
13	<a href="#">Block 13</a>	Disabled	Enabled on all posts	
14	<a href="#">Block 14</a>	Disabled	Enabled on all posts	
15	<a href="#">Block 15</a>	Disabled	Enabled on all posts	
16	<a href="#">Block 16</a>	Disabled	Enabled on all posts	

Default insertion for posts for each code block can be configured on [Ad Inserter Settings page](#): enabled / disabled on all posts (no exceptions) or enabled / disabled by default with exceptions set here.

Enabled on all posts or Disabled on all posts means that exceptions are not enabled. Use On all except selected or Only on selected to enable exceptions on this page for each block.

Post and static page exceptions are normally enabled only for administrators. You can define minimum user role for exceptions on plugin settings page (tab ⚙️):

General
Viewports
Header
Footer
Ad Blocking
Debugging

Syntax Highlighter Theme Ad Inserter ▼

Block Class Name wrapper

Min. User Role for Ind. Exceptions Editing Administrator ▼

Dynamic blocks Server-side ▼

Functions for Paragraph Counting Standard ▼

No Paragraph Counting Inside blockquote, figure, li, div

Sticky Widget Top Margin 45 px

Plugin priority 99999

When code block has individual exceptions set on posts or static pages, there will be a small button next to Posts text. Click on this button shows a list of all exceptions for this code block. Click on the ID number opens post/page, while click on the title opens post/page editor to edit exceptions. Click on the small red cross in the top right corner clears all exceptions for this code block.

Individually disabled ▼ Posts

Homepage  Category pages

▼ Static pages

Search pages  Tag / Archive pages

ID	Type	Title	✕
8	Post	How To Start A Successful Blog	✕

Ad Inserter Pro has also additional **Exceptions** tab on the plugin settings tab ⚙️. Here you can open page with exceptions, open editor to edit exceptions, clear exceptions for specific block, clear exceptions for specific post/page or clear all exceptions (click on the cross in the top right corner).

General
Viewports
Header
Footer
GEO Targeting
Exceptions
Debugging

ID	Type	Page / Post	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	✕
31608	Page	Activity	X																✕
2	Page	About	X																✕
31609	Page	Members	X																✕
31615	Post	Lorem Ipsum	X																✕
31516	Post	How To Become A Good Essay Writer	X																✕
31514	Post	Travel Risk Management	X																✕

## Paragraphs

Use insertion option **Before** or **After** paragraph to automatically insert ads between paragraphs.

Paragraph number(s)


Paragraph number(s) for Automatic Insertion options Before and After Paragraph:

- value 1 or more means paragraph number
- negative value -1 or less means paragraph number counted from the opposite direction as configured
- value between 0 and 1 means relative position in post or page (e.g. 0.3 means paragraph 30% from top or bottom)
- 0 means random paragraph position – each time a different paragraph will be selected
- Empty value means all paragraphs – use Filter to define insertions

You can also specify comma separated list of paragraph numbers for multiple insertion, for example: **0.3, 0.6** would insert at 30% and at 60% of post paragraphs.

## Counting

Paragraphs can be counted from top or from bottom. It is also possible to count only paragraphs that contain/do not contain certain text or count only paragraphs that have some minimum or maximum number of words. If more than one text is defined (comma separated) and "contain" is selected then the paragraph must contain ALL texts.

Paragraphs are not counted inside special elements like `<blockquote>`, `<figure>` and `<li>`. Of course, there is an option to enable counting also inside these special elements and you can define these special elements as **No Paragraph Counting Inside** setting on the tab  / tab General.

Count  paragraphs with tags  that have between  and  words  
and  Minimum number of paragraphs   
 Count inside special elements Minimum number of words in paragraphs above

**Please Note:** Paragraph processing works on every post or page according to settings. Therefore, if you enable insertion also on blog pages (home, category, archive, search pages) and your theme does not display post excerpts but complete posts, Ad Inserter will by default insert code blocks into ALL posts on the blog page (according to settings). To enable insertion only into specific post(s) on blog pages define Filter. You can also leave Filter (click Misc button) empty (means all posts on the blog page) and define maximum number of insertions.

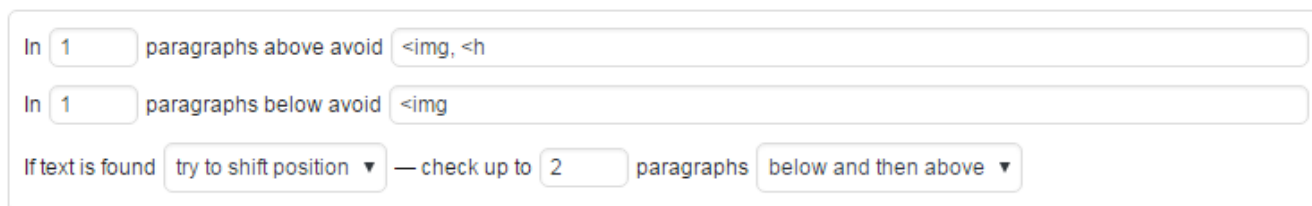
You can also define paragraph HTML tags. Normally only `<p>` tags are used. If your post contains also `<div>` or header tags, you can define comma separated list of tags used to count paragraphs (e.g. p, div, h2, h3). Use debugging function **Show HTML tags** to visualize HTML tags used in post on page.

WARNING: Each code block you insert in post adds one `<div>` block unless you use No wrapping style. Before Paragraph will insert code before `<tag>`, After Paragraph will insert code after closing `</tag>`.

PLEASE NOTE: Image element `<img>` has no closing tag which is used for **After paragraph** insertion, however, Ad Inserter can still properly detect image element and insert after it.  
Use `#` as tag if paragraphs have no tags and are separated with the `\r\n\r\n` characters.

**Minimum number of paragraphs** / Minimum/Maximum page/post words: do not insert if the number of paragraphs or the number of words is below or above limit (used only for position Before or After selected paragraph).

**Minimum number of words in paragraphs above:** insert only if the number of words in paragraphs above is above some limit (used only for position Before or After selected paragraph).



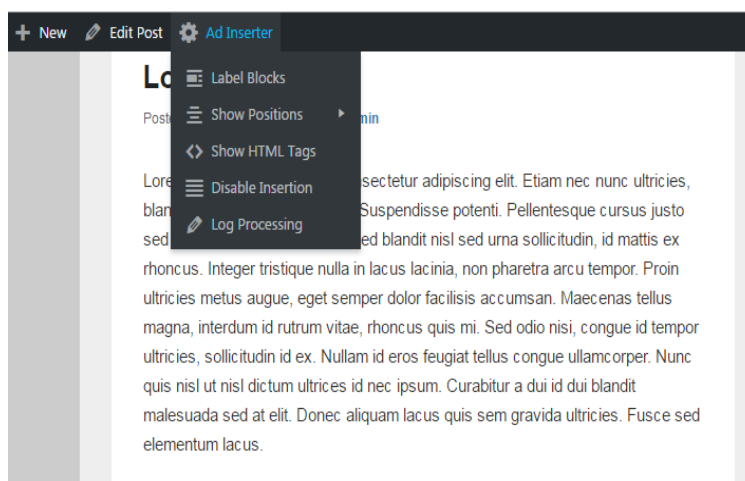
## Clearance

If you define **only one paragraph number** you can define parameters to avoid insertion at paragraph positions where above or below is some unwanted element (heading, image, title). This is useful to avoid inserting ads where they may not look good or where it is not allowed.

You can define in how many paragraphs above and below should specified text be avoided. And if the text is found you can choose to either skip insertion or try to shift insertion position up or down up to the specified number of paragraphs.

On every post/page there is a toolbar on the top. Ad Inserter menu item has few functions to visualize HTML tags and positions for automatic insertion:

- **Show HTML tags:** visualizes HTML tags
- **Show positions:** shows available positions for automatic insertion. It uses paragraph tags for blocks configured for After or Before paragraph



## Ad rotation

To rotate different ad versions separate them with `[ADINSERTER ROTATE]` - Ad Inserter will randomly select one of the ad versions. For example, to rotate 3 different images you can use the following code:

```

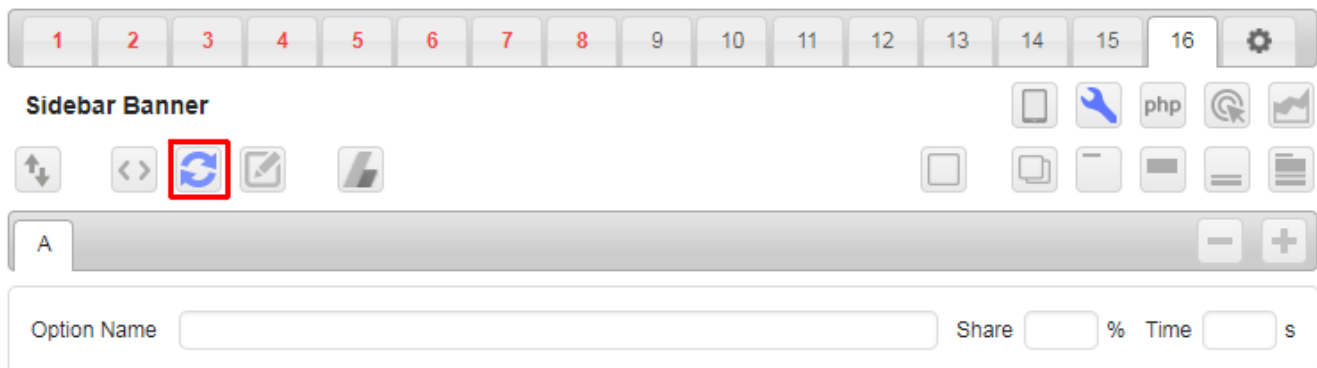
[ADINSERTER ROTATE]

[ADINSERTER ROTATE]

```

**PLEASE NOTE:** Previous versions of the plugin used the `|rotate|` separator for rotation. This separator will continue to work.

There is an alternative way to create code for rotation. Use rotation editor available on the code tools toolbar.



Rotation editor will **AUTOMATICALLY import** rotation code from the code window (if present). In our case with the code above it would generate three options: A, B and C.



When rotation editor is active the **code window shows ONLY code for the selected option**. For each option you can copy/paste code, use code generator or visual HTML editor. Each option can also be named. This name is then used in statistics reports (Pro only) and is displayed when debugging function Label blocks is used.

After importing the code for the first option you can name each option. Please note that **when switching rotation options** and using code generator **you need to import (and generate) code manually for each option**. Once you are happy with the rotation settings **click on the rotation editor button** and **the plugin will generate rotation code** in the code window of the current block.



**WARNING:** If you are using caching ad rotation by default may not work as expected. It works only when the page is generated and Ad Inserter is called. In such cases please make sure you have disabled caching or use client-side **Dynamic blocks** - but you need to be aware of how it works. Please check section Caching - Ad rotation for details.

Ad Inserter is perfect for displaying any kind of ads. It can also be used to display various versions of ad, for example [AdSense ads using channels](#) to test which format or color combination performs best – but please, in such case set **Dynamic blocks** to either **server-side** or **client-side insert** – to prevent generating hidden ads.

You can also specify rotation share (percentage) for each option. Either use rotation code generator and define shares in percents or create code using `[ADINSERTER ROTATE share="percentage"]` separator with attribute `share` where you define percentage of the rotation share:

```
[ADINSERTER ROTATE share="75"]
ad_code_1
[ADINSERTER ROTATE share="25"]
ad_code_2
```

If you omit one share value it will be automatically calculated:

```
ad_code_1
[ADINSERTER ROTATE share="25"]
ad_code_2
```

Ad Inserter supports also **timed rotation** – for each option you can define rotation time. When at least one option has defined time, the plugin will generate client-side code (using CSS / Javascript code) that will show one option for the time specified. If an option has no time defined the rotation will stop there. If you set time to 0 then this option will be skipped. When you define also share values for timed rotation, share values will be used only for the initial display of the first chosen option.

The following code will show ad1 for 30 s and then ad2 for 10 s and then ad1 for 30 s...

```
[ADINSERTER ROTATE time="30"]
ad_code_1
[ADINSERTER ROTATE time="10"]
ad_code_2
```

You can also rotate ads configured in other blocks. In such case enable shortcuts for those blocks and use shortcodes for rotation. For example, this code will rotate blocks 10, 11 and 12.

```
[ADINSERTER code="10"]
[ADINSERTER ROTATE]
[ADINSERTER code="11"]
[ADINSERTER ROTATE]
[ADINSERTER code="12"]
```

In this case we have used shortcode `[ADINSERTER code="10"]` that will return only pure block code (without wrapping div with alignment CSS code). You can also use shortcode `[ADINSERTER block="10"]` that will return code with wrapping div and alignment CSS code. In such case you can rotate between different alignments – just make sure to set alignment of the rotation block to **No wrapping**.

## Ad counting

To insert different ads when there are multiple insertions on the page, separate the ads with `[ADINSERTER COUNT]` - Ad Inserter will select one ad according to the insertion counter. For example, to insert 3 different images between first four posts on homepage (if supported by theme) you can set **Automatic insertion** to **Between posts**, set Filer to **1, 2, 3** and enable insertion on **Homepage** and then use the following code:

```

[ADINSERTER COUNT]

[ADINSERTER COUNT]

```

Ad Inserter will insert image 1 between posts 1 and 2, image 2 between posts 2 and 3 and image 3 between posts 3 and 4. Normally the order of insertion of codes will be the same as configured. Use `[ADINSERTER COUNT="shuffle"]` for **FIRST** separator only to shuffle the codes so the insertion will be randomized.

**PLEASE NOTE:** Previous versions of the plugin used the `|count|` separator for counting. This separator will continue to work.

## Buttons

Below the code window there are buttons which toggle additional settings for code block.

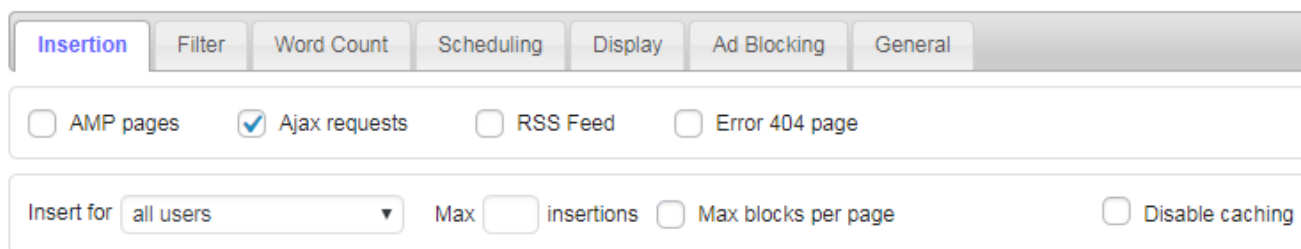


## Button Misc



Misc section is divided into tabs.

## Insertion



By default code blocks will not be inserted on AMP pages, Error 404 page (Page Not Found) and in RSS feeds. Check **AMP Pages**, **Error Page** or **RSS Feed** checkbox to enable insertion there. **Ajax** requests (e.g. for infinite scroll) are by default enabled, but you can uncheck option to disable insertion on Ajax calls.

Each code block can be inserted for:

- All users (default)
- Logged in users
- Not logged in users
- Administrators (**useful for testing / debugging**)

**WARNING:** If you are using **caching** this may not work as expected. The check works only when the page is generated and Ad Inserter is called. Make sure you have disabled caching when you are using such settings. Check Caching section for details.

For each code block you can limit how many times on the page the block (or ad) will be inserted. There are two settings for this. **Max N insertions:** this is a simple limit for the first N insertions for the block. You can also limit maximum number of blocks per page. This limit is defined on the general plugin settings page (tab ⚙️ / tab General). If you want this block to count for maximum blocks per page check **Max blocks per page**.

There is also an option to **disable caching** on specific pages. This option will prevent caching of pages where code blocks with this option enabled are inserted. This option works with WP Super Cache, W3 Total Cache and WP Rocket caching plugins. After you enable this option make sure you flush the cache.



## Scheduling

The screenshot shows the 'Scheduling' tab selected in a navigation bar. Below the navigation bar, there is a form with the following elements: a dropdown menu labeled 'Insert between dates' with a downward arrow, two text input fields containing '2017-08-01' and '2017-08-31' separated by the word 'and', and a dropdown menu labeled 'fallback' with '1 - AdSense' selected and a downward arrow.

Here you can define when the code block will be inserted:

- Insert immediately: no scheduling
- Delay insertion for N days after publishing: waits for N days after the time when the post was published (available only for before/after content and before/after paragraph)
- Insert between dates START\_DATE and END\_DATE (**Pro only**): inserts code only between START\_DATE (including) and END\_DATE (excluding) - on START\_DATE it starts inserting, on END\_DATE it ends inserting. You can also define fallback code block to be inserted when the original block expires.

## Display

The screenshot shows the 'Display' tab selected in a navigation bar. Below the navigation bar, there is a form with the following elements: two checkboxes labeled 'Ad label' and 'Lazy loading', a 'Close button' label, and a dropdown menu with 'None' selected and a downward arrow.

Here you can enable ad label (usually Advertisements) above the ad as defined under general plugin options (tab ⚙ / tab **General** / **Ad label**).

You can also enable **Lazy loading** (Pro only). Lazy loading means that the code block (ad) is not loaded (inserted) until it is visible in the viewport (or it is close to become visible). This may decrease impressions for that block as the impression is not counted until the ad is actually loaded and it is not loaded until the page is scrolled to make this ad visible.

How close to the viewport (in pixels) the ad must be in order to be loaded can be defined on the general plugin settings (tab ⚙ / tab **General**) – **Lazy loading offset**.

Code blocks can have close button (Pro only). Useful mainly for sticky ads. Choose position of the button.

## Word Count

The screenshot shows the 'Word Count' tab selected in a navigation bar. Below the navigation bar, there is a form with the text 'Post/Static page must have between' followed by two empty text input fields separated by the word 'and', and the word 'words' at the end.

Here you can define minimum and maximum post/page word length. Empty value means no limit.

## Filter

This setting is a filter to limit/filter insertions when the theme inserts code block more than once (for example between posts or between comments). For example, when using **Between posts** automatic insertion Filter set to [2, 4, 6](#) means insertion after posts 2, 4, and 6. You can also leave Filter number empty (means all insertions) and define maximum number of insertions (tab **Insertion**) which is equivalent to Filter [1, 2, 3, ..., N](#).

**Filter** defines which insertions are enabled - single number or comma separated numbers. [%N](#) filter item means filter every N-th insertion. Filter also works for paragraph counting. Leave **Paragraph number** empty, set filter to **Paragraphs** and define comma separated list of paragraph numbers where the code should be inserted ([%N](#) can also be used). **Invert Filter** inverts insertion – listed insertions are disabled.

This is useful in many cases where you can't remove unwanted insertions of the code with other settings:

- If you need to insert an ad before the first, third and fifth excerpt on the homepage you simply specify [1, 3, 5](#) for the filter.
- If you need to insert an ad between the second and third comment and between the fourth and fifth comment on posts you set [2, 4](#) for the filter.
- If you need to insert an ad every fifth comment on posts you set [%5](#) for the filter.
- In some WP themes hooks (that call Ad Inserter insertion functions) are called more than once. In such case you might get unwanted insertions. Simply set the filter to the number of the wanted call(s). Use debugging function **Show positions** on every post/page to show available positions for automatic insertion with counters.
- If you use [adinsserter](#) PHP function and you don't want that for each time the function is called on the page the code is inserted, you can simply filter calls.
- If you only need the first N calls (insertions) then leave filter empty and use Max N insertions instead.

**Please Note:** Paragraph processing works on every post or page according to settings. Therefore, if you enable insertion also on blog pages (home, category, archive, search pages) and your theme does not display post excerpts but complete posts, Ad Inserter will by default insert code blocks into ALL posts on the blog page (according to settings). To enable insertion only into specific post(s) on blog pages define Filter. You can also leave Filter empty (means all posts on the blog page) and define maximum number of insertions.

## Ad Blocking (Pro only)

This tab is described under Ad Blocking Detection chapter.

## General

Insertion   Scheduling   **Display**   Word Count   Filter   Ad Blocking   General

General tag

General tag: text used for `[adinserteer data='tag']` and `[adinserteer data='smart-tag']` shortcodes if the post has no tags - useful for contextual ads - **works only inside posts/static pages!**

## Button Lists - Black and White Lists

Lists   Manual   Devices   Misc   Preview   **Save All Settings**

Black lists and white lists are two ways of filtering insertion. White list means that insertion will be enabled only for the items on the white list, while black list means that insertion will be disabled for all the items on the black list.

Lists are used to limit insertion on pages/posts that have something in common (category, tag, taxonomy, url pattern) or to limit insertion based on visitor's data (referrer, IP address, country). Click on the **Lists** button to show available lists.

Categories	<input type="checkbox"/>	<input type="text"/>	<input checked="" type="radio"/> Black List	<input type="radio"/> White List
Tags	<input type="checkbox"/>	<input type="text"/>	<input checked="" type="radio"/> Black List	<input type="radio"/> White List
Taxonomies	<input type="checkbox"/>	<input type="text"/>	<input checked="" type="radio"/> Black List	<input type="radio"/> White List
Post IDs	<input type="checkbox"/>	<input type="text"/>	<input checked="" type="radio"/> Black List	<input type="radio"/> White List
Urls	<input type="checkbox"/>	<input type="text" value="/about, /contact"/>	<input checked="" type="radio"/> Black List	<input type="radio"/> White List
Url parameters	<input type="checkbox"/>	<input type="text"/>	<input checked="" type="radio"/> Black List	<input type="radio"/> White List
Referers	<input type="checkbox"/>	<input type="text"/>	<input checked="" type="radio"/> Black List	<input type="radio"/> White List
IP Addresses	<input type="checkbox"/>	<input type="text"/>	<input checked="" type="radio"/> Black List	<input type="radio"/> White List
Countries	<input type="checkbox"/>	<input type="text"/>	<input checked="" type="radio"/> Black List	<input type="radio"/> White List

- Leave list empty and set it to Black List to disable the list (default setting).
- If you white-list items on the list, at least one of the items on the list must be present to enable insertion (item1 OR item2 OR item3...).
- If you black-list items on the list, at least one of the items on the list must be present to disable insertion (item1 OR item2 OR item3...) i.e. no item on the list should be present to enable insertion.
- List item separator is comma.
- For categories and tags use slugs (name used in the url - web address).
- Click on the small button next to list to open list editor. For categories, tags, taxonomies, post IDs and countries you can select items from the list of the available items. For urls, url parameters, referers and IP addresses you can enter one item per line in the editor.
- If there is more than one non-empty list defined for the same code block please note the following: all lists must pass the check for insertion (list1 AND list2 AND list3...).

PLEASE NOTE: if you are using lists you don't need to define individual exceptions for post or static pages - leave default value blank unless you are using individual post/page exceptions:

The screenshot shows a settings panel with several options. A red box highlights two dropdown menus. The first dropdown is selected with a checkmark and is labeled 'Posts'. The second dropdown is not selected and is labeled 'Static pages'. To the right of these are four checkboxes: 'Homepage', 'Category pages', 'Search pages', and 'Tag / Archive pages'.

## Categories

List categories (category slugs) where you want insertion (white list) or don't want insertion (black list). The list can be used for posts and category pages. Click on the small button left of the list to open category selection.

## Tags

Do not insert in posts with certain tags (black list) or insert only in posts with certain tags (white list). Use comma separated tag slugs. The list can be used for posts and tag / archive pages. Click on the small button left of the list to open tag selection.

The screenshot shows a 'Tags' section with a text input field containing 'oak-flooring, ower-back-pain'. There are two radio buttons: 'Black List' (selected) and 'White List'. Below the input is a search bar and a list of tags. A double-headed arrow points from the list to a selection box on the right. The selection box contains 'oak-flooring (oak flooring)' and 'ower-back-pain (ower back pain.)'.

## Taxonomies (also Users and Post Types)

Do not insert ads in posts with certain custom taxonomies or taxonomy terms (black list) or insert ads only in posts with certain taxonomies (white list). Leave list empty and set it to Black list to insert ads for all taxonomies. You can use the following comma separated taxonomy slugs: `taxonomy`, `term` or `taxonomy:term`. Click on the small button left of the list to open taxonomy selection.

This list works in addition to Categories and Tags lists as categories and tags are default taxonomies. Therefore, it is possible to have categories and tags in one list, for example `category:category_slug, post_tag:tag_slug`. This way you can list categories and tags at the same time. Of course, if you black-list some category with Categories list or tag with Tags list then even if Taxonomies list is empty those categories or tags would be black-listed.

## Users, User Roles and Authors

You can also use Taxonomies list to filter users with `user:username` or user roles with `user-role:role`. This can be used to not insert ads for certain users or user roles (black list) or to insert ads only for certain users or user roles (white list). For example, to filter all users with Administrator role use `user-role:administrator`, to filter user with username editor2 use `user:editor2`.

Similarly you can use `author:author-username` to filter posts by author.

## Post Types

In Taxonomies list you can also define post types. Use `post-type:POST_TYPE`. For example to black-list portfolio post types you should black-list `post-type:portfolio`.

## Post IDs


Do not insert ads in posts/pages with certain post IDs (black list) or insert ads only in posts with certain IDs (white list). Use this list only if you need to specify IDs or there are only few posts to include/exclude, otherwise use individual Post/Page Exceptions when there are more posts to filter. Click on the small button left of the list to open post/page ID selection.

## Urls

Use to filter urls (web addresses). Do not insert ads on pages with certain urls (black list) or insert ads only on pages with certain urls (white list). Url used here is everything starting from the / after the domain name. For example: if web address is `http://domain.com/lorem-ipsum`, url to white/black-list is `/lorem-ipsum`. You can also use partial urls with `*`. To filter all urls starting with `/url-start` use `/url-start*`, to filter all urls that contain `url-pattern` use `*url-pattern*`, to filter all urls ending with `url-end` use `*url-end`. Use this list only if you need to include/exclude few urls or when urls are not posts or pages, otherwise if you need to filter posts or pages use individual Post/Page Exceptions instead. Click on the small button left of the list to open url editor.


## Url parameters and Cookies

This list checks url parameters (PHP variable `$_GET`) and cookies (PHP variable `$_COOKIE` or Javascript cookies if **Client-side Dynamic blocks** are configured). Do not insert ads on pages with certain url query parameters or cookies (black list) or insert ads only on pages with certain url parameters or cookies (white list). You can specify either parameters or parameters with values. For example for url `http://example.com?data=2&customer-id=22&device=0` you can define url parameters `data`, `customer-id=22` to insert ad only for urls where there is data parameter and customer-id parameter with value 22. Separate parameters with comma.

The same applies also to cookies (use `cookie_name` or `cookie_name=value`). Click on the small button left of the list to open url parameter editor. Url parameters are always checked server-side. Cookies are checked according to the setting **Dynamic blocks** (tab  / tab General).

## How to show ads based on visitor's consent?

Ad Inserter supports inserting (showing) ads based on cookies or cookie values. You can define cookie conditions by black/white-listing **Url parameters** (where also **cookies are checked**).

**PLEASE NOTE:** If you are using caching (very likely) you also need to set **Dynamic blocks** to **Client-side insert** (tab  / tab General) in order to check cookies in visitor's browser and not when the page is generated.

For example, If you are using one of the following plugins for cookie consent you need to **white-list the following cookie and value in Url parameter list**:

- |  |            |  |
|--|------------|--|
| • <a href="#">GDPR Cookie Consent</a>    | White-list | <code>viewed_cookie_policy=yes</code>    |
| • <a href="#">Cookie Notice for GDPR</a> | White-list | <code>cookie_notice_accepted=true</code> |
| • <a href="#">Cookie Consent</a>         | White-list | <code>catAccCookies=1</code>             |

## Referers

Do not insert ads to users from certain referers (domains) e.g [technorati.com](#), [facebook.com](#),... (black list) or insert ads only for certain referrers (white list): use # for no referer (direct visit). Click on the small button left of the list to open referer editor.

**WARNING:** If you are using [caching](#), set **Dynamic blocks** to **Client-side insert**. Use debugger widget to show and debug referers (server-side). Check Caching section for details.









## IP addresses (Pro only)

Do not insert ads for certain IP addresses (black list) or insert ads only for certain IP addresses (white list). List should contain comma separated IPv4 or IPv6 addresses. You can also use partial IP addresses (or ranges) with \*. To filter all IP addresses starting with [ip-address-start](#) use [ip-address-start\\*](#), to filter all IP addresses that contain [ip-pattern](#) use [\\*ip-pattern\\*](#), to filter all IP addresses ending with [ip-end](#) use [\\*ip-end](#). Click on the small button left of the list to open IP address editor.

## Countries (Pro only)

GEO targeting - do not insert ads for certain countries (black list) or insert ads only for certain countries (white list). Country list should contain comma separated country ISO Alpha-2 codes. You can also use 6 country groups which you can define on the Ad Inserter settings page (tab \*, Geolocation). Click on the small button left of the list to open country selection.

Countries   Black List  White List

ger	
 Algeria (DZ)	 Belgium (BE)
 Germany (DE)	 France (FR)
 Niger (NE)	 Luxembourg (LU)
 Nigeria (NG)	 United Kingdom (GB)

Country is determined from the IP address of the website visitor. IP address database is updated automatically each month. You can choose between Webnet77 (default) and MaxMind geolocation database (Ad Inserter Pro will download and use GeoLite2 data created by [MaxMind](#)). Ad Inserter Pro supports also [CloudFlare geolocation](#).

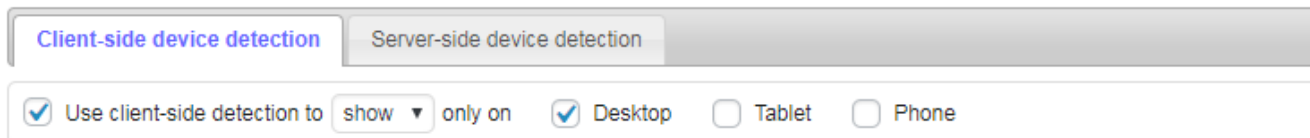
**PLEASE NOTE:** country code EU means non-specific European Union location - use individual country codes to target EU countries (use GB for United Kingdom) and AP means non-specific Asia-Pacific location - use individual country codes to target this region.

**WARNING:** If you are using caching, IP address detection and GEO targeting by default may not work as expected. It works only when the page is generated and Ad Inserter is called or when client-side IP address and country detection is enabled. Check Caching - IP address and country detection (Pro only) for details.

## Button Devices – Device detection

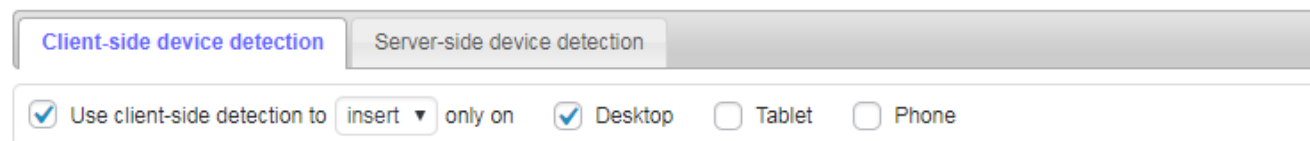


If you would like to insert ads for all devices then you don't have to use this function. Use it only if you would like to display ads or codes only for specific devices (desktop, tablets, phones). **IMPORTANT:** There are two types of device detection: **server side** and **client-side** and **client-side** type has two actions: **show** and **insert**.



**Client-side** detection of mobile/desktop devices works always as it is done in visitor's browser. CSS media queries and viewport (browser's screen) width are used to show/hide or insert Ad Inserter code blocks. You can define devices (name and minimum screen width) as **Viewports** on tab ⚙️ / tab Viewports.

**Client-side** detection has two actions: **show** and **insert**:



Both actions use viewports and CSS media queries to detect screen size:

- **Show:** Shows code blocks for enabled devices, for other devices code blocks remain hidden
- **Insert:** Inserts code blocks for enabled devices using Javascript code when the page is loaded

**PLEASE NOTE:** In most cases you should use **ONLY client-side** detection type. Works perfectly with responsive designs as they use CSS media queries. Of course, when using **Insert** action the code will be inserted only when the page is loaded so when you resize the page the code blocks will not change – you need to reload the page to re-insert code blocks for new viewport.

**BUT BE CAREFUL:** Some ad networks (like AdSense) limit ads per page and do not allow hiding ads (action **Show**). The ads are still inserted (loaded and counted) for all devices, but for unwanted devices they are hidden by the browser using CSS media queries based on viewport widths. In such case you should use AdSense ads with sizes **Fixed by viewport** as described in the AdSense code generator section.

**PLEASE NOTE:** some ad codes use old JavaScript method `document.write` which can't be used asynchronously (after the page is loaded and created). **Insert** action will not work with such ad codes.

Viewport Settings used for client-side device detection

Viewport 1 name	<input type="text" value="Desktop"/>	min width	<input type="text" value="980"/>	px
Viewport 2 name	<input type="text" value="Tablet"/>	min width	<input type="text" value="768"/>	px
Viewport 3 name	<input type="text" value="Phone"/>	min width	<input type="text" value="700"/>	px
Viewport 4 name	<input type="text"/>	min width	<input type="text" value="0"/>	px
Viewport 5 name	<input type="text"/>	min width	<input type="text" value="0"/>	px
Viewport 6 name	<input type="text"/>			

Up to 3 viewport names and widths can be defined on the Ad Inserter Settings tab ⚙ / tab Viewports (Ad Inserter Pro supports up to 6 viewports). Default values are:

- Desktop: 980 pixels or more
- Tablet: from 768 pixels to 979 pixels
- Phone: less than 768 pixels

**Server-side** detection:

- Desktop devices
- Mobile devices (tablets and phones)
- Tablet devices
- Phone devices
- Desktop and tablet devices
- Desktop and phone devices

**Server-side** detection of mobile/desktop devices works only when Ad Inserter plugin is called. It is called by Wordpress when it needs to generate a page. However, when you are using caching, it saves created page for quicker serving. In such cases **the user might get (saved) page for wrong device** (used by some previous visitor who triggered page caching). To solve this issue use themes that generate separate pages for desktop and mobile devices or use Mobile Theme Switcher plugin. Server-side detection uses User-Agent string combined with specific HTTP headers to detect the environment.

Client-side device detection
  Server-side device detection

Use server-side detection to insert code only for  devices

**PLEASE NOTE:** Use server-side device type detection only when you need to server-side insert (display and count) ONLY code blocks for specific device type. In all other cases switch it off.

**PLEASE NOTE:** If you are using **No Wrapping** style and need to hide code on some devices using client-side detection (CSS Media Queries) then you need to add appropriate class to your CSS code ([ai-viewport-1](#), [ai-viewport-2](#), [ai-viewport-3](#)). This doesn't apply to widgets as they always contain a wrapping div.

By default code blocks will not be inserted on Error 404 page (Page Not Found). Check Error '404 Page checkbox' to enable code block on error 404 page (button Misc / tab Insertion).



## Special Insertions

By default code blocks will not be inserted on AMP pages, Error 404 page (Page Not Found) and in RSS feeds. Check **AMP pages**, **Error Page** or **RSS Feed** checkbox to enable insertion there (button Button Misc). **Ajax** requests (e.g. for infinite scroll) are by default enabled, but you can disable insertion on Ajax calls for each code block.

## Button Manual - Manual Insertion

- Lists
- Manual
- Devices
- Misc
- Preview

Save All Settings

In addition to automatic insertion Ad Inserter supports also manual insertion of codes with widgets, shortcodes and PHP function calls. Insertion is called manual as you need to manually do something in order to insert the code (place widget in the sidebar, insert shortcode in the post, insert PHP function call into theme PHP file). The insertion is then, of course, automatic. Click on the **Manual** button to open Manual section.

- Widget Main Sidebar
- Shortcode `[adinsserter block="1"]` or `[adinsserter name="AdSense 728x90"]`
- PHP function `<?php if (function_exists ('adinsserter')) echo adinsserter (1); ?>`

## Disable insertion

To insert code block you need to use automatic or manual insertion. Once you configure settings including insertion, the code will be inserted. If for some reason you need to temporarily disable (pause) insertion you can disable automatic insertion or uncheck manual insertion. To achieve the same effect without changing the settings you can simply disable (pause) insertion of each code block. To disable insertion of code block click on the **pause button** (it will be red when paused) and save settings.



## Widgets

Widgets for all code blocks are enabled by default - simply drag Ad Inserter widget to any widget position (e.g. Sidebar), select code block, save and you're done. Optionally you can enable **Sticky widget** - this means that this widget (and widgets below) will stay fixed in the sidebar when the page is scrolled. You can also make other widgets sticky even if you don't use Ad Inserter widgets - drag Ad Inserter widget to the sidebar ABOVE the top widget that needs to be sticky, select Dummy Widget for Block, check Sticky and save widget. **Use only one sticky widget in each sidebar!** In general plugin settings (tab ⚙) you can also define **Sticky Widget Top Margin** to precisely define top position where sticky widget will stick. There are two modes for sticky widgets (or sticky sidebars) available as **Sticky widget mode** in general plugin settings (tab ⚙):

- **CSS mode**  
**This mode is the best approach** but may not work with all themes. CSS mode works by changing sidebar CSS to `position: sticky`. This works in most themes but not all. If your widgets and sidebar aren't sticking as expected you can use **JavaScript** mode described below.

If you would like to try to make CSS mode working the first thing to check are the rules applied to the sidebar parent containers. Specifically, look for any overflow property set on the parent. You can't use: `overflow: hidden`, `overflow: scroll` or `overflow: auto` on the parent of a `position: sticky` sidebar. If your theme is not using overflow and still having problems it's worth checking if a height is set on the parent. This may constrain the sticky positioning, stopping it from occurring. Remove the height and see if that fixes the problem.

- **JavaScript mode**  
This mode should work with practically all themes but may reload ads on page load when sticky sidebar is initialized. Use only when CSS mode does not work.

## Shortcodes

Insert shortcode `[adinsserter block="BLOCK_NUMBER"]` or `[adinsserter name="BLOCK_NAME"]` into post or page HTML code to insert block with BLOCK\_NAME name or BLOCK\_NUMBER number at this position. PLEASE NOTE: Shortcodes **IGNORE individual** post/static page exception settings! To force checking individual post/page exceptions use shortcode `[adinsserter block="BLOCK_NUMBER" check="exceptions"]`.

You can also insert shortcode that ignores enabled page types with `[adinsserter block="BLOCK_NUMBER" ignore="page-type"]`. To insert only block code (no wrapping div with alignment CSS code) use `[adinsserter code="BLOCK_NUMBER"]`.

## PHP function call

```
<?php if (function_exists ('adinsserter')) echo adinsserter (BLOCK_NUMBER); ?>
```

 - Insert code block BLOCK\_NUMBER at any position in template file. You can also define Filter for PHP function - define which call(s) to the function will actually insert code. This is useful if you put a call to the `adinsserter` function inside a loop in a template file (e.g. for homepage) and you need to insert ads only few times between posts. You can also use PHP function calls that ignore enabled page types 

```
<?php if (function_exists ('adinsserter')) echo adinsserter (BLOCK_NUMBER, 'page-type'); ?>
```

In addition to PHP function calls Ad Inserter supports also Custom Hooks. This is a standard WordPress PHP function call to allow themes and other plugins to insert custom code. Instead of inserting `adinsserter` PHP function call you can install custom hook at any theme position:

```
<?php do_action ('custom_action_name'); ?>
```

The difference between `adinsserter` PHP function call and custom hook is that `adinsserter` PHP function will insert

only the defined code block, while custom hook position is available as position for automatic insertion and can be used for any code block.

## Special Shortcodes

To rotate different ad versions separate them with `[ADINSERTER ROTATE]` - Ad Inserter will randomly select one of the ads.

**WARNING:** If you are using caching ad rotation may not work as expected. Normally it works only when the page is generated and Ad Inserter is called. In such cases please make sure you either disable caching or use **Client-side Dynamic Blocks** – check caching and Ad rotation for more.

To insert different ad on each call (insertion) for the code block (when the page is create and plugin called) separate ads with `[ADINSERTER COUNT]` - Ad Inserter will sequentially select one of the ads.

In the code blocks you can use the following shortcodes:

- `[adinsserter data='title']` - Title of the post
- `[adinsserter data='short-title']` - Short title (first 3 words) of the post title
- `[adinsserter data='category']` - Category of the post (or short title if there is no category)
- `[adinsserter data='short-category']` - First words before ", " or "and" of the category of the post (or short title if there is no category)
- `[adinsserter data='tag']` - The first tag or general tag if the post has no tags (**works only inside posts**)
- `[adinsserter data='smart-tag']` - Smart selection of post tag in the following order:
  - If there is no tag then the category is used;
  - If there is a two-word tag then it is used;
  - If the first tag is a substring of the second (or vice versa) then the first tag is not taken into account
  - If the first and second tags are single words then both words are used
  - First three words of the first tag
  - General tag
- `[adinsserter data='search-query']` - Search engine query that brought visitor to your website (supports Google, Yahoo, Bing and Ask search engines), `[adinsserter data='smart-tag']` is used when there is no search query. You need to disable caching to use this tag. Please note that most search queries are now encrypted.
- `[adinsserter data='country-iso2']` - Country ISO Alpha-2 lowercase code (based on visitor's IP address, **Pro only**, works only when server-side **Dynamic blocks** are used), use `[adinsserter data='country-ISO2']` for uppercase code.
- `[adinsserter data='ip-address']` - Visitor's IP address (**Pro only**, works only when server-side **Dynamic blocks** are used)
- `[adinsserter data='author']` - Post author username (works only inside posts when server-side **Dynamic blocks** are used)
- `[adinsserter data='author-name']` - Post author name (works only inside posts when server-side **Dynamic blocks** are used)
- `[adinsserter custom-field='CUSTOM_FIELD_NAME']` – Custom fields as [defined in posts](#)

The order of processing separator shortcodes is the following:

1. `[ADINSERTER COUNT]`
2. `[ADINSERTER ROTATE]`
3. `[ADINSERTER AMP]`

This means that you can have AMP separator inside ROTATE options and they can be used as COUNT options.

# AdSense Ads

---

AdSense is a free, simple way to make money online by placing ads on your website. [Google AdSense](#) is a program run by Google that allows publishers in the Google Network of content sites to serve automatic text, image, video, or interactive media advertisements, that are targeted to site content and audience.

To show AdSense ads you need to place code for AdSense on your website. Ad Inserter is a perfect tool for this.

To install Ad Inserter plugin log in to your WordPress dashboard, click on **Plugins** in the WordPress menu, click on **Add New** on top of the plugin page, search for **Ad Inserter**, select **Ad Inserter** from the results, click **Install** and then **Activate** the plugin.



The screenshot shows the WordPress 'Add Plugins' interface. At the top, there's a search bar with 'ad inserter' entered. Below the search bar, the 'Ad Inserter - WordPress Ads Management' plugin is featured. It includes a thumbnail image of a website with ads, the title 'Ad Inserter - WordPress Ads Management', a description 'Insert and manage ads: AdSense, Amazon, banners, ad rotation, sticky ad widgets, shortcodes, AMP, PHP,...', and the author 'By Igor Funa'. There are also buttons for 'Install Now' and 'More Details'. At the bottom, it shows a 5-star rating with 480 reviews, '100,000+ Active Installations', and 'Last Updated: 3 weeks ago'. A checkmark indicates it is 'Compatible with your version of WordPress'.

Here we'll explain step by step how to place AdSense ads on your website. If you don't have AdSense account yet you first need to [sign up](#).

## New to AdSense?

If you are new to AdSense you should first install the Ad Inserter plugin as described above and then jump to Connect your site. Check also **Get started with ad code**: <https://support.google.com/adsense/answer/181947>

## Ad unit

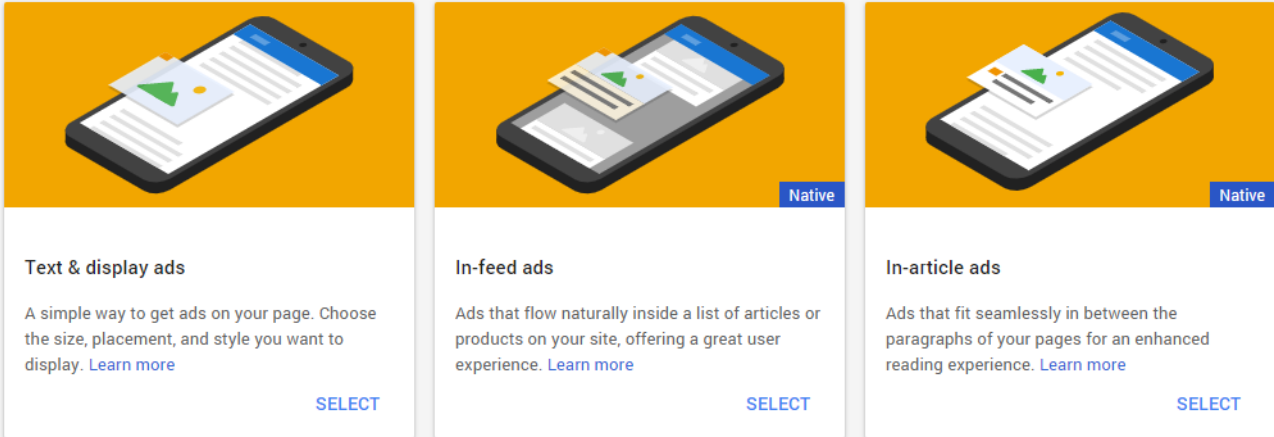
With ad units you can define type of ads for each ad space on your website. For each ad unit you get ad code which can be inserted where you would like to show ads. This way you have total control over the ads on your website.

**Ad code:** a snippet of HTML that when placed on any page of an approved site shows Google ads in a specific ad layout and format.

**Ad unit:** a set of Google ads displayed as a result of one piece of AdSense ad code.

The first step is to create ad unit. Once you are logged in to AdSense go to **My Ads / Content / Ad Units** and click on **New ad unit**. You will be asked to choose the type of ad.

What type of ad would you like to create?





Text & display ads	In-feed ads	In-article ads
<p>A simple way to get ads on your page. Choose the size, placement, and style you want to display. <a href="#">Learn more</a></p>	<p>Ads that flow naturally inside a list of articles or products on your site, offering a great user experience. <a href="#">Learn more</a></p>	<p>Ads that fit seamlessly in between the paragraphs of your pages for an enhanced reading experience. <a href="#">Learn more</a></p>
<a href="#">SELECT</a>	<a href="#">SELECT</a>	<a href="#">SELECT</a>


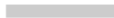










In most cases where you would like to place ads on your posts or static pages you'll need **Text & display ads**. Give your ad unit a name. It's a good idea to use a unique, descriptive name that will help you find this ad unit later.

## Text & display ads

In the "Ad size" section, choose the size of the ads that you'd like to show:

Ad size 

Showing **Recommended** 

Automatic size Responsive	728 x 90 Leaderboard	336 x 280 Large Rectangle	320 x 100 Large Mobile Banner	300 x 600 Large Skyscraper
 <a href="#">Preview</a>	 <a href="#">Preview</a>	 <a href="#">Preview</a>	 <a href="#">Preview</a>	 <a href="#">Preview</a>
Recommended:  	Recommended: 	Recommended: 	Recommended:  	Recommended: 

Here you can choose between **responsive** (automatic size) ad unit and fixed sizes. In general, wider ad units perform better. In most cases you can choose responsive ad unless you want a fixed size ad. In such case choose one of recommended ad sizes, which have shown to perform well across a variety of sites. Ad sizes that perform well on mobile are shown with a mobile icon.

## Responsive ads

This ad unit will automatically adapt its size to fit your page layout. It's intended to work with sites built with responsive design, but will work on non-responsive sites too. Responsive ad units allow you to support a wide range of devices by working with your responsive design web pages.

**IMPORTANT:** If you would like to use floating ads (ads wrapped with text) or sticky ads (ads that stay fixed when the page scrolls) you need to choose fixed size ads.

However, regardless of what type of Text & display ad you will choose here you will be able to change size or responsiveness in Ad Inserter code editor for AdSense.

Next you need to choose **Ad type**. We highly recommend that you display both text and display ads. By displaying more ad types, you can earn more revenue.

You can also choose text ad styles, define custom channels and what to display when no ad is available. When you are happy with settings click on Save and get code. Once the ad unit is created it will show in the list of ad units where you'll be able to edit it or get the code for it. Ad unit changes may take up to two hours to appear in performance reports.

**WARNING:** If you are using **ad blocker** make sure it is disabled when using Ad Inserter. If ad blocker is active you may experience issues and some functions may not work as described here.

## Ad code

When ad unit is created you can get ad code for it. In the list of ad units click on Get code link below ad unit for which you need to get code. A window will pop up. You can choose between **Asynchronous** and **Synchronous** code type. In most cases you should use **Asynchronous** code. It will look something like this:

```
<script async src="//pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
<!-- Ad Inserter 300x250 -->
<ins class="adsbygoogle"
      style="display:inline-block;width:300px;height:250px"
      data-ad-client="ca-pub-4504087291203902"
      data-ad-slot="3297227625"></ins>
<script>
(adsbygoogle = window.adsbygoogle || []).push({});
</script>
```

Once you have the code you can copy it to Ad Inserter. In this example we'll create one code block for ads above second paragraph on all posts. Select one empty code block (here we'll use code block 10), click on the name to change it to something descriptive, paste the code for ad unit into code window below and set Automatic insertion to **Before paragraph**, Paragraph number to **2** and leave checked only **Posts** page type.

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

**AdSense Post**

```

<script async src="//pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
<!-- Ad Inserter 300x250 -->
<ins class="adsbygoogle"
      style="display:inline-block;width:300px;height:250px"
      data-ad-client="ca-pub-4504087291203902"
      data-ad-slot="3297227625"></ins>
<script>
(adsbygoogle = window.adsbygoogle || []).push({});
</script>

```

Lists Manual Devices Misc Preview **Save All Settings**

Automatic Insertion:  Alignment and Style:  **Show**

Posts  Homepage  Category pages  
  Static pages  Search pages  Tag / Archive pages

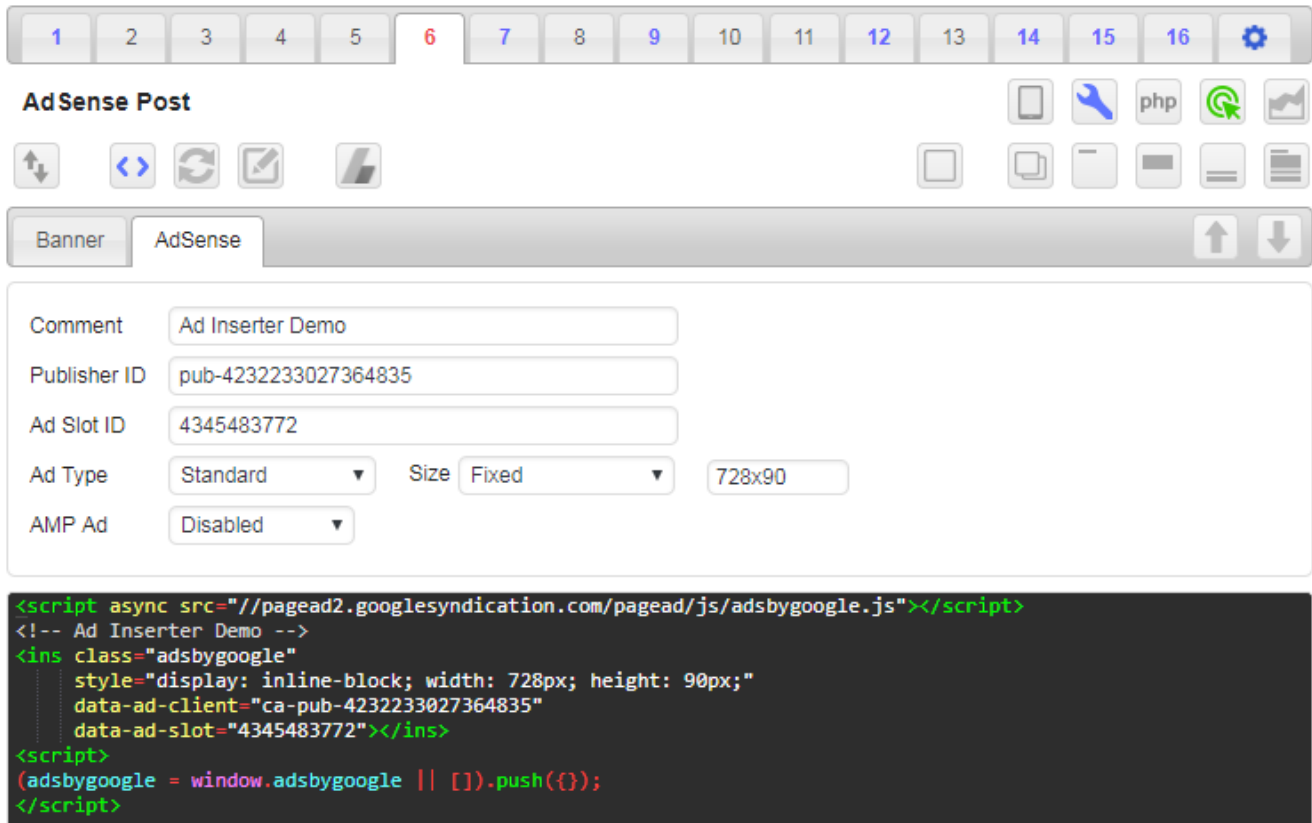
Paragraph(s)  **Counting** **Clearance**

Save settings and check posts. Your code should be inserted and ads displayed before second paragraph in each post. However, it may happen that you'll only get blank ad blocks. Check Ads not displayed – troubleshooting section below.



## AdSense code generator

Ad Inserter has also a code generator tool for AdSense. Check Code editing for details. This tool is mainly intended to change the type of AdSense ads. But please note: not all conversions are possible: you can convert standard text & display ads to link ads or vice versa, fixed width ads to responsive ads and vice versa and you can enable or disable ads on AMP pages. Once you have the **Publisher ID** and **Ad Slot ID** entered (or imported) you can change ad type, ad size (fixed, fixed by viewport or responsive) and ad type for AMP pages.



AdSense Post

Comment: Ad Inserter Demo

Publisher ID: pub-4232233027364835

Ad Slot ID: 4345483772

Ad Type: Standard | Size: Fixed | 728x90

AMP Ad: Disabled


```
<script async src="//pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
<!-- Ad Inserter Demo -->
<ins class="adsbygoogle"
  style="display: inline-block; width: 728px; height: 90px;"
  data-ad-client="ca-pub-4232233027364835"
  data-ad-slot="4345483772"></ins>
<script>
(adsbygoogle = window.adsbygoogle || []).push({});
</script>
```






Once you are happy with the settings click on the generate button and the the plugin will generate ad code for the AdSense ad in the code window of the current block.
















For standard and link ads you can choose between **fixed**, **fixed by viewport** and **responsive** ad sizes.

- **Responsive** means that the ad will resize to fill available width. Suitable for all devices, however, it will not work when available width is not known (floating ads). In such case you need to specify ad width with custom CSS style.
- **Fixed** means that you define one ad size and it will be used for all insertions of this code block.
- **Fixed by viewport** means that you can specify fixed ad size for each viewport (device or screen size as defined in general plugin settings tab ⚙ / tab Viewports). This is the best choice for AdSense ads when there is a need to use different ad sizes on different devices.

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 

**AdSense Post**     

Banner AdSense  

Comment  Desktop

Publisher ID  Tablet

Ad Slot ID  Phone

Ad Type  Size

AMP Ad

```

<style>
.ai-adsense-6 {display: inline-block; width: 336px; height: 280px;} /* Phone, default */
@media (min-width: 768px) { .ai-adsense-6 {width: 468px; height: 60px;} } /* Tablet */
@media (min-width: 980px) { .ai-adsense-6 {width: 728px; height: 90px;} } /* Desktop */
</style>
<script async src="//pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
<!-- Ad Inserter Demo -->
<ins class="adsbygoogle ai-adsense-6"
      data-ad-client="ca-pub-3222303405748352"
      data-ad-slot="2926129402"></ins>
<script>
(adsbygoogle = window.adsbygoogle || []).push({});
</script>

```

**WARNING:** If you are using **ad blocker** make sure it is disabled when using Ad Inserter. If ad blocker is active you may experience issues and some functions may not work as described here.

## AdSense Integration

**WARNING:** If you are using **ad blocker** make sure it is disabled when using Ad Inserter. If ad blocker is active you may experience issues and some functions may not work as described here.

Ad Inserter can be configured to get AdSense ad codes directly from the plugin settings page - no need to copy ad codes from the AdSense admin pages and paste them to code blocks. You can then list ad units, preview AdSense ads and get ad codes where you need them.

When access to your AdSense account is not enabled yet there will be **AdSense ad units** button in AdSense code generator.

The screenshot shows a configuration window with two tabs: 'Banner' and 'AdSense'. The 'AdSense' tab is active. Below the tabs are several input fields and controls: 'Publisher ID' (text input), 'Ad Slot ID' (text input), 'Ad Type' (dropdown menu set to 'Standard'), a 'Responsive' checkbox, 'Width' (text input) followed by 'px', 'Height' (text input) followed by 'px', and 'AMP Ad' (dropdown menu set to 'Disabled'). A button labeled 'AdSense ad units' is highlighted with a red rectangular border.

Click on this button to show **AdSense Integration - Step 1**.

**PLEASE NOTE:** In most cases AdSense integration will skip this step and start with Step 2 when only authorization code needs to be entered (described below). You will have to go through Step 1 only when using own Google API IDs. If you have decided to use own IDs and would like to go back using default IDs click on the button **Use default API IDs** and you'll be able to skip this step.

The screenshot shows a page titled 'Ad Inserter Pro AdSense Integration - Step 1'. It contains a list of 17 numbered steps for setting up a Google API project and enabling AdSense Management API. Below the steps are input fields for 'Client ID' and 'Client secret', and buttons for 'Use default API IDs' and 'Save'.

1. Go to [Google APIs and Services console](#)
2. Create **Ad Inserter** project - if the project and IDs are already created click on the **Credentials** in the sidebar and go to step 16
3. Click on **Select a project** selection and then click on the **+** button to create a new project
4. Enter **Ad Inserter** for project name and click on the **Create** button
5. Click on project selection, wait for the project to be created and then select **Ad Inserter** as the current project
6. Click on **ENABLE APIS AND SERVICES**
7. Search for adsense and enable **AdSense Management API**
8. Click on **Create credentials**
9. For **Where will you be calling the API from?** select **Other UI**
10. For **What data will you be accessing?** select **User data**
11. Click on **What credentials do I need?**
12. Create an OAuth 2.0 client ID: For **OAuth 2.0 client ID** name enter **Ad Inserter client**
13. Set up the OAuth 2.0 consent screen: For **Product name shown to users** enter **Ad Inserter**
14. Click on **Continue**
15. Click on **Done**
16. Click on **Ad Inserter client** to get **Client ID** and **Client secret**
17. Copy them to the appropriate fields below

Client ID   
Client secret

The first step is to create a Google API project in order to get **Client ID** and **Client Secret**. Follow the instructions to get Client ID and Client secret and enter them in the appropriate fields:

1. Go to <https://console.developers.google.com/>
2. Create **Ad Inserter** project - if the project and IDs are already created click on the **Credentials** in the sidebar and go to step 16
3. Click on **Select a project** selection and then click on the **+** button to create a new project
4. Enter **Ad Inserter** for project name and click on the **Create** button
5. Click on project selection, wait for the project to be created and then and select **Ad Inserter** as the current project
6. Click on **ENABLE APIS AND SERVICES**
7. Search for adsense and enable **AdSense Management API**
8. Click on **Create credentials**
9. For **Where will you be calling the API from?** Select **Other UI**
10. For **What data will you be accessing?** Select **User data**
11. Click on **What credentials do I need?**
12. Create an OAuth 2.0 client ID: For **OAuth 2.0 client ID** name enter **Ad Inserter client**
13. Set up the OAuth 2.0 consent screen: For **Product name shown to users** enter **Ad Inserter**
14. Click on **Continue**
15. Click on **Done**
16. Click on **Ad Inserter client** to get **Client ID** and **Client secret**
17. Copy them to the appropriate fields

Click **Save** to get to **AdSense Integration - Step 2**.

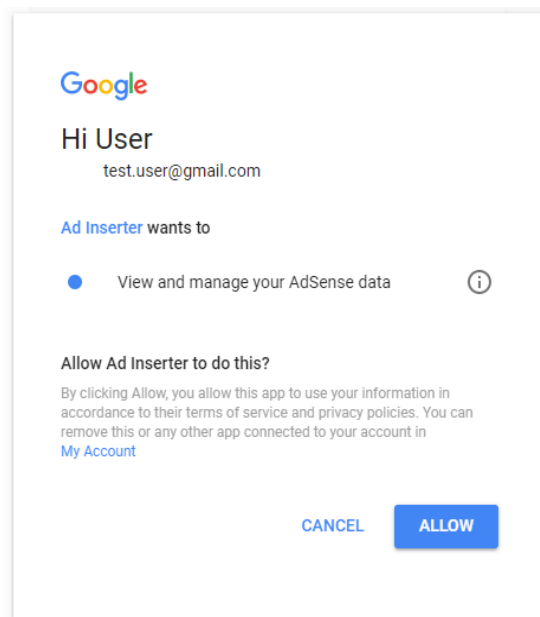
**PLEASE NOTE:** In most cases AdSense integration will start at this step. If you get an error, can't access ad units or would like to use own Google API IDs, click on the button **Use own API IDs** and you'll be redirected to the Step 1 described above.

### Ad Inserter Pro AdSense Integration

Authorize Ad Inserter Pro to access your AdSense account. Click on the **Get Authorization Code** button to open a new window where you can allow access. When you get the code copy it to the field below and click on the button **Authorize**.

If you get error, can't access ad units or would like to use own Google API IDs click on the button **Use own API IDs** to enter Client ID and Client Secret.

Now you need to authorize Ad Inserter plugin to access your AdSense account to get ad units and ad codes. Click on **Get Authorization Code** to open a new tab where you will be able to choose AdSense account and grant read-only access to the account.



When you'll click on **Allow** you'll get authorization code which should be pasted in the appropriate field. Then click on **Authorize** and you should get list of AdSense ad units. Above the list there is a search files where you can search for ad units. You can search using multiple keywords.

Ad unit	Name	Slot ID	Type	Size
<input type="checkbox"/>	Test 200x200	1823623876	Text, Image	200x200
<input type="checkbox"/>	Demo Guest	2026402745	Text, Image	336x280
<input type="checkbox"/>	Demo Sidebar	2037365485	Text, Image	Responsive

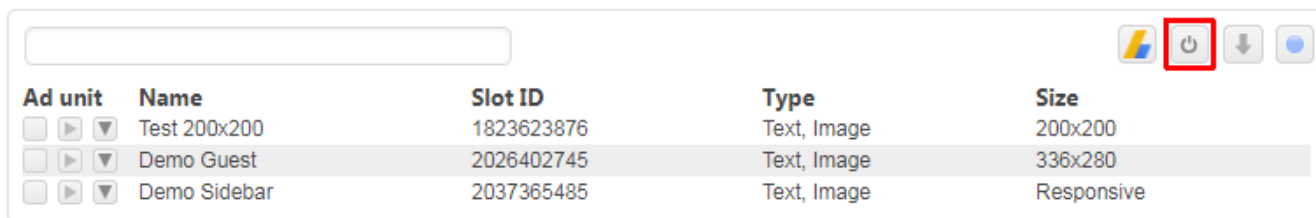
After you finish authorization process there will be additional AdSense button on the tools toolbar. Click on this button toggles list of AdSense ad units.



Click on the AdSense button will open AdSense admin pages.

Ad unit	Name	Slot ID	Type	Size
<input type="checkbox"/>	Test 200x200	1823623876	Text, Image	200x200
<input type="checkbox"/>	Demo Guest	2026402745	Text, Image	336x280
<input type="checkbox"/>	Demo Sidebar	2037365485	Text, Image	Responsive

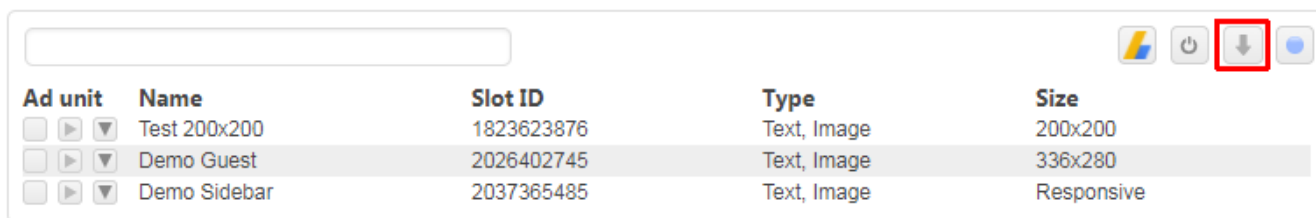
Click on **Clear authorization** button will remove access to your AdSense account.



The screenshot shows the AdSense ad units interface. At the top right, there are four icons: a blue lightning bolt, a power button (highlighted with a red box), a download arrow, and a blue circle. Below the icons is a table with the following columns: Ad unit, Name, Slot ID, Type, and Size. The table contains three rows of data.

Ad unit	Name	Slot ID	Type	Size
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Test 200x200	1823623876	Text, Image	200x200
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Demo Guest	2026402745	Text, Image	336x280
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Demo Sidebar	2037365485	Text, Image	Responsive

Click on **Reload** button will reload AdSense ad units. If you create a new ad unit or make changes to existing ads you'll need to reload the list to get access to the new ad units.



The screenshot shows the AdSense ad units interface. At the top right, there are four icons: a blue lightning bolt, a power button, a download arrow (highlighted with a red box), and a blue circle. Below the icons is a table with the following columns: Ad unit, Name, Slot ID, Type, and Size. The table contains three rows of data.

Ad unit	Name	Slot ID	Type	Size
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Test 200x200	1823623876	Text, Image	200x200
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Demo Guest	2026402745	Text, Image	336x280
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Demo Sidebar	2037365485	Text, Image	Responsive

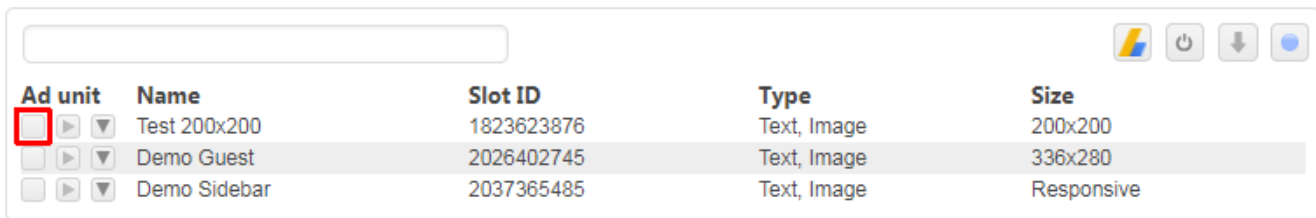
Click on **Active/All** ad units button will toggle between active ad units and all ad units.



The screenshot shows the AdSense ad units interface. At the top right, there are four icons: a blue lightning bolt, a power button, a download arrow, and a blue circle (highlighted with a red box). Below the icons is a table with the following columns: Ad unit, Name, Slot ID, Type, and Size. The table contains three rows of data.

Ad unit	Name	Slot ID	Type	Size
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Test 200x200	1823623876	Text, Image	200x200
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Demo Guest	2026402745	Text, Image	336x280
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Demo Sidebar	2037365485	Text, Image	Responsive

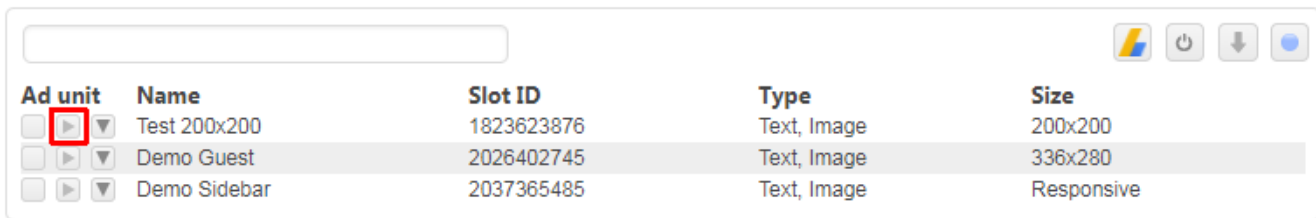
Each ad unit has three small buttons left of ad unit name. **Copy AdSense code** button copies the ad code to the Ad Inserter clipboard. You can then paste it to any code block.



The screenshot shows the AdSense ad units interface. At the top right, there are four icons: a blue lightning bolt, a power button, a download arrow, and a blue circle. Below the icons is a table with the following columns: Ad unit, Name, Slot ID, Type, and Size. The table contains three rows of data. The first row's 'Ad unit' column has three small buttons: a checkbox, a play button, and a dropdown arrow (the play button is highlighted with a red box).

Ad unit	Name	Slot ID	Type	Size
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Test 200x200	1823623876	Text, Image	200x200
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Demo Guest	2026402745	Text, Image	336x280
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Demo Sidebar	2037365485	Text, Image	Responsive

**Preview AdSense ad** opens a preview window and displays this ad unit between dummy paragraphs. Ad block may be blank – this is quite common when ad code has no ads to show.



The screenshot shows the AdSense ad units interface. At the top right, there are four icons: a blue lightning bolt, a power button, a download arrow, and a blue circle. Below the icons is a table with the following columns: Ad unit, Name, Slot ID, Type, and Size. The table contains three rows of data. The first row's 'Ad unit' column has three small buttons: a checkbox, a play button (highlighted with a red box), and a dropdown arrow.

Ad unit	Name	Slot ID	Type	Size
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Test 200x200	1823623876	Text, Image	200x200
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Demo Guest	2026402745	Text, Image	336x280
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Demo Sidebar	2037365485	Text, Image	Responsive

# Preview

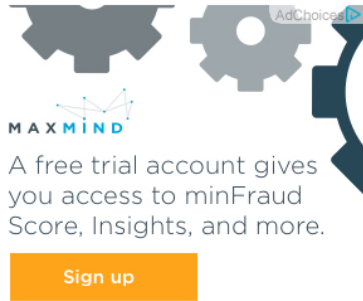
Highlight

## AdSense ad unit

Cancel

### Demo Guest

This is a preview of AdSense ad block between dummy paragraphs. AdSense ad code was loaded from your AdSense account. The ad block is displayed on a dummy page so it may be blank (no ads).



Ad Inserter can be configured to insert any code anywhere on the page. Each code with it's settings is called a code block. Free Ad Inserter supports 16 code blocks, Ad Inserter Pro supports up to 96 code blocks (depending on the license type). The settings page is divided into tabs - 16 code blocks and general plugin settings. Black number means inactive code block (code is not inserted anywhere), red number means block is using automatic insertion, blue number means block is using manual insertion while violet number means block is using automatic and manual insertion.

**Get AdSense code** button copies ad code to the currently selected code block. This is the quickest way to get AdSense ad code into code block.

Ad unit	Name	Slot ID	Type	Size
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Test 200x200	1823623876	Text, Image	200x200
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Demo Guest	2026402745	Text, Image	336x280
<input type="checkbox"/> <input type="play"/> <input type="dropdown"/>	Demo Sidebar	2037365485	Text, Image	Responsive

**WARNING:** If you are using **ad blocker** make sure it is disabled when using Ad Inserter. If ad blocker is active you may experience issues and some functions may not work as described here.

## In-feed ads

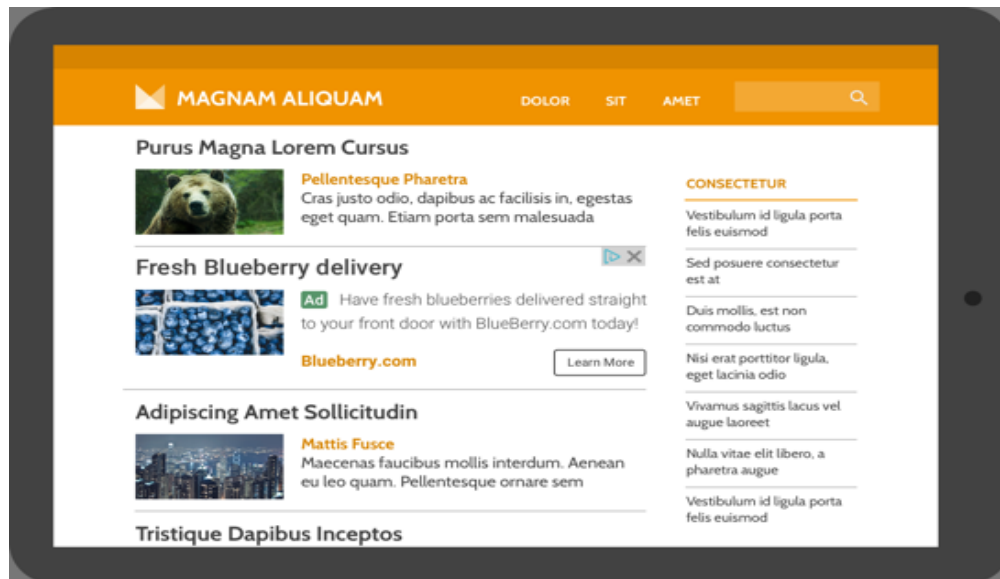
What is a feed?

A feed is a stream of content that you can scroll through. The content appears in similar-looking blocks that repeat one after the other. For example a feed can be editorial (e.g., a list of articles or news) or listings (e.g., a list of products, services, etc.). Feeds can appear anywhere on the page.

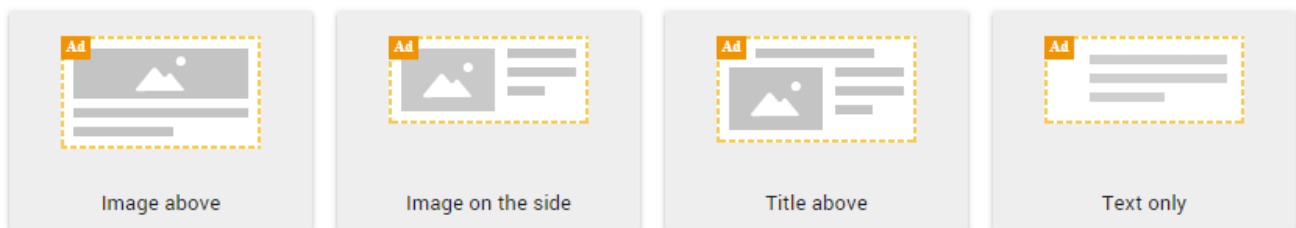
**Note:** For the purposes of In-feed ads, **a feed does not refer to a web syndication feed such as RSS.**

Some examples of feeds on WordPress pages include:

- The list of latest posts on a homepage (showing excerpts or full posts)
- The list of latest posts on a category page (showing excerpts or full posts)
- The list of latest posts on an archive page (showing excerpts or full posts)
- Comments below posts



Ad Inserter supports functions to insert in-feed ads. Generate Ad unit as described above – select in-feed ads. In most cases you should select Create ad style manually and choose a style that matches your WordPress theme.



Set additional options, save ad unit and get the code for it. Once you have the code for in-feed ads you can configure Ad Inserter.



In our case we will insert in-feed ads after 2<sup>nd</sup> and 4<sup>th</sup> paragraph on homepage. Set Automatic insertion to **Between posts**, check only **Homepage** page type and set Filter (button Misc) to 2, 4.

```
<script async src="//pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
<ins class="adsbygoogle"
  style="display:block"
  data-ad-format="fluid"
  data-ad-layout-key="- fg+5r+6l-ft+4e"
  data-ad-client="ca-pub-2934376232623522"
  data-ad-slot="2389237266"></ins>
<script>
  (adsbygoogle = window.adsbygoogle || []).push({});
</script>
```

Save settings and check homepage. Your code should be inserted and ads displayed between posts. However, it may happen that you'll only get blank ad blocks. Check Ads not displayed – troubleshooting section below.

**PLEASE NOTE:** Automatic insertion between posts works only if your theme is using homepage with standard WordPress hooks that are used by plugins to insert ads. If your theme uses custom homepage you may not be able to insert ads this way.

## In-article ads

In-article ads fit seamlessly in between the paragraphs of your pages for an enhanced reading experience. Use settings to insert ads between paragraphs as described for Text & display ads above.

uis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.



### The SunGoose Summer Collection

From mid-century modern to lounge, find the right fit for your home interior design style

[Sungoose.com](http://Sungoose.com)

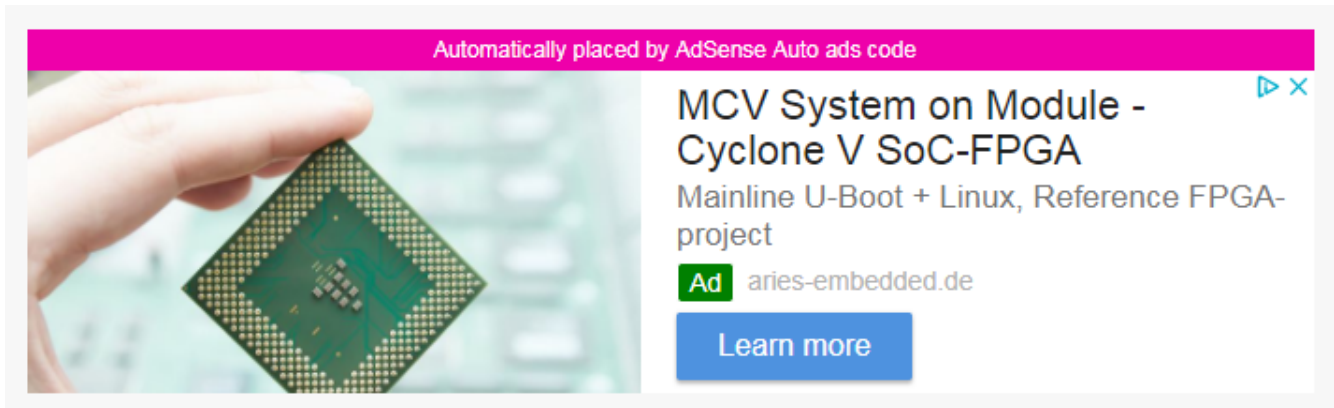
[Learn More](#)

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

## Auto ads




AdSense Auto ads (known also as Page Level Ads) is a code that you insert on every page once and it will **automatically display AdSense ads on positions chosen by the code**. Auto ads use Google's machine learning to pick the best placements on your pages to show ads. You only need to add one piece of code to all your pages, and let Google take care of the rest.

You can use Ad Inserter plugin to automatically insert code (usually in the `<head>` section - described below) for Auto ads on every page. However, **the exact placement of ads will be done by the ad code, not Ad Inserter!** When you enable debugging function **Label blocks**, AdSense Auto ads will be labeled with a colored bar on the top with a text informing you that the ad was automatically placed by AdSense Auto ads code.



**PLEASE NOTE:** Auto ads may not be suitable for some WordPress themes or layouts. In many cases it is easier to place ads at wanted locations by using Ad unit (normal AdSense code) and specifying exact locations where the code should be inserted.

To get code for Auto ads log in to your AdSense account and navigate to My Ads > Auto ads. Auto Ads can be enabled globally for all sites in your account or only for some of them using the group option. Once you choose Auto ads you choose which ad formats you'd like to show across your entire site. These will be your "global preferences" for Auto ads. You can select from a range of ad formats.


	<p><b>Text &amp; display ads</b></p> <p>A simple way to get banner ads on your page. Google will choose the size, placement, and style of ads you show.</p>	<input checked="" type="checkbox"/>
	<p><b>In-article ads</b></p> <p>Native ads that fit seamlessly in between the paragraphs of your pages for an enhanced reading experience.</p>	<input checked="" type="checkbox"/>
<b>Overlay ads</b> <span>?</span>		
	<p><b>Anchor ads</b></p> <p>Mobile ads that stick to the edge of the user's screen and are easily dismissible.</p>	<input checked="" type="checkbox"/>
	<p><b>Vignette ads</b></p> <p>Full-screen mobile ads that appear in between page loads on your site, also easily dismissible.</p>	<input checked="" type="checkbox"/>


Click on **Set up Auto ads** to get the code.

## Connect your site

You need to copy the code and paste it to Ad Inserter. The code will look like this (of course, publisher ID will be yours):

```
<script async src="//pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
<script>
  (adsbygoogle = window.adsbygoogle || []).push({
    google_ad_client: "ca-pub-3424290372640283",
    enable_page_level_ads: true
  });
</script>
```

This code needs to be inserted **once on every page** where you want to automatically show AdSense ads. The easiest way to do this is to insert the code in the `<head></head>` section – go to tab  / tab **Header** and paste the code there – make sure header code is enabled (green button top right). This will automatically insert code for Auto ads on all pages. Step-by-step instructions:

- Click on the last tab (gear icon  - general plugin settings)
- Click on the tab **Header** (for code in the `<head></head>` section)
- Paste AdSense code for Auto ads into the code window
- Click on the **Enable** button above code window to enable header code (green light must be on)
- Click on the **Save Settings** button to save settings

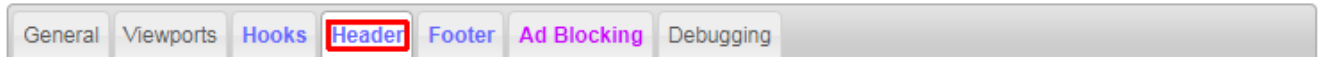


## Ad Inserter Settings 2.3.3

2018-02-28 12:30:59

Reset All Settings

Save Settings



### HTML Page Header Code



Code in the `<head></head>` section of the HTML page

```
<script async src="//pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
<script>
  (adsbygoogle = window.adsbygoogle || []).push({
    google_ad_client: "ca-pub-3424290372640283",
    enable_page_level_ads: true
  });
</script>
```

This code will now automatically display AdSense ads on positions chosen by the code. Auto ads use Google's machine learning to pick the best placements on your pages to show ads. You only need to add this piece of code to all your pages, and let Google take care of the rest.

After the code is inserted Google may need some time (usually hours) to check the pages before it starts to serve ads. If you believe you have followed the instructions above, the code is properly installed and you still don't see ads after some time (or the ad blocks are blank) check Ads not displayed – troubleshooting section for possible solutions. **Please note that the exact placement of ads will be done by the ad code, not Ad Inserter!**

If you don't want to show ads on every page you can either define custom groups for Auto ads or insert the code only on specific pages. In such case use any code block, set Automatic insertion to **Before Header** or **Before Post**, set Alignment and Style to **No Wrapping** and define rules where the code should be inserted. Check also IntroductionHeader and Footer code section for details on header code.

If you want to have control over the ad types and positions where the ads will be displayed, you can use code for AdSense Ad unit. For each ad unit you get ad code which can be inserted where you would like to show AdSense ads.

## Ads not displayed – troubleshooting

There are many reasons why you don't see ads. In order to help you to diagnose the problem Ad Inserter has many debugging functions that can help you to figure out why the ads are not where you expect them. Here we show example for AdSense ads but the same approach can be used also for other ads. This is how blank ad block looks like - empty space:

### WORDPRESS PLUGINS AND WEB DEVELOPMENT

[2017-08-02](#) | [admin](#) | [2 comments](#)

Why should I use WordPress? Why do I need to switch my previous site to this platform? Is my old CMS not good enough? What new does it offer to me? Often beginners ask all these questions. If you are also in the same boat and looking for the right path, so, you are at a right place. Here we are going to tell you that how WordPress is far better than others and what benefit it actually offers to you. It proposes you a plethora of themes, templates, and plugins to use, which make the work of a developer easier. To know about the benefits of its plugins in web development, scroll down and take a look below.

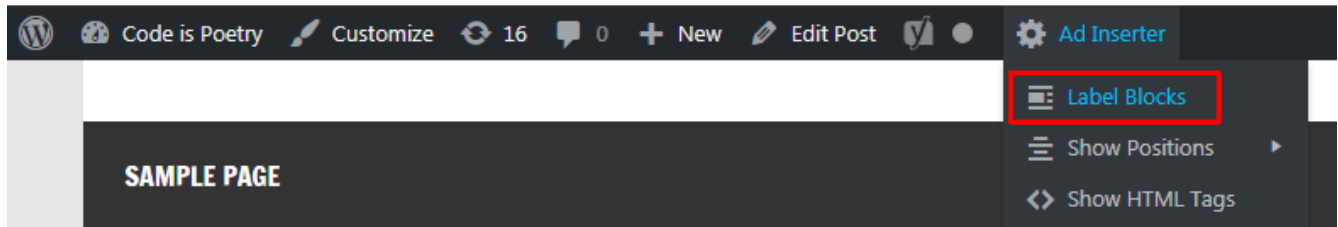


**Prevent Spam:** Needless to say, spamming will drag down the authority of your website and actually create a barrier in between your work. However, if you use WordPress for website development, so, it offers you plugins like WangGuard, Anti-Spam Bee, WP-SpamShield Anti-Spam and WordPress Zero Spam, etc., which help you to prevent spamming.

When you don't see AdSense ads there **two basic possible reasons**:

1. Code for AdSense ads is not inserted where you expect it
2. Code for AdSense ads is inserted but it is not displaying ads – blank ad block like on the screenshot above
3. You don't see ads because you are using ad blocker – disable it to see ads.

Let's first check whether the code is inserted where we expect it. When you are logged in as administrator you'll see admin toolbar above every post or page and there should be **Ad Inserter** menu item. Go to any post where the ads should be displayed and click on **Ad Inserter** / **Label Blocks** in admin toolbar.



In this example we'll check insertion of AdSense ad before the second paragraph. When function **Label Blocks** is enabled Ad Inserter will put a red frame around each code block so you will see it even if the ad code doesn't display anything. In our case we see that the code block 10 (AdSense Post) is inserted properly (before paragraph 2), however the code doesn't show anything. **Label Blocks** function will also mark AdSense ad blocks with green transparent overlay and ad block information: AdSense ad index, ad format, ad slot ID and if AdSense integration is enabled also ad unit name. This confirms two things:

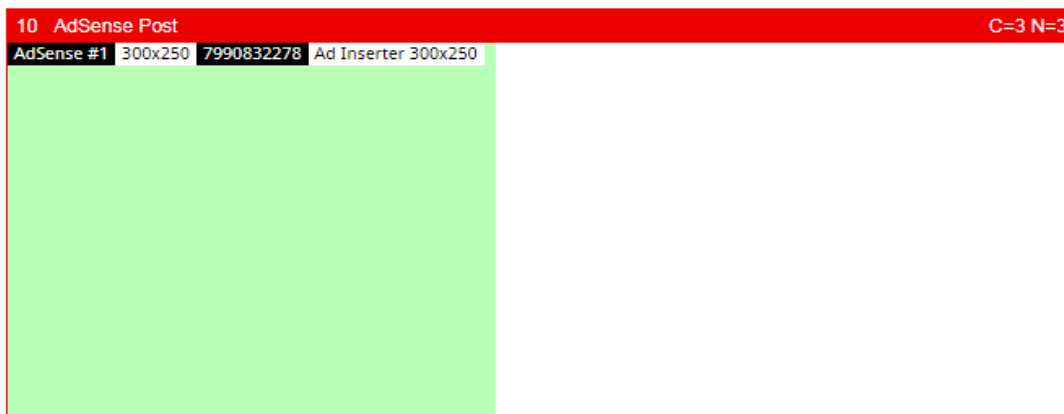
- Code block is properly inserted before paragraph 2 (red bar with red border around ad block)
- AdSense code is running - it has generated ad block (marked with green overlay)

## WORDPRESS PLUGINS AND WEB DEVELOPMENT

2017-08-02 | [admin](#) | [2 comments](#)

[Edit Post](#)

Why should I use WordPress? Why do I need to switch my previous site to this platform? Is my old CMS not good enough? What new does it offer to me? Often beginners ask all these questions. If you are also in the same boat and looking for the right path, so, you are at a right place. Here we are going to tell you that how WordPress is far better than others and what benefit it actually offers to you. It proposes you a plethora of themes, templates, and plugins to use, which make the work of a developer easier. To know about the benefits of its plugins in web development, scroll down and take a look below.



**Prevent Spam:** Needless to say, spamming will drag down the authority of your website and actually create a barrier in between your work. However, if you use WordPress for website development, so, it offers you plugins like WangGuard, Anti-Spam Bee, WP-SpamShield Anti-Spam and WordPress Zero Spam, etc., which help you to prevent spamming.



Blank AdSense ad block means that the code is inserted properly only the ad code doesn't display anything. Typical reasons for this are the following:

- After the ad code is inserted Google needs some time (hours) to check the pages before it starts to serve ads.
- AdSense account is not fully approved yet – if your account is new you need to wait until Google checks ad codes on your website and confirms your account. Usually this takes few days.
- Your website or page is not approved yet - you need to wait until Google web crawlers check your pages.
- Your website or page was not approved for AdSense ads
- Your AdSense account is banned
- No ads available for your website
- Wrong AdSense publisher ID or slot ID - check or regenerate ad code
- Errors in ad code – check or regenerate ad code
- You are using ad blocker

In case you don't see the red frame with code block number and name (as shown below) this means that the code is not inserted there (before the second paragraph) and you need to check settings. If you are unable to figure out the reason for not inserting the code block you can use debugging function **Log Processing** which will append the insertion log at the end of each page – switch to page source and at the end of page there should be a log as HTML comment showing settings used, the status of each code block and the reason why it is not inserted. Check Debugging for more.

## WORDPRESS PLUGINS AND WEB DEVELOPMENT

[2017-08-02](#) | [admin](#) | [2 comments](#)

[Edit Post](#)

Why should I use WordPress? Why do I need to switch my previous site to this platform? Is my old CMS not good enough? What new does it offer to me? Often beginners ask all these questions. If you are also in the same boat and looking for the right path, so, you are at a right place. Here we are going to tell you that how WordPress is far better than others and what benefit it actually offers to you. It proposes you a plethora of themes, templates, and plugins to use, which make the work of a developer easier. To know about the benefits of its plugins in web development, scroll down and take a look below.

**Prevent Spam:** Needless to say, spamming will drag down the authority of your website and actually create a barrier in between your work. However, if you use WordPress for website development, so, it offers you plugins like WangGuard, Anti-Spam Bee, WP-SpamShield Anti-Spam and WordPress Zero Spam, etc., which help you to prevent spamming.





When everything is in place you should see AdSense ad before the second paragraph:

## WORDPRESS PLUGINS AND WEB DEVELOPMENT

2017-08-02 | admin | 2 comments

[Edit Post](#)

Why should I use WordPress? Why do I need to switch my previous site to this platform? Is my old CMS not good enough? What new does it offer to me? Often beginners ask all these questions. If you are also in the same boat and looking for the right path, so, you are at a right place. Here we are going to tell you that how WordPress is far better than others and what benefit it actually offers to you. It proposes you a plethora of themes, templates, and plugins to use, which make the work of a developer easier. To know about the benefits of its plugins in web development, scroll down and take a look below.



**Prevent Spam:** Needless to say, spamming will drag down the authority of your website and actually create a barrier in between your work. However, if you use WordPress for website development, so, it offers you plugins like WangGuard, Anti-Spam Bee, WP-SpamShield Anti-Spam and WordPress Zero Spam, etc., which help you to prevent spamming.

**Please note:** you can use Ad Inserter plugin to automatically insert code (usually in the `<head>` section - as described above) for Auto ads on every page. However, **the exact placement of ads will be done by the ad code, not Ad Inserter!** When you enable debugging function **Label blocks**, AdSense Auto ads (placed by the ad code) will be labeled with a colored bar on the top with a text informing you that the ad was automatically placed by AdSense Auto ads code.

**WARNING:** If you are using **ad blocker** make sure it is disabled when using Ad Inserter. If ad blocker is active you may experience issues and some functions may not work as described here.

## Ad Impression and Click Tracking (Pro only)

---

Ad Inserter Pro supports ad impression and click tracking. This enables you to monitor how many times each ad was displayed and how many times it was clicked. In the case you are displaying two or more ad options (A/B testing) you can easily determine which ad option has the highest CTR (Click-Through Rate).

To use tracking you need to enable it globally (at the level of plugin), select click detection method (if available in your version), enable tracking for individual code blocks (ads) and then you have to clear page cache (if you are using caching). When enabled, Ad Inserter Pro adds tracking code to each page where ads are inserted.

**Please make sure you have cleared all the caches before making any tests otherwise tracking scripts will not be present on pages.** Tracking code uses also nonce verification to prevent unauthorized submission of tracking data. If the verification fails, Ad Inserter Pro treats the tracking submission as spam. This can occasionally cause false-positives when using cache plugins. When an old cache file remains for longer than 24 hours, nonce verification will fail, and you'll not be able to track impressions and clicks. To solve this problem, **make sure you have the appropriate garbage collection (clean-up) settings in the cache plugin.** The cache should be emptied frequently (less than 24 hours). The configuration varies in different cache plugins. Consult your cache plugin's support for details.

Let's first clarify some terms.

### Impressions

Impression is an event when code block (ad) is displayed. This is detected client-side in the browser with Javascript code after the page is loaded (assuming the Javascript is enabled in the browser). Therefore, impression tracking works also when the page is cached, but it doesn't work when Javascript is not enabled, Javascript is not running due to some error or the page was not loaded properly. For impression tracking to work, code block must have a wrapping div (**No wrapping** style can't be used).

### Clicks

Click is an event when somebody clicks on the code block (ad). This is detected client-side in the browser with Javascript code (assuming the Javascript is enabled in the browser). Therefore, click tracking works also when the page is cached, but it doesn't work when Javascript is not enabled, Javascript is not running due to some error or the page was not loaded properly. For click tracking to work, code block must have a wrapping div (**No wrapping** style can't be used).

However, ads come in various forms and not all forms support reliable click tracking. In general, there are two types of ads: simple banners with links and Javascript based ads. Javascript based ads (e.g. AdSense) create `<iframe>` HTML element which is isolated from the main page and doesn't support click detection. However, with some advanced code it is possible to quite reliably "guess" when the user has clicked inside iframe ad. Therefore, Ad Inserter Pro supports two methods for click detection.

### Standard Click Detection

This method can detect clicks on banners with links and is available in all versions of Ad Inserter Pro. An example of banner code where clicks can be reliably detected with standard click detection:

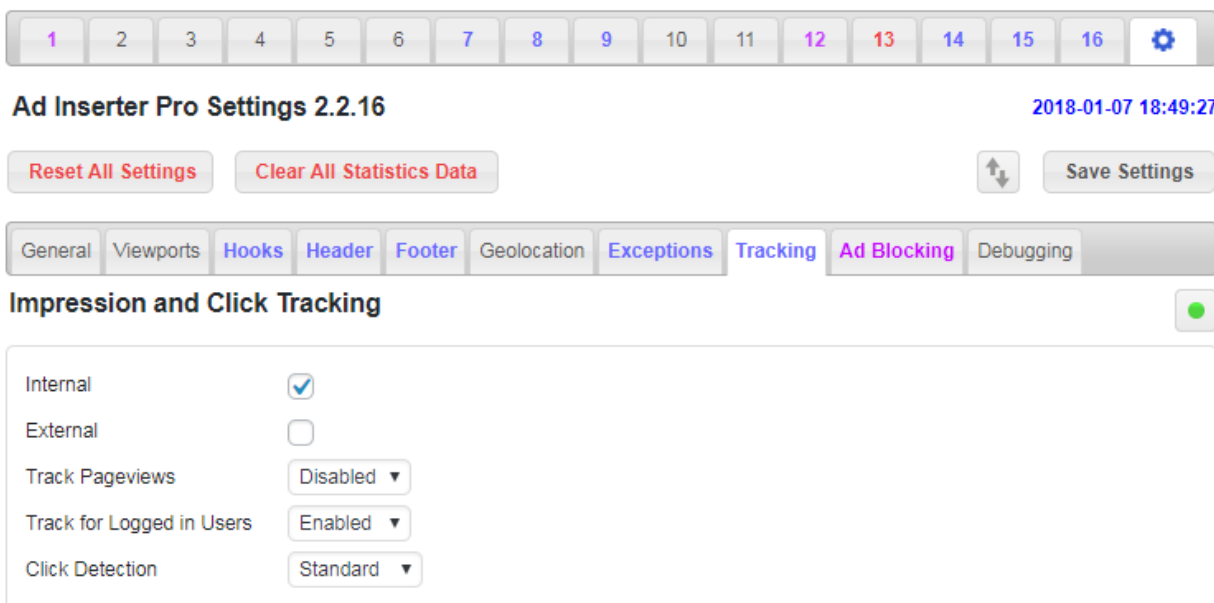
```
<a href="https://wordpress.org/plugins/ad-inserter/" target="_blank"></a>
```

## Advanced Click Detection

This method can detect clicks on any ad code but is not available in all versions of Ad Inserter Pro. Please note that impressions and clicks detected by Ad Inserter Pro might be different from statistics of ad networks. This is mainly because the way how impressions and clicks are counted differs for each ad network and ad networks may not count some events (e.g. invalid clicks for AdSense). However, the method is reliable enough to get CTR in A/B tests and determine which ad gets more clicks.

**PLEASE NOTE:** if you install additional click trackers Ad Inserter Pro click tracking may not work. **If you are using Ad Inserter Pro tracking make sure there are no other click trackers installed.** Tracking Settings

Tracking settings are available on the plugin settings tab (\*).



To globally enable tracking click on the button with green icon and save settings. There are two independent tracking modes: **Internal** and **External**.

- **Internal tracking** means using internal plugin statistics. Each event (impression or click) is sent to to the plugin and you can use internal statistics described below to analyze impressions, clicks, CTR, A/B tests. This may introduce additional load to your database on websites with high traffic.
- External tracking means using tracking via **Google Analytics** or **Piwik** events. Each event (**impression**, **click** or **ad blocking**) is sent to the external tracking service where you can analyze them. Make sure you have **Google Analytics** or **Piwik** tracking code installed. To analyze events go to your tracking service (Google Analytics or Piwik) admin pages and look for events with category **Ad Inserter Pro**.

You can also track all pageviews (with internal tracking) and ad blocking statistics by device (internal and external tracking are supported). Check below for details. If you would like to test tracking and check statistics while you are logged in set **Track Logged In Users** to **Enabled**. Choose (if available) **Click Detection** method according to the explanation above:

- **Standard** click detection can **detect clicks only on banners with links** and is available in all versions of Ad Inserter Pro.
- **Advanced** click detection can **detect clicks on any ad code** (including Javascript `<iframe>` ads) but is not available in all versions of Ad Inserter Pro.

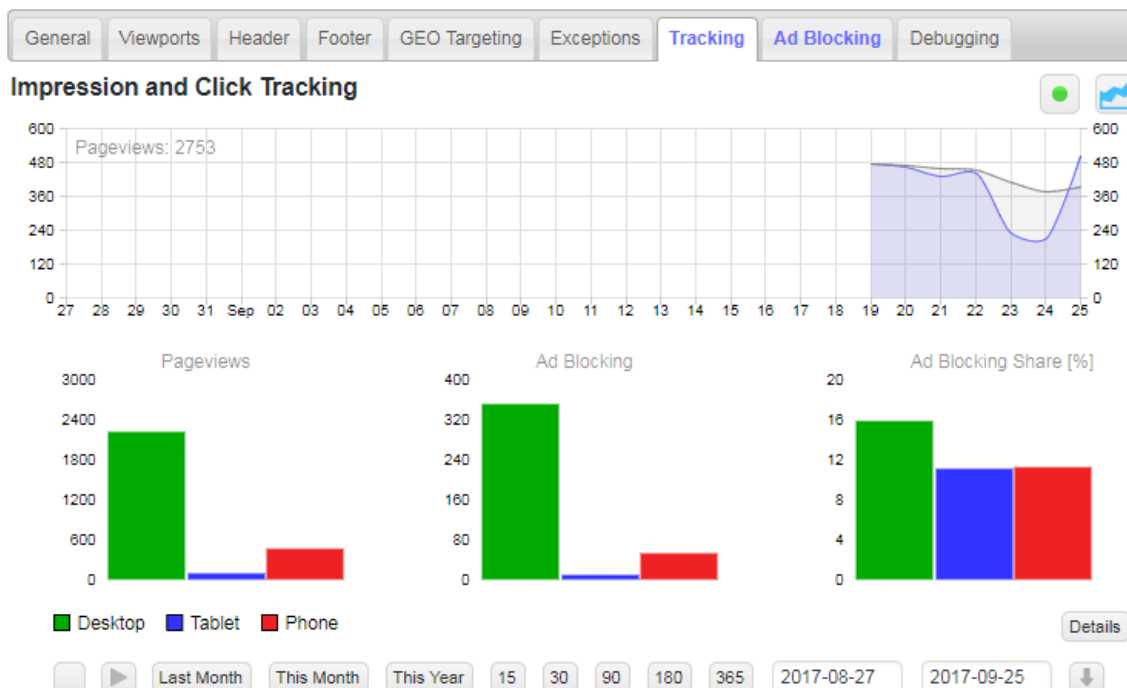
If you are using only banners then choose Standard click detection as it is the most reliable method to detect clicks. Use Advanced method only if you need to detect clicks on AdSense, Amazon, Media Net or other Javascript based ads, for example, to run A/B test with AdSense and Amazon Native Shopping Ads to determine which ad has the highest CTR.

Clear All Statistics Data clears all impression and click data for all code blocks. Use it only when you need to delete all statistics.

## Pageview Tracking

When enabled, Ad Inserter tracks each pageview (page load). To see the (internal) statistics click on the Statistics button.

Statistics show pageviews by device and, if ad blocking detection is enabled, also events where ad blocking was detected and ad blocking share for each device. Devices are detected based on Viewports settings.



## Ad Tracking

To track individual code block (or ad) you need to enable tracking for this code block. Click on the tracking button and save settings.



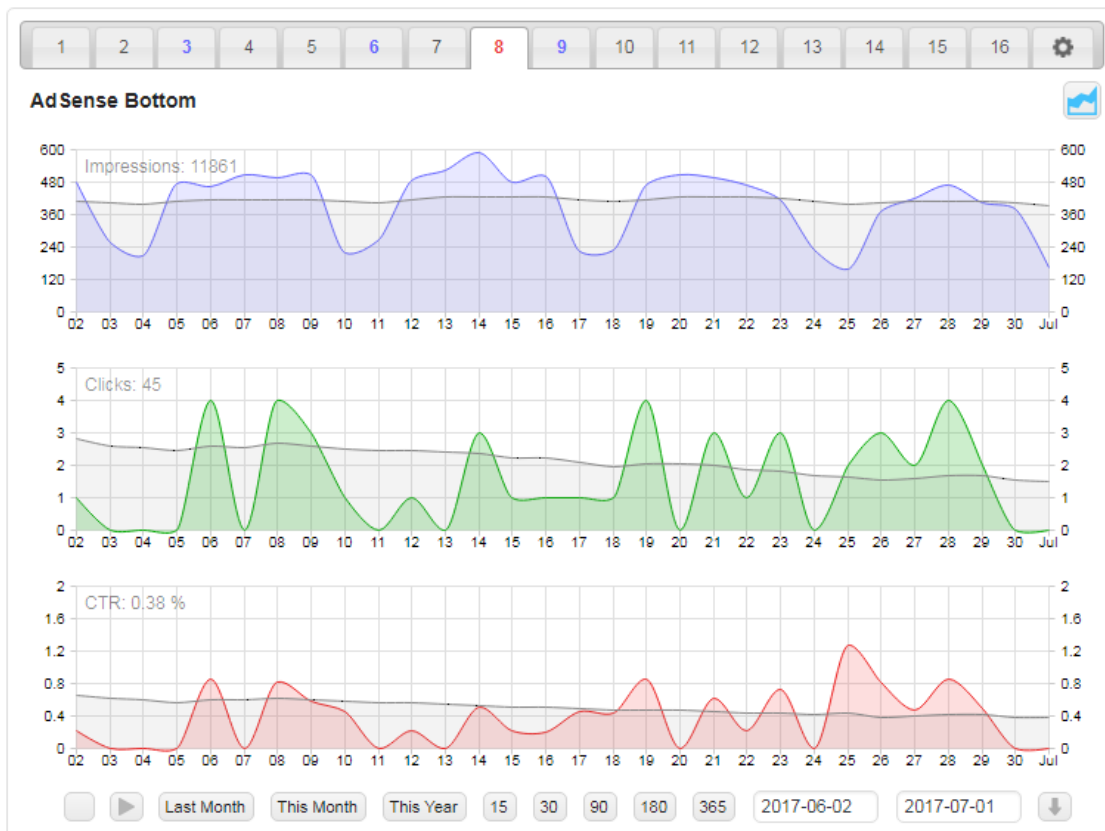
Once the tracking for the code block is enabled, Ad Inserter Pro will insert tracking code on the page where this ad is inserted and will track impressions and clicks on this ad. If you are using caching you need to clear cache before tracking can be used.

## Internal Tracking Statistics

To see impression and click statistics for this block click on the statistics button.



This button toggles code editor and statistics display.



Tracking statistics shows impressions, clicks and CTR for the selected range including average value for the last 30 days. You can select one of the predefined ranges (last month, this month, this year, last 15, last 30, last 90 last 180 or last 365 days) or select custom date range. The statistics data is kept for one year. When you select a predefined range the data is loaded automatically, for custom data range you need to click on the load button to load statistics data for this range.



Sometimes you might get error message when trying to load data. This is usually due to expired security token (the page has been loaded for a while). In such case simply reload Ad Inserter Pro settings page.

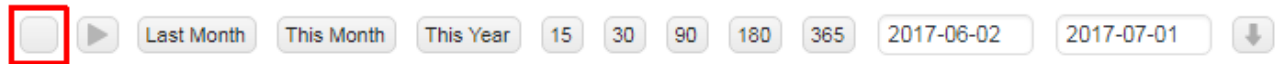


Ad Inserter Pro supports also live tracking - you can watch in real time how impressions, clicks and CTR change when visitors visit your website. For this you need to enable Auto Refresh function.



When this function is enabled Ad Inserter Pro will reload data every 60 seconds.

Sometimes you may want to clear statistics data for some date range. To do this select wanted date range and click on the clear button. All the statistics data for this block will be deleted and the charts will show no data for this date range.

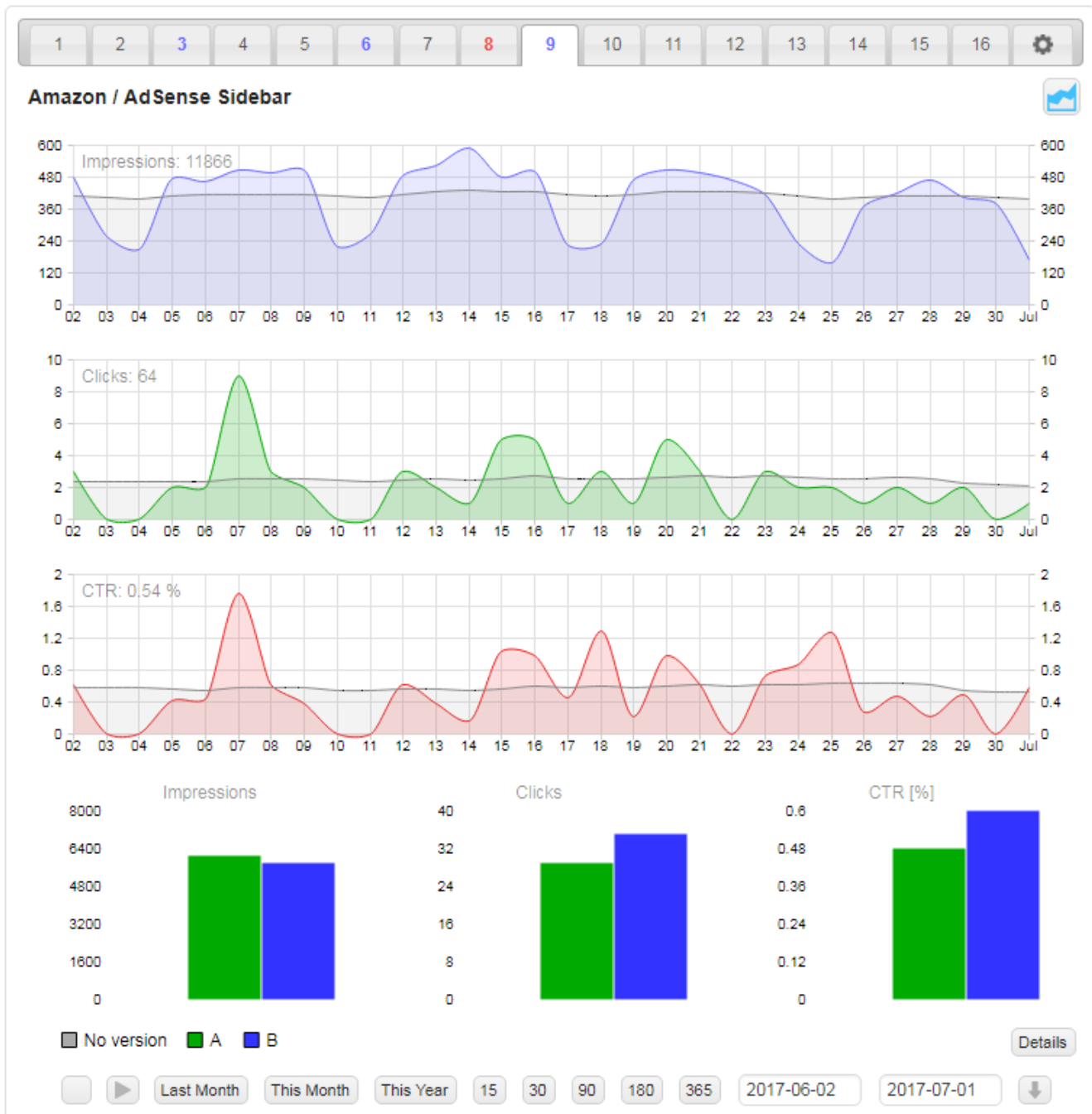


When tracking is globally or just for this block disabled, you may still analyze stored statistics data. In such case there will be a small yellow sign indicating disabled tracking.



## A/B testing

When code block (or ad) has two or more versions (using `[ADINSERTER ROTATE]` function), tracking statistics will show total data (impressions, clicks and CTR) and data for each version. This way you can easily determine which version gets more clicks.

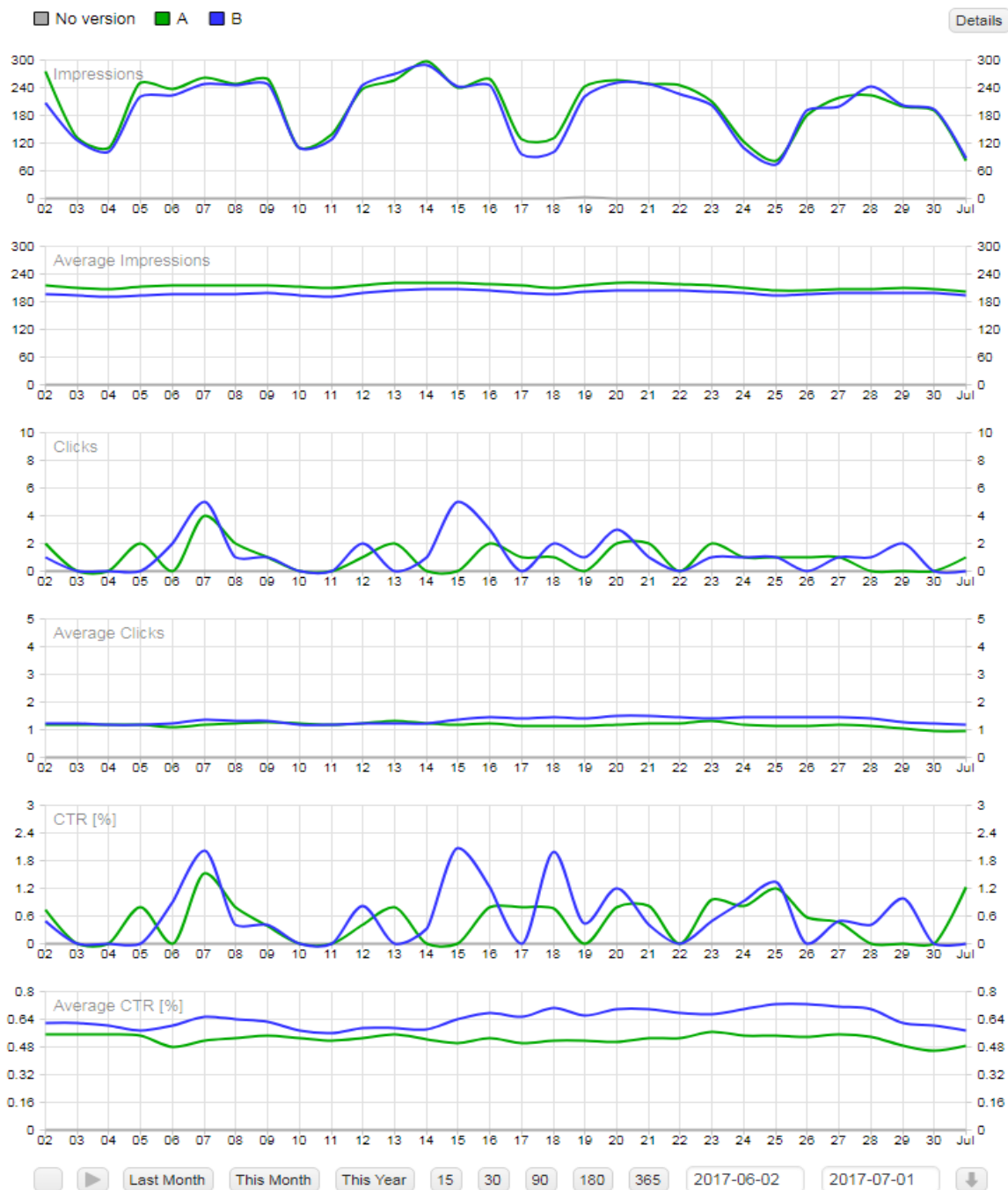


Please note that there might be some slight imbalance between version impressions. The `[ADINSERTER ROTATE]` function randomly selects one ad on each page load so on long run you'll get almost equal number of

impressions for each ad version. However, the CTR will always be accurate as it is the ratio of actual clicks and impressions.

No version label simply means that there are also impressions or clicks where there was only one ad version or version was not detected (due to some error on the page). If you have defined names for the options they will be used in the legend. If you have defined names for the rotation options (in the rotation editor or directly in the rotation separators) they will be shown here in the legend.

You can also get impression, click and CTR charts for individual ad versions. Click on Details button to toggle version charts.



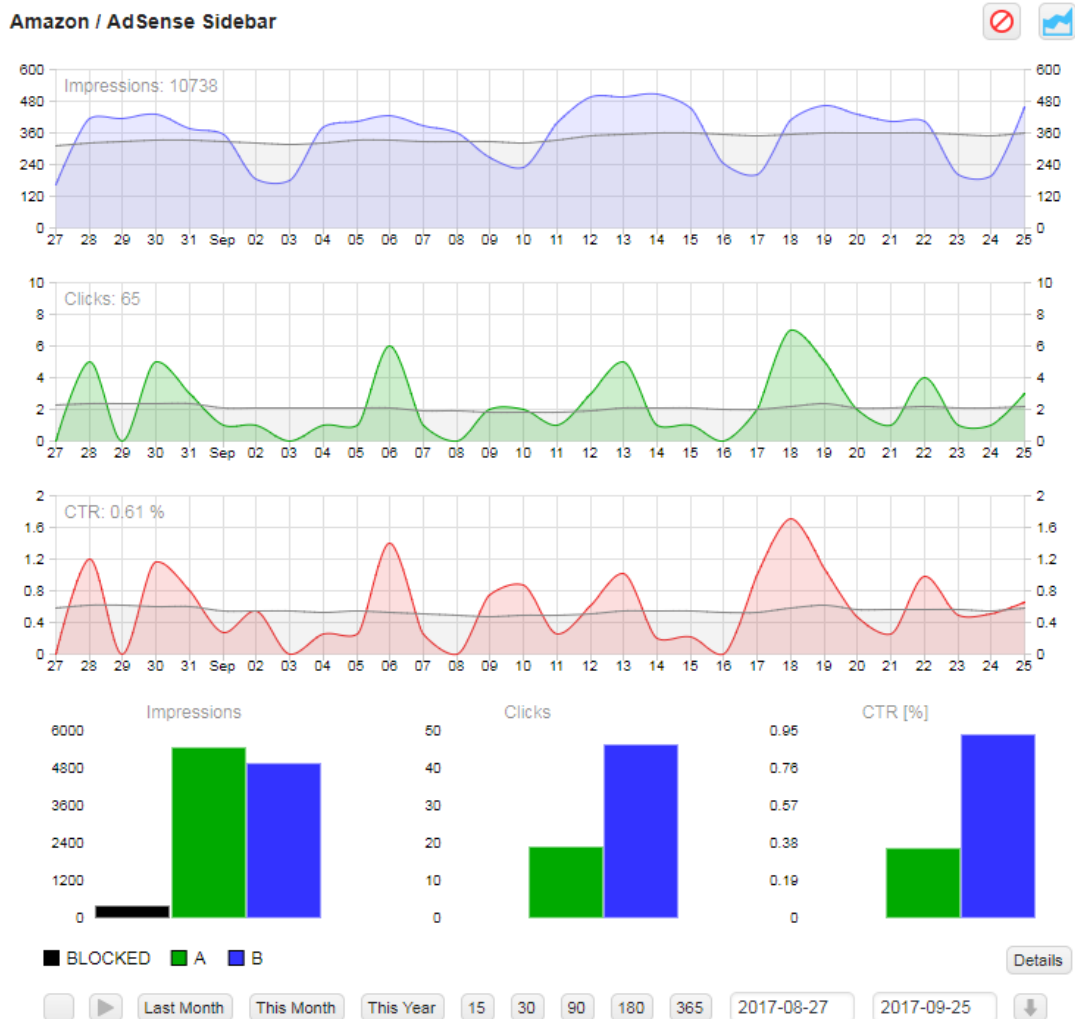


## Ad Blocking Statistics

Ad Inserter supports Ad Blocking Detection. When enabled, it tracks also ad blocking events. To see ad blocking statistics for individual code block click on the **Ad Blocking Statistics** button.



- When ad blocking detection is not enabled (tab ⚙ / tab Ad Blocking) all impressions are normally counted regardless whether the ad was blocked or not.
- When ad blocking detection is enabled, default statistics show only impressions where the ad was not blocked. **To count ad as blocked, ad blocking must be detected and ad height must be 0.**
- When ad blocking detection is enabled, statistics with ad blocking (**Ad Blocking Statistics** enabled) show all impressions and share of ad blocking events. In the case of ad rotation, ad blocking events are represented as one version (blocked impression).



# Ad Blocking Detection

---

Ad blocking or ad filtering is a type of software, that can remove or alter advertising content from a website - banners, text ads, pop-ups, tracking code, malware, etc. Ad blockers are available for a range of computer platforms, including desktop and laptop computers, tablet computers and smartphones. A variety of methods have been used for blocking advertisements. The benefits of ad blocking software as seen from the visitor's perspective are mainly reduced page loading time and ad-free web pages. Some use ad blockers also to prevent tracking scripts to collect data.

The use of ad blocking software is increasing. However, many media owners (publishers) rely on advertising to fund the free content that they provide to users. Some have already taken counter-measures against users who block ads on the sites they visit. More and more websites are asking (or sometimes requiring) visitors to turn off ad blockers before allowing them to view the site's contents.

In most cases it is possible to detect if the visitor is using ad blocking software. However, it is not possible to force to show the ads that were blocked. But if you know that the user is using ad blocking you can do few things: you can ask him to turn it off, you can redirect the user to some page explaining the problem, you can protect the content if ads are not shown and you can replace normal ads that were blocked with some replacement ads that will not be blocked. Ad Inserter Pro supports all of this.

Ad Blocking detected – possible actions with Ad Inserter:

1. Show popup message – user can close the message and continue reading
2. Show popup message – user can't close the message – navigation not possible
3. Redirect to some page
4. Replace ads with some other content or different ads (Pro only)
5. Protect (hide or modify) the content (Pro only)

## Ad Blocking Statistics (Pro only)


Ad Inserter Pro supports also ad blocking statistics. Check Ad Impression and Click Tracking (Pro only) for details.

## Ad Blocking Detection Settings

The settings for ad blocking detection are divided into 3 parts:

1. ad blocking detection
2. general actions when ad blocking is detected
3. actions for individual code blocks and content (Pro only)

Ad blocking detection is a code that is inserted when ad blocking detection is enabled. It checks various scripts and insertion of dummy ads and determines if ad blocking is present or not. Of course, this detection is not 100% reliable but it is very robust and hard to trick. There will always be some way for users to trick websites they are not using ad blockers – this is an endless game. However, in most cases it will be possible to detect if ads were blocked or not.

Ad blocking detection needs to be enabled for any ad blocking action to work. Ad blocking detection settings are available on the tab  / tab **Ad Blocking**.

To enable Ad blocking detection click on the right button to turn it green.

**PLEASE NOTE:** Ad blocking detection needs to be enabled even if you will not use general actions available here (popup message or page redirection) but you will only replace ads or protect content.

Ad blocking detection has some additional parameters that you may need. **Action** here is a general action that is fired immediately when ad blocking is detected. Possible actions are **None**, **Popup message** and **Page redirection**. When you define action here it will be enabled for all pages. To enable action only on specific pages set **Action** to **None** and insert shortcode `[adinsserter adb="message"]` or `[adinsserter adb="redirection"]` anywhere on pages where you want ad blocking action. The shortcode will generate no output so you can insert it at any position. This shortcode will only generate ad blocking action code in the footer. You can insert it manually or use any code block to insert it automatically on wanted pages (use **No wrapping** style).

When the user using ad blocker comes to your website you can delay action until few pages are visited. **Delay action** is the number of pages that the visitor must visit before the action triggers. Leave empty for immediate action.

When the action is triggered and the visitor for example closes the popup message the action will not trigger again during the **No Action Period**. This is a period in days (saved in cookie) when the user can browse the website without any warning or redirection. Leave empty to always trigger action every time the page is visited.

**PLEASE NOTE:** **Delay action** and **No Action Period** affect only actions set here. Ad replacement and content protection are not affected by these settings.

In order to improve or customize ad blocking detection Ad Inserter supports custom CSS selectors that are checked in order to detect ad blocking. This is an advanced feature and must be used with care – you need to know some HTML and CSS basics in order to use it.

CSS selector can be any class name or id that can identify your ad that may be blocked. Blocking is detected by checking ad height. If it is 0 it means that ad blocking prevented the ad from showing. For example, you have the following ad:

```
<div id="header-ad">  
<a href="http://example.com/"></a>  
</div>
```

To use this banner for ad blocking detection use `#header-ad` for custom CSS selector.

Ad Inserter Pro has additional ad blocking detection algorithms. In case there are problems with ad blocking detection you can switch to **Standard** mode.

## Popup Message

Popup message is a typical action when you want to inform visitor that ad blocking was detected and he needs to do something. Message is normally displayed over semi-transparent overlay which covers the page. You can configure the message text (including HTML), message CSS and overlay CSS.

Message HTML code



```
<p><strong>Blocked because of Ad Blocker</strong></p>
<p>It seems that you are using some ad blocking software which is preventing the page from fully loading.
Please whitelist this website or disable ad blocking software.</p>
```

Message CSS	<input type="text" value="width: 300px; padding: 10px; border: 5px solid #f00; border-radius: 5px;"/>
Overlay CSS	<input type="text" value="background: #000; opacity: 0.85;"/>
Undismissible Message	<input type="checkbox"/>

Check **Undismissible message** to prevent the visitor from closing the popup window.

**WARNING:** This is very annoying and should be used with care! You are preventing the user from visiting your website!



## Redirection

When ad blocking is detected you can redirect visitor to any existing static page or custom url. Custom url can be absolute or relative.

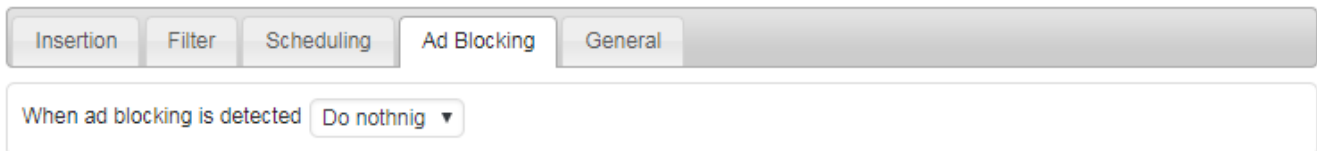
Redirection Page	Custom Url ▼
Custom Redirection Url	<input type="text"/>

## Ad Replacement (Pro Only)

In addition to popup messages and page redirection you can replace ads that were blocked with some that may not be blocked. You can also hide or show some ads (or whatever is in the code block) when ad blocking is detected. Each code block has a setting for this – button **Misc** / tab **Ad Blocking**.

**PLEASE NOTE:** Ad Blocking detection needs to be enabled for ad replacement to work (tab ⚙️ / tab **Ad Blocking**). You also can't use **No Wrapping** style as wrapping div has to be used in order to mark code blocks which need to be replaced/shown/hidden.

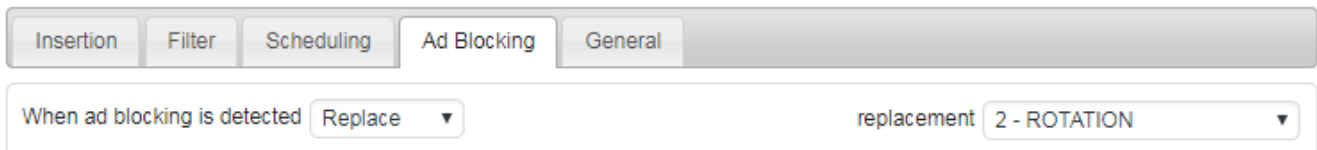
### Do nothing



The screenshot shows a settings bar with tabs: Insertion, Filter, Scheduling, Ad Blocking, and General. The 'Ad Blocking' tab is selected. Below the tabs, a dropdown menu is set to 'Do nothing'.

This is the default setting for code block. When ad blocking is detected it does nothing.

### Replace

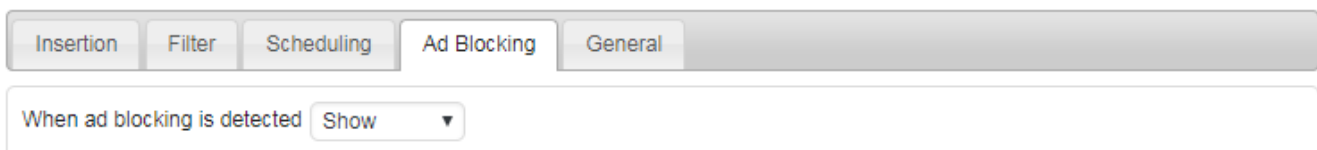


The screenshot shows the same settings bar as above. The 'Ad Blocking' tab is selected. The dropdown menu is set to 'Replace', and a secondary dropdown menu to its right is set to '2 - ROTATION'.

When ad blocking is detected the code block will be replaced with replacement code block (ad). Replacing means that the replacement block is normally inserted but hidden (using CSS). When ad blocking is detected the original ad is hidden (regardless whether it was blocked or not) and the replacement ad is shown.

Choose replacement ads wisely. Do not name images with standard names (ad, ads, banner) or with ad dimensions (300x250). Use some neutral name that will not alert ad blockers.

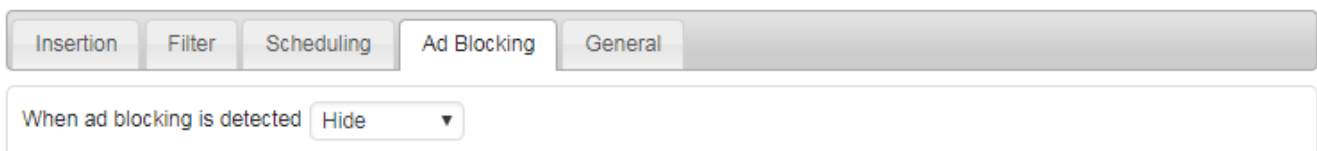
### Show



The screenshot shows the same settings bar. The 'Ad Blocking' tab is selected. The dropdown menu is set to 'Show'.

The block is normally inserted but hidden (using CSS). When ad blocking is detected the ad (or any code) is shown.

### Hide



The screenshot shows the same settings bar. The 'Ad Blocking' tab is selected. The dropdown menu is set to 'Hide'.

The block is normally inserted and visible. When ad blocking is detected the ad (or any code) is hidden.

## Content protection (Pro only)

Ad Inserter Pro supports also few methods to protect content when ad blocking is detected. This way you can hide, delete or change (via Javascript and CSS) some parts of the content that you don't want to show to visitors using ad blocking. Content protection works by inserting special shortcodes to mark content block that needs to be protected.

Normally you would need two code blocks: one to mark beginning and one to mark end of protected content. For example, if you would like to protect text in posts except the first and last paragraph you would insert shortcode to mark the beginning after the first paragraph and shortcode to mark the end before the last paragraph.

Use code block insertion options as for inserting any other code. The only difference is that you should use **No Wrapping** style so the codes are inserted without any wrapping code. You can also use only shortcode to mark the beginning of protected block as protection will work until the end of the parent container - inside posts this usually means until the end of the post. Of course, you don't need to use automatic insertion. Once the code block is configured you can also manually insert shortcodes where needed.

Protection works by Javascript code which once ad blocking is detected looks for HTML "markers" (generated by shortcodes) and then applying requested protection until the end marker is found or there are no more elements inside container (usually a div). This may sound complicated but it is easy to use and very powerful tool to protect content.

### Hide Content

Hiding works by applying CSS `display: none;`. The content is still present on the page but it is not visible. Any advanced user could open web inspector in browser and change visibility back. However, for most visitors this will look like the content is missing or protected. Shortcodes:

```
[ADINSERTER adb='hide']  
[ADINSERTER adb='hide-end']
```

### Delete Content

To the visitor this looks the same as hiding. The difference is that here is the content actually deleted from the page DOM. Therefore, it is not possible to open web inspector in browser and get the content back. Shortcodes:

```
[ADINSERTER adb='delete']  
[ADINSERTER adb='delete-end']
```

### Change Style

You can also change the style of the protected content. The content is still present on the page but it is modified to decrease readability or to show it is protected. Shortcodes:

```
[ADINSERTER adb='css' css='ANY CSS CODE']  
[ADINSERTER adb='css-end']
```

For example, to change the color of the protected text to some light gray you would use

```
[ADINSERTER adb='css' css='color: #ddd;']
```



## Replace Text

You can also change text of the protected content. The page still looks similarly as with the original text, however, when the visitor comes to the protected part he sees that there is something wrong. Ad Inserter will try to replace the original content with new text taking into account the length of the paragraph so the content layout will look like with the original text. Shortcodes:

```
[ADINSERTER adb='replace' text='ANY TEXT']  
[ADINSERTER adb='replace-end']
```

Of course, you can use text replacement and style change. For example:

```
[ADINSERTER adb='css' css='color: #ddd;']  
[ADINSERTER adb='replace' text='You are using ad blocker. ']
```

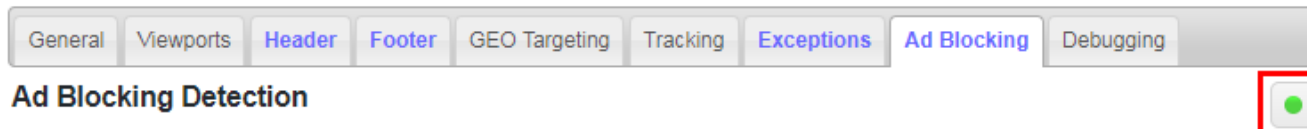
## Content Selectors

To have more control over what is protected, Ad Inserter Pro supports to define comma separated list of CSS selectors in shortcodes. When no selector is specified a default selector `p` is used as most posts use `p` paragraphs. This is useful when your content that needs to be protected uses different elements (`p`, `div`, `ul`, `ol`, `li`, `blockquote`, `figure`, etc.). With selectors you can have a detailed control over what will be protected and how. For example, if posts have also `<li>` elements in lists `<ul>` you would use shortcode

```
[ADINSERTER adb='css' css='color: #ddd;' selectors='p, li']
```

## Troubleshooting Content Protection

1. Make sure ad blocking is enabled



2. Make sure code blocks to mark protected block (where shortcodes are placed) have **No Wrapping** style.
3. Use debugging functions to simulate and test ad blocking.

## Debugging Ad Blocking Detection

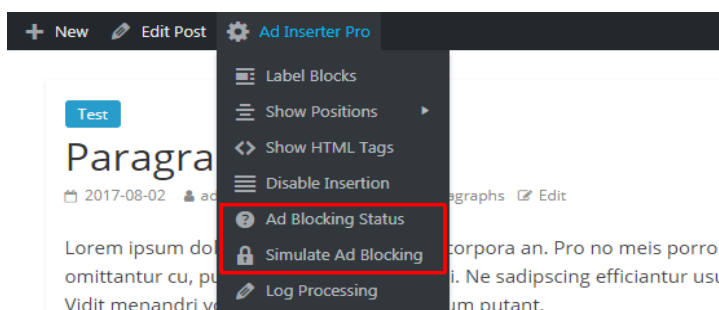
Ad blocking detection works by checking if all scripts were properly loaded and no elements were blocked. For this to work properly Ad Inserter needs working Javascript with no errors and the scripts used by the plugin need to be present on the page exactly where they were inserted by the plugin. Some caching plugins may include those scripts as inline scripts and some plugins may "optimize" JavaScript code and consequently modify the order in which scripts used for ad blocking detection are loaded. In such case the plugin will assume ad blocking is present - false ad blocking detection. The easiest way to diagnose the problem is to disable caching and optimizing plugins that may change page source. Once ad blocking works properly enable plugins one by one to see which causes problems. Take into account also external caching and Javascript code optimizations with services like CloudFlare.

If you see that ad blocking detection is triggered even when no ad blocker is active check the following:

- There are no Javascript errors on the page - check browser's console for any messages or errors
- All the scripts used by the plugin are present in the original order - check ad blocking detection with disabled caching and optimizing plugins. You may need to exclude those scripts from optimization:

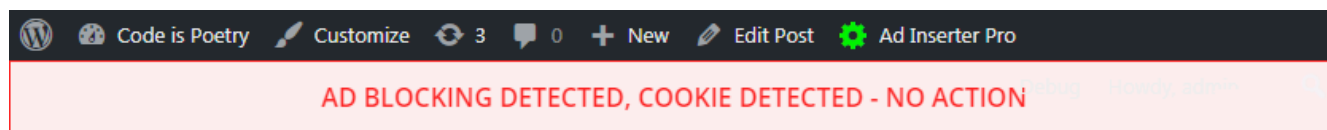
```
<script type='text/javascript' src='http://example.com/wp-content/plugins/ad-inserter/js/ads.js?ver=7297366'></script>
<script type='text/javascript' src='http://example.com/wp-content/plugins/ad-inserter/js/sponsors.js?ver=2310751'></script>
```

In order to test ad blocking functions without actually using ad blocking software Ad Inserter Pro supports two debugging functions available from the Ad Inserter toolbar on the top of the pages – available when you are logged in, Toolbar is enabled for the user and Ad Blocking Detection is enabled (tab ⚙️ / tab Ad Blocking).



## Ad Blocking Status

When this function is enabled you will see a bar below the toolbar showing the status of ad blocking detection: whether ad blocking is detected, whether action cookie is present and how many page views were already counted (if used).



You can click on the bar to delete ad blocking detection cookies.

## Simulate Ad Blocking

With this function you can simulate and test ad blocking detection without actual ad blocking software. Ad Inserter will behave as there is ad blocking present. Of course, no ads will be blocked, only plugin ad blocking functions will be triggered when the page is loaded.

## Practical example of content protection

Here we have a post with few paragraphs. When ad blocking is used we would like to protect all the paragraphs except the first three. For protection we will use shortcode to delete content. This is the original post:

Test

### Paragraphs

📅 2017-08-02 👤 admin 💬 0 Comments 🏷 paragraphs ✎ Edit

Lorem ipsum dolor sit amet, mel aperiam corpora an. Pro no meis porro mollis, duo hinc atomorum omittantur cu, purto vituperatoribus te mei. Ne sadipscung efficiantur usu, an accusam inimicus gloriatur eam. Vidit menandri voluptaria eos ut, ei vis natum putant.

Vel eu mundi euripidis, lorem delectus assentior qui ei, inimicus gloriatur et vix. Omnium noluisse voluptaria per an. Te nec dicant invenire deseruisse, usu ne cibo forensibus. Id sea illum nonumes aliquando. Nam ea novum quodsi scripta.

Mel no eius solet delenit, erat ullum te vel, ad vim lorem mollis. Has eu tibi que dissentiet. Qui id tation qualisque, nec an aperiri propriae interesset. Veritus offendit id cum, id vel regione iracundia. At eam dicit eruditi detracto, viris latine percipit ad sed, pro justo novum elit ne.

Usu eligendi vivendum an, ad errem fuisset cum. Usu augue altera feugiat eu. Sea labore explicari no. Viris verear denique nec ad, vero putent inciderint ad quo. Malis salutandi definiebas id vix, duo an veri facilis, illud tation graecis nec ex.

No laudem ceteros duo, no has libris quaerendum. Vel quidam constituam et. Clita timeam id nam, ad tation detracto usu. Fugit ponderum mel et, sit tale sale libris te. Ad usu ferri officis, eu impedit nominati assentior mel, ullum impetus probatus cu duo. At solum mollis mea, pro saepe impetus fastidii ei.

Lorem ipsum dolor sit amet, mel aperiam corpora an. Pro no meis porro mollis, duo hinc atomorum omittantur cu, purto vituperatoribus te mei. Ne sadipscung efficiantur usu, an accusam inimicus gloriatur eam. Vidit menandri voluptaria eos ut, ei vis natum putant.

## Ad Inserter Pro settings for content protection of code block:

- Automatic insertion: Before paragraph
- Paragraph number: 4
- Alignment and Style: No Wrapping
- Code: `[ADINSERTER adb='delete']`

### Post Protection



```
[ADINSERTER adb='delete']
```

Lists Manual Devices Misc Preview Save Settings 17 - 32

Automatic Insertion:  Alignment and Style:  Show

Posts  Homepage  Category pages  
  Static pages  Search pages  Tag / Archive pages

Paragraph number(s)  Counting Clearance

Post/Static page must have between  and  words

We will not use shortcode to end protection as we are protecting until the end of the post. Now either activate ad blocking or use **Simulate Ad Blocking** debugging function and reload post. You should see only the first three paragraphs.

Test

## Paragraphs

2017-08-02 admin 0 Comments paragraphs Edit

Lorem ipsum dolor sit amet, mel aperiam corpora an. Pro no meis porro mollis, duo hinc atomorum omittantur cu, purto vituperatoribus te mei. Ne sadipscing efficiantur usu, an accusam inimicus gloriatur eam. Vidit menandri voluptaria eos ut, ei vis natum putant.

Vel eu mundi euripidis, lorem delectus assentior qui ei, inimicus gloriatur et vix. Omnium noluisse voluptaria per an. Te nec dicant invenire deseruisse, usu ne cibo forensibus. Id sea illum nonumes aliquando. Nam ea novum quodsi scripta.

Mel no eius solet delenit, erat ullum te vel, ad vim lorem mollis. Has eu tistique dissentiet. Qui id tation qualisque, nec an aperiri propriae interesset. Veritus offendit id cum, id vel regione iracundia. At eam dicit eruditi detracto, viris latine percipit ad sed, pro justo novum elitr ne.

## Example with changed font size and color

- Automatic insertion: Before paragraph
- Paragraph number: 4
- Alignment and Style: No Wrapping
- Code: `[ADINSERTER adb='css' css='font-size: 11px; color: #ddd;']`

The post when adblocking is active:

Test

# Paragraphs

📅 2017-08-02 👤 admin 💬 0 Comments 🏷 paragraphs ✎ Edit

Lorem ipsum dolor sit amet, mel aperiam corpora an. Pro no meis porro mollis, duo hinc atomorum omittantur cu, purto vituperatoribus te mei. Ne sadipscing efficiantur usu, an accusam inimicus gloriatur eam. Vidit menandri voluptaria eos ut, ei vis natum putant.

Vel eu mundi euripidis, lorem delectus assentior qui ei, inimicus gloriatur et vix. Omnium noluisse voluptaria per an. Te nec dicant invenire deseruisse, usu ne cibo forensibus. Id sea illum nonumes aliquando. Nam ea novum quodsi scripta.

Mel no eius solet delenit, erat ullum te vel, ad vim lorem mollis. Has eu tibiue dissentiet. Qui id tation qualisque, nec an aperiri propriae interesset. Veritus offendit id cum, id vel regione iracundia. At eam dicit eruditi detracto, viris latine percipit ad sed, pro justo novum elitr ne.

Usu eligendi vivendum an, ad errem fuisset cum. Usu augue altera feugiat eu. Sea labore explicari no. Viris verear denique nec ad, vero putent inciderint ad quo. Malis salutandi definiebas id vix, duo an veri facilis, illud tation graecis nec ex.

No laudem ceteros duo, no has libris quaerendum. Vel quidam constituam et. Clita timeam id nam, ad tation detracto usu. Fugit ponderum mel et, sit tale sale libris te. Ad usu ferri officilis, eu impedit nominati assentior mel, ullum impetus probatus cu duo. At solum mollis mea, pro saepe impetus fastidii ei.

Lorem ipsum dolor sit amet, mel aperiam corpora an. Pro no meis porro mollis, duo hinc atomorum omittantur cu, purto vituperatoribus te mei. Ne sadipscing efficiantur usu, an accusam inimicus gloriatur eam. Vidit menandri voluptaria eos ut, ei vis natum putant.

Vel eu mundi euripidis, lorem delectus assentior qui ei, inimicus gloriatur et vix. Omnium noluisse voluptaria per an. Te nec dicant invenire deseruisse, usu ne cibo forensibus. Id sea illum nonumes aliquando. Nam ea novum quodsi scripta.

## Example with replaced text

- Automatic insertion: Before paragraph
- Paragraph number: 4
- Alignment and Style: No Wrapping
- Code: `[ADINSERTER adb='replace' text='You are using Ad Blocker. ']`

The post when adblocking is active:

Test

# Paragraphs

📅 2017-08-02 👤 admin 💬 0 Comments 🏷 paragraphs ✎ Edit

Lorem ipsum dolor sit amet, mel aperiam corpora an. Pro no meis porro mollis, duo hinc atomorum omittantur cu, purto vituperatoribus te mei. Ne sadipscing efficiantur usu, an accusam inimicus gloriatur eam. Vidit menandri voluptaria eos ut, ei vis natum putant.

Vel eu mundi euripidis, lorem delectus assentior qui ei, inimicus gloriatur et vix. Omnium noluisse voluptaria per an. Te nec dicant invenire deseruisse, usu ne cibo forensibus. Id sea illum nonumes aliquando. Nam ea novum quodsi scripta.

Mel no eius solet delenit, erat ullum te vel, ad vim lorem mollis. Has eu tibiue dissentiet. Qui id tation qualisque, nec an aperiri propriae interesset. Veritus offendit id cum, id vel regione iracundia. At eam dicit eruditi detracto, viris latine percipit ad sed, pro justo novum elitr ne.

You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker.

You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker.

You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker.

You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker. You are using ad blocker.

## Example with ad blocking information and deleted text

This is probably the best approach for content protection as you are protecting the content and you are also informing the visitor why the content is missing.

One code block for the message:

- Automatic insertion: After paragraph
- Paragraph number: 3
- When ad blocking is detected: Show
- Code:

```
<hr />
<p><strong>Blocked because of Ad Blocker</strong></p>
<p>It seems that you are using some ad blocking software which is preventing the page from fully loading. Please whitelist this website or disable ad blocking software.</p>
```

One code block for content protection:

- Automatic insertion: Before paragraph
- Paragraph number: 4
- Alignment and Style: No Wrapping
- Code: `[ADINSERTER adb='delete']`

We can't use one code block for the message and shortcode since the message needs wrapping div to show it only when ad blocking is detected.

The post when adblocking is active:

Test

## Paragraphs

📅 2017-08-02 👤 admin 💬 0 Comments 🗑 paragraphs ✎ Edit

Lorem ipsum dolor sit amet, mel aperiam corpora an. Pro no meis porro mollis, duo hinc atomorum omittantur cu, purto vituperatoribus te mei. Ne sadipscing efficiantur usu, an accusam inimicus gloriatur eam. Vidit menandri voluptaria eos ut, ei vis natum putant.

Vel eu mundi euripidis, lorem delectus assentior qui ei, inimicus gloriatur et vix. Omnium noluisse voluptaria per an. Te nec dicant invenire deseruisse, usu ne cibo forensibus. Id sea illum nonumes aliquando. Nam ea novum quodsi scripta.

Mel no eius solet delenit, erat ullum te vel, ad vim lorem mollis. Has eu tistique dissentiet. Qui id tation qualisque, nec an aperiri propriae interesset. Veritus offendit id cum, id vel regione iracundia. At eam dicit eruditi detracto, viris latine percipit ad sed, pro justo novum elitr ne.


---

**Blocked because of Ad Blocker**

It seems that you are using some ad blocking software which is preventing the page from fully loading. Please whitelist this website or disable ad blocking software.

# Ad Inserter General Plugin Settings



Ad Inserter general plugin settings are available on the  tab. Plugin settings are divided into tabs – some are available only in Pro version.

## General

General Viewports Hooks Header Footer Geolocation Exceptions Tracking Ad Blocking Debugging

License Key

Plugin priority

Output buffering  ▾

Syntax highlighting theme  ▾

Min. user role for ind. exceptions editing  ▾

Sticky widget mode  ▾

Sticky widget top margin  px

Dynamic blocks  ▾

Functions for paragraph counting  ▾

No paragraph counting inside

Ad label

Max blocks per page

Main content element

Lazy loading offset  px

Block class name  Block class  Block number class  Inline styles

Preview of the block wrapping code

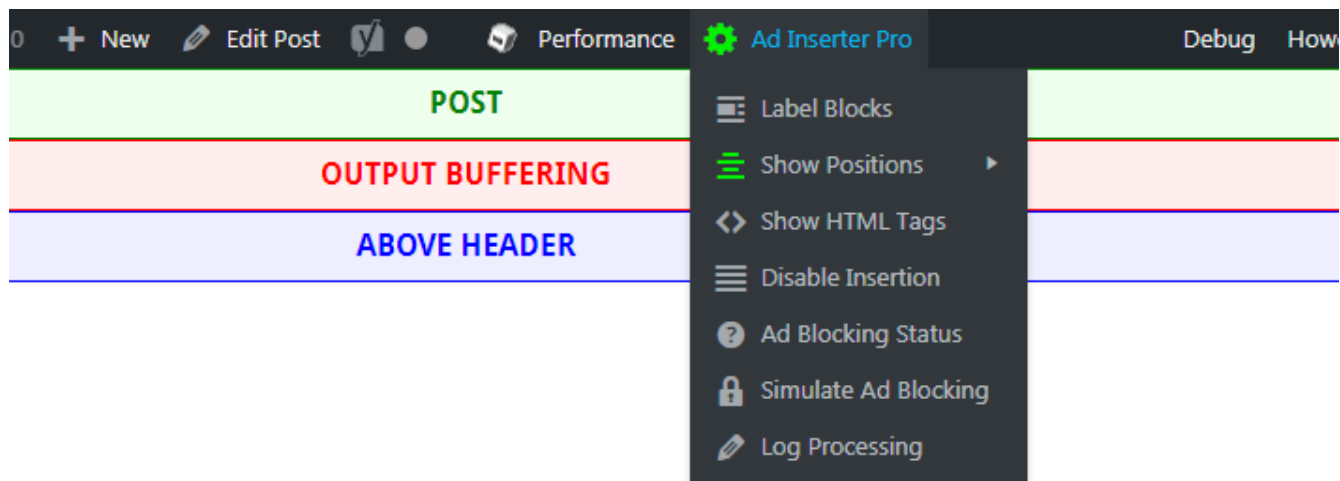
```
<div class="code-block code-block-default code-block-n">
  BLOCK CODE
</div>
```

Default Ad Inserter **plugin priority** (processing order) is 99999. It is used to specify the order in which the plugin functions are executed. Lower numbers correspond with earlier execution. You can change this value if you have problems with the processing order of other plugins.

Insertion position **Above header** and server-side insertion **Before/After HTML element** need special code processing and are **available only** when you enable **Output buffering**, however, this may not work with all themes. Output buffering means that the code generated by the WordPress (including your theme and plugins)



is buffered (saved) and then processed. To check if your theme supports output buffering, go to general plugin settings and enable it, then go to homepage and enable **Show positions** debugging function (Ad Inserter menu in top admin bar). If you see **OUTPUT BUFFERING** (and **ABOVE HEADER**) bar then output buffering works as expected.



You can choose between many **syntax highlighting themes** (light and dark).

By default Ad Inserter individual exceptions on posts/static pages are enabled only for administrators. You can define **minimum user role** for page/post Ad Inserter individual exceptions editing in Posts/pages editor.

There are two modes for sticky widgets (or sticky sidebars) available as **Sticky widget mode**. CSS mode is the best approach but may not work with all themes. Use JavaScript mode only when CSS mode does not work. Check Widgets for more.

Here you can also define **Sticky Widget Top Margin** to precisely define top position where first sticky widget will stop.

Ad Inserter supports **server-side** and **client-side dynamic blocks** (ad rotation, IP address and country detection). If you are using caching you should use client-side (or server-side using W3 Total Cache if you are using this caching plugin) but you need to be aware of how it works. Please check section Caching for details.

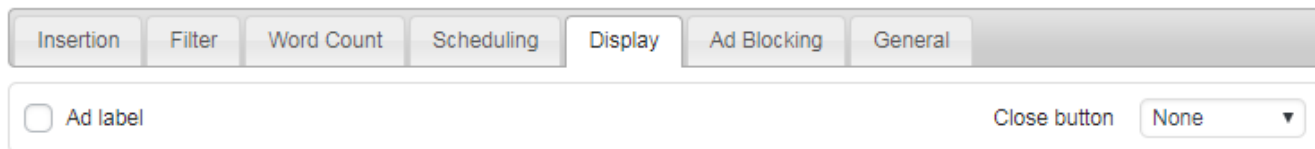
When you set **Dynamic blocks** to **client-side** the plugin will generate code that will be executed in the visitor's browser. Ad Inserter supports two client-side modes for dynamic blocks: **Client-side show** and **Client-side insert**. Both generate Javascript code to run in the browser. **Client-side show** uses CSS to hide unwanted code while **Client-side insert** uses Javascript to insert code when needed or enabled. **Client-side show** is fine with banners as they are all loaded and only shown when needed but some adverts might not work properly or can't be used this way and you might violate their Terms of Service – use **Client-side insert** mode in such case.

**PLEASE NOTE:** some ad codes use old JavaScript method `document.write` which can't be used asynchronously (after the page is loaded and created). **Client-side insert** mode will not work with such ad codes.

Standard PHP **functions for paragraph counting** are faster and work in most cases, use Multibyte functions if paragraphs are not counted properly on non-english pages.

**No Paragraph Counting Inside:** here you can define tags inside which paragraphs (in posts on pages) will not be counted when using Before/After **paragraph** automatic insertion.

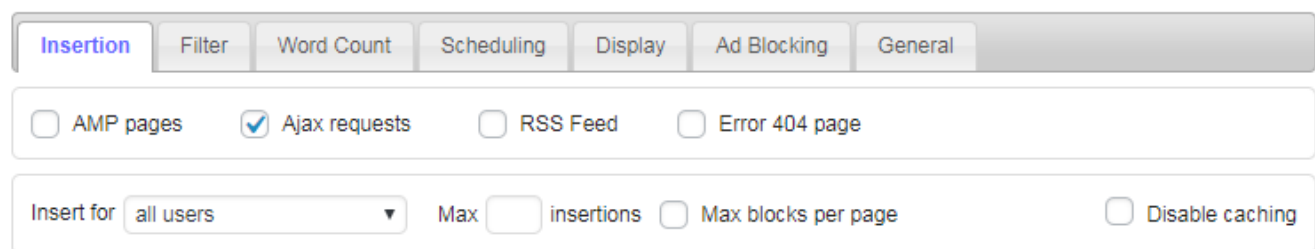
**Ad label** defines text which can be displayed above ads. For each code block (ad) you can also show a label above each ad. To enable ad label above the ad go to code block for this ad, click on the **Misc** button and then on the **Display** tab check **Ad label**.



If you configured only text for **Ad label** then the following code will be added above ad code:

```
<div class="code-block-label">AD_LABEL_TEXT</div>
```

You can also limit maximum number of blocks per page with **Max blocks per page**. If you want some block to count for maximum blocks per page check **Max blocks per page** (button **Misc** / tab **Insertion**).



**Stick to the content** horizontal position displays ads next to the left or right edge of page content (as sidebar). In most cases the plugin will automatically calculate page content width (assuming that the content is centered horizontally) and position ads automatically. However, some themes may require to define **Main content element** manually. Here you can enter either CSS selector (**#id** or **.class**) of the main (outermost) page element or content width in pixels (numerical value only).

**Lazy loading offset** defines how close to the viewport (in pixels) lazy loading ad must be in order to be loaded. Lazy loading can be enabled on tab **Display** (button **Misc**).

`code-block` is the **Block class name** as defined below, `AD_LABEL_TEXT` is the text for Ad Label. You can then style the label using CSS for class `.code-block-label` (the best place for this code is Header section). If HTML code is configured for ad label then this code will be used (instead of the `div` code above).

In case you need a different label for some ad you can simply add label HTML code above the ad code. For example:

```
<h3>Advertisements</h3>
```

**Block class name** defines class that is used in wrapping divs. Wrapping divs for code blocks have `code-block` and `code-block-n` classes which can be used for custom styles. Both classes can be disabled if needed. If you are using client-side device detection (CSS media queries) then the wrapping div for the code block will have also some of the following classes: `ai-viewport-1`, `ai-viewport-2`,... `ai-viewport-n`.

```
<div class="code-block code-block-n" style="margin: 8px 0; clear: both;">
  BLOCK CODE
</div>
```

By default each code block uses **inline styles** for alignment. You can disable them to use alignment classes instead.

## Viewports

Up to 3 (6 in Pro) viewport names and widths can be defined – they are used for client-side device detection.

Viewport Settings used for client-side device detection


Viewport 1 name	<input type="text" value="Desktop"/>	min width	<input type="text" value="980"/>	px
Viewport 2 name	<input type="text" value="Tablet"/>	min width	<input type="text" value="768"/>	px
Viewport 3 name	<input type="text" value="Phone"/>	min width	<input type="text" value="700"/>	px
Viewport 4 name	<input type="text"/>	min width	<input type="text" value="0"/>	px
Viewport 5 name	<input type="text"/>	min width	<input type="text" value="0"/>	px
Viewport 6 name	<input type="text"/>			

## Header and Footer code

Ad Inserter supports insertion of code into the following HTML page sections:

- Header code (scripts in the `<header>` section)
- Footer code (scripts before the `</body>` tag)

You can also use **shortcodes** for code blocks to insert page-specific code, for example to insert the code only on specific pages (either use `[adinserter code="BLOCK_NUMBER"]` or set Style to **No wrapping** to insert only block code without wrapping div) and you can use `[ADINSERTER AMP]` to separate code for normal and AMP pages - the code above the separator will be inserted on normal pages, the code below the separator will be inserted on AMP pages. Don't forget to **enable header/footer code** – button next to PHP button.

PLEASE NOTE: Insertion position **Above header** inserts code after `<body>` tag, while **Header** code in general plugin setting (tab ) inserts the code (scripts) in the `<head>` section of the HTML page.

Example of **Header** code (code in the `<head>` section):

General Viewports Hooks **Header** Footer Ad Blocking Debugging

### HTML Page Header Code

Code in the `<head></head>` section of the HTML page

```
<script async src="//pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
<script>
  (adsbygoogle = window.adsbygoogle || []).push({
    google_ad_client: "ca-pub-4334549272393622",
    enable_page_level_ads: true
  });
</script>
```

Example of **Footer** code (code before the `</body>` tag):

General Viewports Hooks **Header** Footer Ad Blocking Debugging

### HTML Page Footer Code

Code before the </body> tag of the the HTML page


```
<script src="https://cdn.rawgit.com/michalsnik/aos/2.0.1/dist/aos.js"></script>
```

## Page-specific Header and Footer code

Normally the code for Header and Footer will be inserted on all pages. In some cases you may need to insert code in header or footer only on specific pages or to exclude insertion on some pages. There are basically two approaches for this. In both cases you need to configure one code block with the code you need to insert. Set **Alignment and Style** to **No wrapping** (this code needs no wrapping div), enable wanted Page types (normally all page types should be enabled: posts, pages, homepage,...) and configure other block settings to limit insertions (in most cases you'll need lists for this).

- Approach with **shortcodes**: set **Automatic insertion** to **Disabled**, enable **shortcode** for this code block (button **Manual**) and put shortcode for the code of this block into **Header** or **Footer** code window: `[adinsserter code="N"]`. This shortcode for block N will insert only the code without wrapping div.

**WARNING:** if you are using individual exceptions please note that shortcodes **IGNORE** post/static page exception settings! To force checking individual post/page exceptions use shortcode `[adinsserter code="N" check="exceptions"]`.

- Approach with **Automatic insertion**: set **Automatic insertion** to **Header** (code in the `<head></head>` section) or **Footer** (code before the `</body>` tag). Normally, position for automatic insertion **Header** does not exist and needs to be created. Go to tab  / tab **Hooks** and create and enable hook with name **Header** and action `wp_head`. After you save settings you'll get **Header** position for automatic insertion for each code block. Please note that this Header position for automatic insertion has nothing to do with the header of your theme. This is HTML page header - code in the invisible `<head></head>` section.

## Raw HTTP Header

You can also insert raw HTTP header lines using `[ADINSERTER HTTP]` separator. Header code above the separator will be inserted into HTTP header. You can also use PHP code and functions for HTTP header (header, setcookie,...). Each raw header line must contain column. For example, the following header code block:

```
<?php
header ('AI-PHP-header1: PHP1');
header ('AI-PHP-header2: TWO');
echo 'AI-PHP-header3: A?';
setcookie ('utm_test', 'ad_insserter', time() + (1 * 60), COOKIEPATH);
?>
```

```
AI-header1: test1
AI-header2: test2
TEST
```

```
[ADINSERTER HTTP]
<!-- NORMAL HEADER CODE -->
```

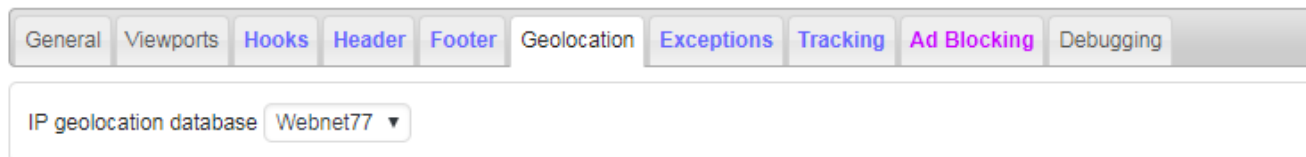
```
[ADINSERTER AMP]
<!-- AMP HEADER CODE -->
```

would insert the following raw HTTP header lines:

```
AI-PHP-header1: PHP1
AI-PHP-header2: TWO
AI-PHP-header3: A?
Set-Cookie: utm_test=ad_inserter; expires=Sat, 14-Oct-2017 18:56:09 GMT; Max-Age=60; path=/
AI-header1: test1
AI-header2: test2
```

## Geolocation (Pro only)

Geolocation (or GEO targeting) means determining visitor's country and display targeted ads. Country is determined from the IP address of the website visitor. IP/country address database is updated automatically each month. You can choose between Webnet77 (default) and MaxMind geolocation database (Ad Inserter Pro will download and use GeoLite2 data created by **MaxMind**). Ad Inserter Pro supports also **CloudFlare geolocation**.

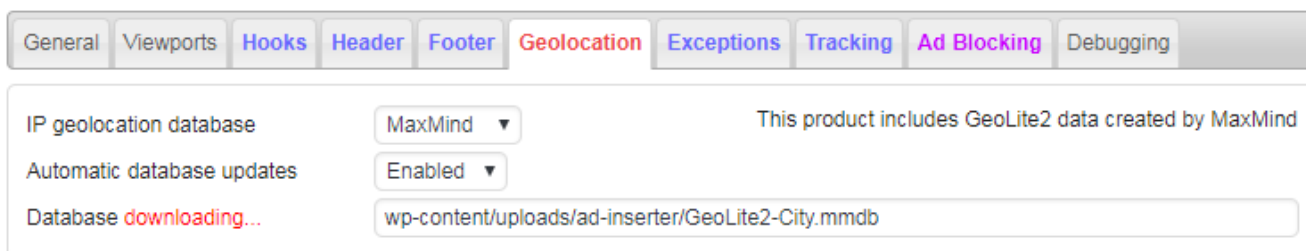


General Viewports Hooks Header Footer Geolocation Exceptions Tracking Ad Blocking Debugging

IP geolocation database Webnet77 ▼

Default Webnet77 database is lightweight (around 2 MB) but slightly less accurate than MaxMind. If you select MaxMind database and enable automatic updates Ad Inserter Pro will download and use GeoLite2 database (around 60 MB, data created by **MaxMind**).

When you select MaxMind for geolocation database and save settings Ad Inserter Pro will show additional options and display a warning message because the database file will not be found. Now you can either manually download and extract the file (useful if you use commercial MaxMind database) or set **Automatic database updates** to Enabled. In such case Ad Inserter Pro will download GeoLite2 database and the warning should disappear. Default database location is in the uploads folder but you can define any location on the server. If you use absolute path it must start with a slash.



General Viewports Hooks Header Footer Geolocation Exceptions Tracking Ad Blocking Debugging

IP geolocation database MaxMind ▼ This product includes GeoLite2 data created by MaxMind

Automatic database updates Enabled ▼

Database downloading... wp-content/uploads/ad-inserter/GeoLite2-City.mmdb

In some cases MaxMind may limit downloading and you may get error message when the plugin tries to download the file. In such case you can manually download the file, **extract it** (use any gzip tool) and upload GeoLite2-City.mmdb file to your hosting server to the configured location.









MaxMind GeoLite2 database (gzipped) for manual download:


<http://geolite.maxmind.com/download/geoip/database/GeoLite2-City.mmdb.gz>


## GEO targeting Groups


You can define up to 6 country groups for easier GEO targeting of individual code blocks. Then you can use group codes in the country list of each code block.

Group 1  countries

-  Afghanistan (AF)
-  Aland Islands (AX)
-  Albania (AL)
-  Algeria (DZ)
-  American Samoa (AS)
-  Andorra (AD)
-  Angola (AO)
-  Anguilla (AI)

 Austria (AT)

 Germany (DE)

 Switzerland (CH)

Group 2  countries

Group 3  countries

Group 4  countries

Group 5  countries

Group 6  countries

## Exceptions (Pro only)

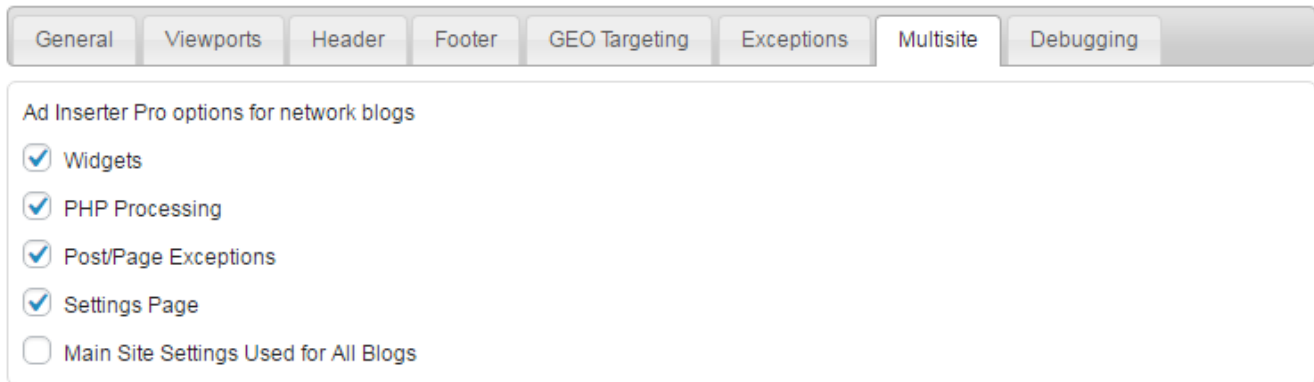
Here you can open page with exceptions, open editor to edit exceptions, clear exceptions for specific block, clear exceptions for specific post/page or clear all exceptions (click on the cross in the top right corner).

General Viewports Header Footer GEO Targeting Exceptions Debugging

ID	Type	Page / Post	<span style="float: right; border: 1px solid red; padding: 2px;">✕</span>															
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
31608	Page	Activity	X															X
2	Page	About	X															X
31609	Page	Members	X															X
31615	Post	Lorem Ipsum	X															X
31516	Post	How To Become A Good Essay Writer	X															X
31514	Post	Travel Risk Management	X															X

## Multisite (Pro only)

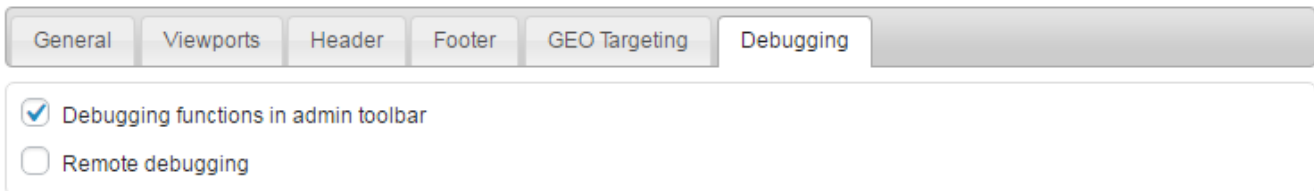
Ad Inserter supports multisite WordPress installations. Normally, the plugin is available with settings and widgets to all the blogs on the network. Ad Inserter Pro supports options to disable Settings page, widgets, PHP processing and post/page exceptions for sites (except the main one).



The screenshot shows the 'Multisite' tab selected in a settings menu. The menu includes tabs for General, Viewports, Header, Footer, GEO Targeting, Exceptions, Multisite, and Debugging. The main content area is titled 'Ad Inserter Pro options for network blogs' and contains five checkboxes: 'Widgets' (checked), 'PHP Processing' (checked), 'Post/Page Exceptions' (checked), 'Settings Page' (checked), and 'Main Site Settings Used for All Blogs' (unchecked).

It is also possible to use settings for the main site (blog) on all network blogs – in such case the **Settings Page** option is disabled.

## Debugging



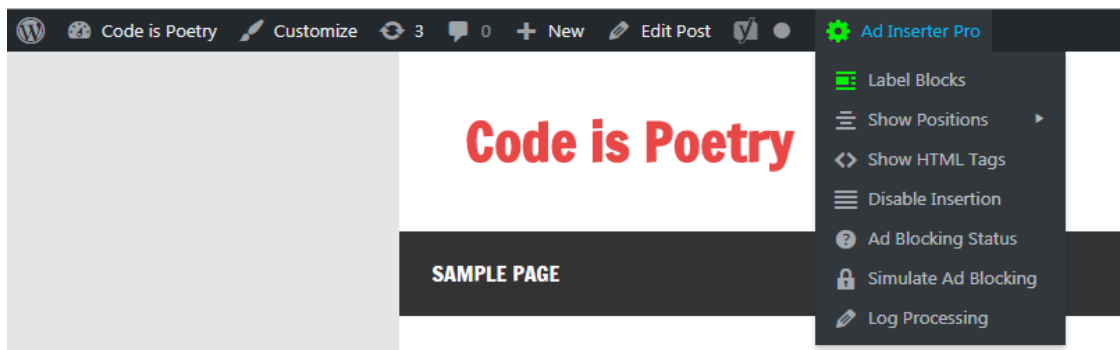
The screenshot shows the 'Debugging' tab selected in a settings menu. The menu includes tabs for General, Viewports, Header, Footer, GEO Targeting, and Debugging. The main content area contains two checkboxes: 'Debugging functions in admin toolbar' (checked) and 'Remote debugging' (unchecked).

Debugging functions in admin toolbar are enabled by default. If you don't need them you can disable them here. Remote debugging enables other, non-logged in users by using url parameters to see Debugger widget and code insertion debugging (blocks, positions, tags, processing). Enable this option to allow other people to see Debugger widget, labeled blocks and positions in order to help you to diagnose problems. For logged in administrators debugging is always enabled. See Debugging section for more information.



# Debugging

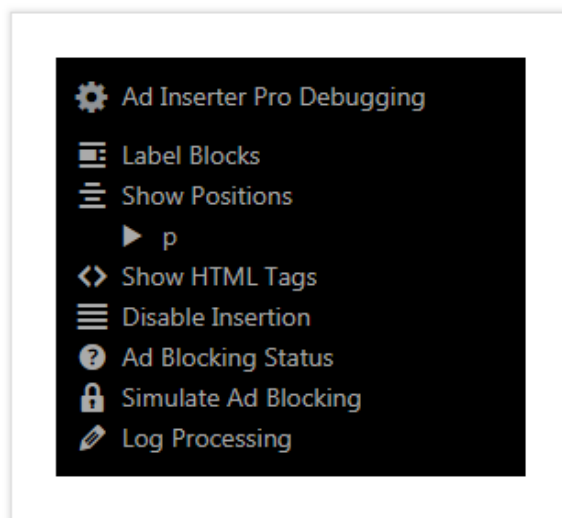
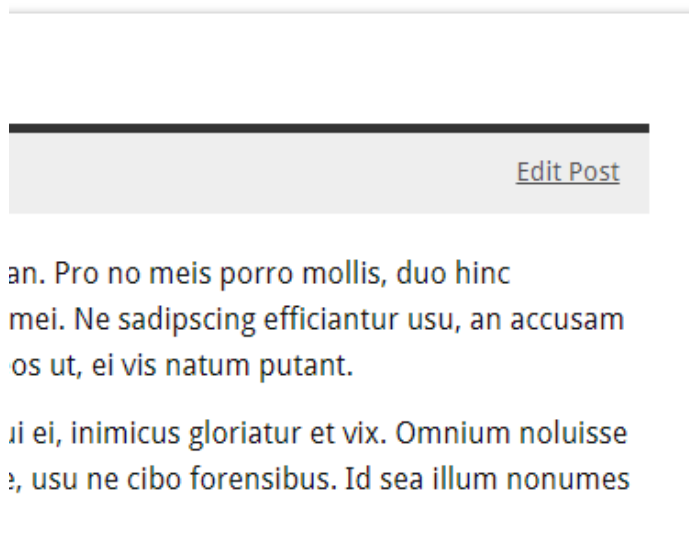
Ad Inserter has many debugging functions that can help you to diagnose the problem when you don't see your ads at expected positions. When you are logged in each post/page has a WordPress toolbar on the top. When you are logged in as administrator the toolbar contains also Ad Inserter menu item.



**WARNING: Make sure caching is disabled while debugging!** If you don't see the admin toolbar check Ad Inserter Plugin settings tab ⚙️ / tab **Debugging** and make sure **Debugging functions in admin toolbar** is checked and admin toolbar is enabled (WordPress menu **Users** / **Your Profile** / **Toolbar**).

## Debugging Tools widget

It may happen that your theme does not support the toolbar or it is not displayed for some other reason. In such case you can use Ad Inserter **Debugging Tools** widget which has the same menu items. Simply drag Ad Inserter widget into the sidebar and select **Debugging Tools**. Of course, you still need to be logged in as administrator to see the widget.



The following debugging functions are available:

- **Label Blocks:** Each inserted block is labeled with a thin red border and red bar with block number, name and counters. Blocks that use client-side detection and are hidden are shown with blue bar and viewport name - resize the browser to check display for other devices (or screen widths). If you see only red bar then this means that **the block with your code is inserted** but the code doesn't display anything.
- **Show Positions:** Enable this function to show available positions for automatic insertion. Displayed positions are based on the theme layout and configured paragraph counting. You can choose between all paragraph tag lists (or counting parameters) used for blocks configured for Before or After paragraph. If you click on the Show Positions menu item you'll see default paragraph positions for p tags.
- **Show HTML tags:** Enable this function to see HTML tags used in the post. Use this function to determine post structure in order to configure paragraph counting.
- **Disable insertion:** Use this function to temporarily disable insertion of code blocks - everything else on the page will look like the code blocks were processed and inserted.
- **Ad Blocking Status**  
When this function is enabled you will see a bar below the toolbar showing the status of ad blocking detection: whether ad blocking is detected, whether action cookie is present and how many page views were already counted (if used). You can click on the bar to delete ad blocking detection cookies. Ad Blocking Detection needs to be enabled to use this function.
- **Simulate Ad Blocking**  
With this function you can simulate and test ad blocking detection without actual ad blocking software. Ad Inserter will behave as there is ad blocking present. Of course, no ads will be blocked, only plugin ad blocking functions will be triggered when the page is loaded. Ad Blocking Detection needs to be enabled to use this function.

- **Log Processing:** Use this function to log insertion process in order to determine why some code block was not inserted. **The log is added as HTML comment at the end of the page - check page source.**

**Processing Log** widget: Place **Processing Log** widget in some widget area to see basic WordPress page and Ad Inserter data (User status, Page Type, Post ID, Url, Referer, etc). With this widget you can also check saved block settings and client-side viewport name. **Processing Log** Widget in Ad Inserter Pro shows also visitor's IP address and country. The widget shows the same data as **Log processing** function.

```

Ad Inserter          2.0.0

GENERATED (WP time): 2016-09-24 13:34:02
GENERATED (Server time): 2016-09-24 11:34:02
SETTINGS:           2.0.0
SETTINGS TIMESTAMP: 2016-09-24 11:34:47
MULTISITE:          NO
USER:               LOGGED-IN ADMINISTRATOR
PAGE TYPE:          CATEGORY PAGE
ID:                 31371
URL:                /category/fun-stuff
REFERER:            http://example.com/
CLIENT-SIDE DETECTION: NOT USED
SERVER-SIDE DETECTION: NOT USED

BLOCK SETTINGS      Po Pa Hp Cp Ap Sp Fe 404 Wi Sh PHP
1 AdSense           o . . . . . . . . . . None Custom CSS paragraph_number: 2 min_paragraphs
2 Block 2           o . . . . . . . . . . x . . . None Float Right min_paragraphs: 0 min_words: 0 mi
3 Block 3           o . . . . . . . . . . x . . . None Float Left min_paragraphs: 0 min_words: 0 min
4 Block 4           o . . . . . . . . . . x . . . None Center min_paragraphs: 0 min_words: 0 min_par
5 Block 5           o . . . . . . . . . . x . . . None No Wrapping min_paragraphs: 0 min_words: 0 mi
6 Block 6           o . . . . . . . . . . . . . . None No Wrapping min_paragraphs: 0 min_words: 0 mi
7 AdSense Post      o . . . . . . . . . . . . . . Before Paragraph Float Left paragraph_number: 3 mi
8 Block 8           o . . . . . . . . . . . . . . None No Wrapping min_paragraphs: 0 min_words: 0 mi
9 AdSense Sidebar   o . . . . . . . . . . . . . . None Custom CSS min_paragraphs: 0 min_words: 0 min
10 Block DESET      o . . . . . . . . . . . . . . None Custom CSS min_paragraphs: 0 min_words: 0 min
11 Block 11         o . . . . . . . . . . . . . . None Custom CSS min_paragraphs: 0 min_words: 0 min
12 Block 12         o . . . . . . . . . . . . . . None Custom CSS min_paragraphs: 0 min_words: 0 min
13 Block 13         o . . . . . . . . . . . . . . None Float Left min_paragraphs: 0 min_words: 0 min
14 /effective-ways-that- o . . . . . . . . . . . . . . None None min_paragraphs: 0 min_words: 0 min_parag
15 Block 15         o . . . . . . . . . . . . . . None No Wrapping process_php: 1 min_paragraphs: 0 :
16 Block 16         o . . . . . . . . . . x . . . Before Paragraph No Wrapping paragraph_number: 2 m

TOTAL BLOCKS
CONTENT HOOK BLOCKS: 7, 16

PAGE TYPE BLOCKS

```



Example of page with enabled debugging functions:

**BEFORE POST**

[← Previous](#)

## Lorem Ipsum

Posted on [December 19, 2015](#) by [admin](#)

**BEFORE CONTENT** 1032 words

**BEFORE PARAGRAPH 1**

**<p>** Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam nec nunc ultricies, blandit tortor nec, tincidunt nisl. Suspendisse potenti. Pellentesque cursus justo sed lectus bibendum pulvinar. Sed blandit nisl sed urna sollicitudin, id mattis ex rhoncus. Integer tristique nulla in lacus lacinia, non pharetra arcu tempor. Proin ultricies metus augue, eget semper dolor facilisis accumsan. Maecenas tellus magna, interdum id rutrum vitae, rhoncus quis mi. Sed odio nisi, congue id tempor ultricies, sollicitudin id ex. Nullam id eros feugiat tellus congue ullamcorper. Nunc quis nisl ut nisl dictum ultrices id nec ipsum. Curabitur a dui id dui blandit malesuada sed at elit. Donec aliquam lacus quis sem gravida ultricies. Fusce sed elementum lacus. **</p>**

**AFTER PARAGRAPH 1**

**BEFORE PARAGRAPH 2**

16 Block 16 C=1 N=1

### High Quality Stock Photos

▶

Free HD Images - Over 30 Million At The Premiere Stock Photo Network

● ○▶

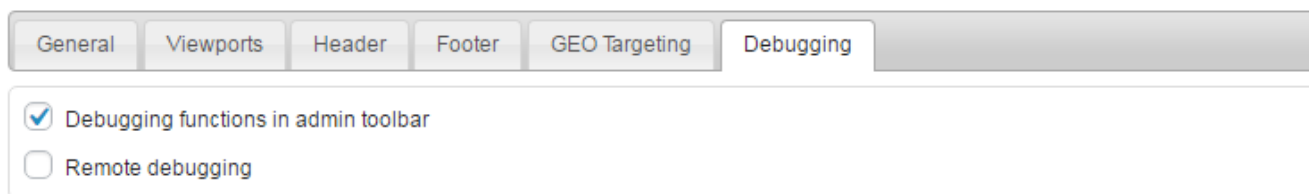
**<p>** <span id="more-31615"> **</p>**

**WARNING: Make sure caching is disabled while debugging!** All debugging functions you enable will be visible only to you! Post/page debugging works by adding url parameters to the url (web address):

- Label Blocks: `ai-debug-blocks=1` (use `ai-debug-blocks=0` to turn it off)
- Show Positions: `ai-debug-positions=BLOCK_NUMBER`, 0 means default counting of paragraphs with `p` tags, numbers between 1 and 16 (64 in Pro) use paragraph counting as configured for the block (use `ai-debug-positions=` to turn it off)
- Show HTML tags: `ai-debug-tags=1` (use `ai-debug-tags=0` to turn it off)
- Disable insertion: `ai-debug-no-insertion=1` (use `ai-debug-no-insertion=0` to turn it off)
- Log Processing: `ai-debug-processing=1` (use `ai-debug-processing=0` to turn it off)
- Country test: `ai-debug-country=XX` (replace XX with country ISO Alpha-2 code)
- IP address test: `ai-debug-ip-address=IP_ADDRESS`
- Ad blocking simulation: `ai-debug-adb=1`
- Ad blocking detection status: `ai-debug-adb-status=1`

Country test and IP address test work only when you have enabled debugging on some page and debugging cookie is created.

When browsing other pages on the website debugging settings are temporarily saved (for the session). To disable all debugging functions click on the **Ad Inserter** top menu item in the toolbar (or use `ai-debug=0`)



The image shows a horizontal toolbar with several tabs: General, Viewports, Header, Footer, GEO Targeting, and Debugging. The Debugging tab is currently selected. Below the tabs, there are two checkboxes: 'Debugging functions in admin toolbar' which is checked, and 'Remote debugging' which is unchecked.

If you enable **Remote debugging** you can also allow other people using url parameters to see post/page with debugging data. Remote debugging option is located on the Ad Inserter Settings tab - Debugging tab below. Remote debugging enables other, non-logged in users by using url parameters to see Debugger widget and code insertion debugging (blocks, positions, tags, processing).

**Enable this option to allow other people to see Debugger widget, labeled blocks and positions in order to help you to diagnose problems.** For logged in users debugging is always enabled.

## Troubleshooting

---

Free **Ad Inserter** can be installed from Wordpress (Plugins / Add New / search for Ad Inserter). It contains most of the functionality of the Pro version. The main difference is the number of code blocks supported and support via email - see below for details. Before you purchase license for Ad Inserter Pro please check whether free Ad Inserter can be installed on your website, it works as expected, you are not experiencing issues and it fits your needs. If free Ad Inserter does not work, Ad Inserter Pro will also not work. If you are having issues with the free Ad Inserter please use [Ad Inserter WP support forum](#) and ask for help or advice there.

### Plugin upload fails

Ad Inserter Pro must be installed from the file. You get the download link when you purchase the license. In some cases WordPress will not allow you to upload the plugin. This is very likely because your web hosting has set limitations on upload file size. In such case you can do the one following suggestions:

1. Upload the plugin manually with FTP client, [WP File Manager](#) plugin or hosting cPanel File Manager - Extract the zip file and upload the [ad-inserter](#) (or [ad-inserter-pro](#)) folder to `/wp-content/plugins/` folder of your website. Make sure you upload the innermost [ad-inserter](#) folder located inside the zip file and not the folder that may be created when you unzip the file. Inside the [ad-inserter](#) folder there should be plugin files and plugin subfolders.
2. Increase the upload file size limit - In the folder `/wp-admin/` create file `php.ini` with the following content:  

```
upload_max_filesize = 64M
post_max_size = 64M
```
3. Contact your hosting company to investigate the problem and install the plugin for you.
4. Contact us and provide hosting cPanel and Wordpress admin access and we'll try to help you.

### Ads not displayed

One of the common problems is that you don't see the ad displayed where you expect it. In such case you should first check if the code is inserted. This can be checked either by examining the source code of the page (make sure the caching is disabled) or by debugging function **Label Blocks**. If the code block is inserted then Ad Inserter works as expected, just the code doesn't display anything – you should check the code. But if the code block is not inserted then you can use debugging functions that can help you to diagnose the problem – check Debugging section for details.

If you get blank AdSense ads then this might be because there is some error in the code (wrong IDs), your AdSense account is not fully approved yet, your website was not accepted or your AdSense account is banned. You can also try [Media net ads](#) as good alternative for contextual ads.

### Page blocked

**PAGE BLOCKED**

This page was not loaded properly. Please check browser and plugins that may block this page.  
If you are using Ad Blocker please select "Disable on this page" or "Don't run on this page".

First thing to check is Javascript console. Errors are the first indicator of what might be wrong.

Some themes like to add `async` or `defer` tags to javascript files. This changes execution order of the scripts needed by Ad Inserter and you may get errors or partially loaded page. Your theme or other plugins **SHOULD NOT MODIFY** the code generated by Ad Inserter! Most of the scripts used are loaded from the bottom of the page and the order of script execution is important for the code to work properly. Make sure your theme does not add `async` or `defer` tags to javascript files.

Some Ad Blockers and security plugins may also block Ad Inserter settings page. If you don't see normal tabs for code blocks or there is no save button, make sure you have white-listed Ad Inserter settings page.

You also get this warning when the settings page was not loaded properly due to some **Javascript error**. Check Javascript console in your browser.

## Incompatible file loaded

Ad Inserter 2.0.0 [Doc](#) [Donate](#) [Go Pro](#) [Review](#)

**javascript 1.7.0** Incompatible javascript file loaded, probably due to caching. Please try to delete browser's cache and reload this page to see if this warning will disappear.

You may get such warning if wrong (old) javascript or CSS file is loaded. In such cases clear browser's cache and all other caches used and reload the page (e.g. with Ctrl F5). Also make sure the caching software **DOES NOT REMOVE VERSION INFO** parameter from the url. Javascript file version and CSS file version must match plugin version number. Please note that the javascript file and CSS file are part of the plugin and are needed only for the settings page.

Caching on the backend side (Ad Inserter Settings page) may also cause some unwanted behavior if it is not done properly. The problem can occur when the plugin is updated since the new plugin also provides new javascript and CSS files. In order to prevent browsers from loading old js/css files the plugin appends version info as query parameter to js and css files needed.

For example, in the source code of the settings page it should be like this (note parameter `?ver=2.0.1`):

```
<script type='text/javascript' src='http://example.com/wp-content/plugins/ad-inserter/js/ad-inserter.js?ver=2.0.1'></script>
```

```
<link rel='stylesheet' id='ai-admin-css' href='http://example.com/wp-content/plugins/ad-inserter/css/ad-inserter.css?ver=2.0.1' type='text/css' media='all' />
```

However, on some websites this version parameter is removed (very likely due to aggressive caching) and this may cause problems:

```
<script type='text/javascript' src='http://example.com/wp-content/plugins/ad-inserter/js/ad-inserter.js'></script>
```

```
<link rel='stylesheet' id='ai-admin-css' href='http://example.com/wp-content/plugins/ad-inserter/css/ad-inserter.css' type='text/css' media='all' />
```

Because of this the old cached files (css and js) are loaded which cause warnings and unpredictable behavior. If you are using caching **make sure the caching software DOES NOT REMOVE VERSION INFO (QUERY**



**STRING) parameter from the url.** This is needed for browsers to reload the file when the plugin is updated. Possible caching utilities that might affect query strings:

**Cloudflare:** Caching level / Deselect **Ignore Query String** as it may remove query strings

**W3 Total Cache:** Uncheck Browser cache / **Remove query strings from static resources** (although it should not cache backend settings page)

## Buttons missing or page elements distorted

Some plugins load their CSS files also on the settings pages of other plugins including Ad Inserter. Their CSS code overrides CSS styles of Ad Inserter and consequently changes the elements on the settings page.

OptimizePress plugin is known to cause such issues. It loads this file on the settings page of Ad Inserter:

</wp-content/plugins/optimizePressPlugin/lib/css/assets.min.css?ver=2.5.14.1>

There is no reason to load such CSS files on the settings pages of Ad Inserter or other plugins. This is a very bad practice. If you have identified such case you can contact plugin authors to fix the issue or let us know and we'll contact them.

## document.write errors

Some ad codes use old JavaScript method `document.write` which can't be used asynchronously (after the page is loaded and created). **Client-side insert** mode and client-side device detection action **Insert** will not work with such ad codes.

## Tracking, ad blocking detection or other client-side code is not working

Check browser's console for errors or warning messages that may indicate what the problem is. The issue can be caused by the plugins that try to optimize Javascript code. Ad Inserter generates Javascript code that is needed for client-side functions (tracking, ad blocking detection, client-side rotation, client-side insertion,...). This code needs jQuery library for which the plugin makes sure it is loaded in the header before any code uses it. If some other plugin moves this library to the footer it will be no longer available when needed and consequently the client-side code will fail and report errors. You need to make sure all such optimizing plugins (e.g. **Better WordPress Minify**) DO NOT change or move the jQuery library to the footer. You have to configure them to leave the script at the original position.

In some cases you may get false detection of ad blockers. This is usually because there is some issue with the Javascript code used for ad blocking detection. Check Javascript console for errors or messages. Also make sure detection scripts are present in the page source code:

For Ad Inserter:        ads.js, sponsors.js

For Ad Inserter Pro:    ads.js, sponsors.js, advertising.js, adverts.js

**Those scripts should be present in the page footer and should not be removed or moved to inline code!**

If you are using plugins to optimize Javascript code like **Autoptimize** make sure the scripts listed above are excluded from optimization.

## Ad Blockers

There is no way to prevent Ad blockers. But they do not block Ad Inserter directly, they block the code you are inserting. There are two ways to try to avoid blocking:

- In Ad Inserter settings you can rename (or remove) class name for the wrapping divs. This way it would

be difficult to detect Ad Inserter code blocks. Of course, inserted code could still be disabled by ad blockers. You can also use No wrapping style and your code would be inserted as it is. Again, ad codes would still be blocked, but any other "non-advert" code would not be blocked.

- In the case there is some wrong blocking of non-advertising code (false positive) you can write to list creators to make an exclusion (more or less general) - these lists are used by ad blockers. One such popular list is <https://easylis.to/>
- However, the best option for ad blocker users is to serve alternative ads for them (such ads that will not be blocked, maybe some affiliate banners). Ad Inserter Pro supports replacement of ads when ad blocking is detected.

In some cases you may get false detection of ad blockers. This is usually because there is some issue with the Javascript code used for ad blocking detection. Check Javascript console for errors or messages. Also make sure detection scripts are present in the page source code:

For Ad Inserter: ads.js, sponsors.js

For Ad Inserter Pro: ads.js, sponsors.js, advertising.js, adverts.js

**Those scripts should be present in the page footer and should not be removed or moved to inline code!**

If you are using plugins to optimize javascript code like **Autooptimize** make sure the scripts listed above are excluded from optimization.

## Are you sure you want to do this?

### The link you followed has expired.

You usually get page with this message when the security token on the settings page expires and you are trying to save the settings. In such case simply refresh the settings page (F5), make changes and save them.

You may also get this message when WordPress encounters some error, for example when uploading fails because of too big file.

## Heavy website load

Ad Inserter is built for speed. In normal cases it should not significantly increase page loading time. Of course, if you are using many code blocks this may increase processing time but still it should not cause any problem. In some cases the cause of long loading time may be processing of your PHP code used in some code blocks. However, it is very easy to determine how much time Ad Inserter uses to insert the code. Log in as admin and go to a page where you would like to check Ad Inserter processing time. Enable Log Processing (Ad Inserter menu item in top admin bar, check Debugging for details). Check page source at the bottom. Among other debugging data you should see the following lines:


```
PLUGIN CODE PROCESSING: 81.55 ms
USER CODE PROCESSING: 210.08 ms
TOTAL PROCESSING TIME: 291.63 ms
```

**PLUGIN CODE PROCESSING** this is time used by the plugin to insert code blocks.

**USER CODE PROCESSING** this is time used to process your PHP code in Ad Inserter code blocks. If you are not using PHP code then this time should be 0.

**TOTAL PROCESSING TIME** this is the total time used by the Ad Inserter plugin to process your PHP code and to insert the code into the page.

## License Expired

Warning: Ad Inserter Pro license expired - updates disabled. Please renew the license to enable updates. 

This message means that the license for Ad Inserter Pro has expired. The plugin will continue to function normally, only the updates will not be available. Go to Plugins management page and click on **Renew License** to go to the page where you can renew or upgrade and renew the license.

## License Overused

Warning: Ad Inserter Pro license overused - updates disabled. Please [upgrade the license](#).

This message means that you are using Ad Inserter Pro on more websites than what you are allowed according to the license type purchased.

You are authorized to install Ad Inserter Pro WordPress plugin purchased here according to the license type purchased. Each license type defines the maximum number of websites on which the Ad Inserter Pro plugin can be installed. If you install the plugin on more websites than allowed by the license type, each website using your license key will show a warning of license over-usage and updates will be disabled. When license over-usage occurs you need to either uninstall the plugin from some websites or upgrade the license type. We also can not guarantee proper functioning of the plugin when the license is overused and we reserve the right to refuse support in such case.

Click on **upgrade the license** to go to the page where you can upgrade the license. After you upgrade it may take **up to 24 hours for the websites to update the plugin status**. Of course, you can also uninstall the plugin from some websites. In such case it may take up to 14 days for the license server to delete websites from the license data.

**Please note** that if you are using [www](#) subdomain and you have [www.example.com](#) and [example.com](#) domains you need to redirect one domain to another (preferably [www](#) to non-[www](#)). Otherwise both domains will behave as independent websites and will consume one license each.

## Caching

---

Keep in mind that just installing a caching plugin does not necessarily make your site faster. Doing a bit of optimization will get you a lot of speed increase without caching plugins. To further combat slowness you may want to re-evaluate the performance of your hosting package. Maybe you've outgrown it. In that case, upgrade to a better hosting solution.

Caching on the frontend side (what visitors see) in some cases does speed up page loading but may cause some unwanted behavior. When you are using caching and visitor visits some page, Wordpress creates that page, Ad Inserter is called to do the job, the created page is sent to the visitor and it is also saved for quicker serving later. The next time the page is visited, the visitor gets cached (saved) page. Because of this some Ad Inserter functions may not work because Ad Inserter is not called when the page is cached:

- Server-side block rotation with [\[ADINSERTER ROTATE\]](#)
- User check
- Server-side device detection
- Server-side referer check
- Server-side cookie check
- Server-side GEO targeting (country detection, Pro only)
- Server-side IP address detection (Pro only)
- Debugging functions

When you need the functions listed above you have to switch off caching (if possible only on selected pages where needed). However, Ad Inserter also supports some of the functions above even when caching is enabled:

- Client-side ad rotation with [\[ADINSERTER ROTATE\]](#)
- Client-side device detection
- Client-side referer check
- Client-side cookie check
- Client-side GEO targeting (country detection, Pro only)
- Client-side IP address detection (Pro only)

For this you need to set **Dynamic blocks** (tab ⚙ / tab General) to **client-side** (code executed in the visitor's browser). Ad Inserter supports two client-side modes for dynamic blocks: **Client-side show** and **Client-side insert**. Both generate Javascript code to run in the browser. **Client-side show** uses CSS to hide unwanted code while **Client-side insert** uses Javascript to insert code when needed or enabled. **Client-side show** is fine with banners as they are all loaded and only shown when needed but some adverts might not work properly or can't be used this way and you might violate their Terms of Service – use **Client-side insert** mode in such case.

**PLEASE NOTE:** some ad codes use old JavaScript method `document.write` which can't be used asynchronously (after the page is loaded and created). **Client-side insert** mode will not work with such ad codes.

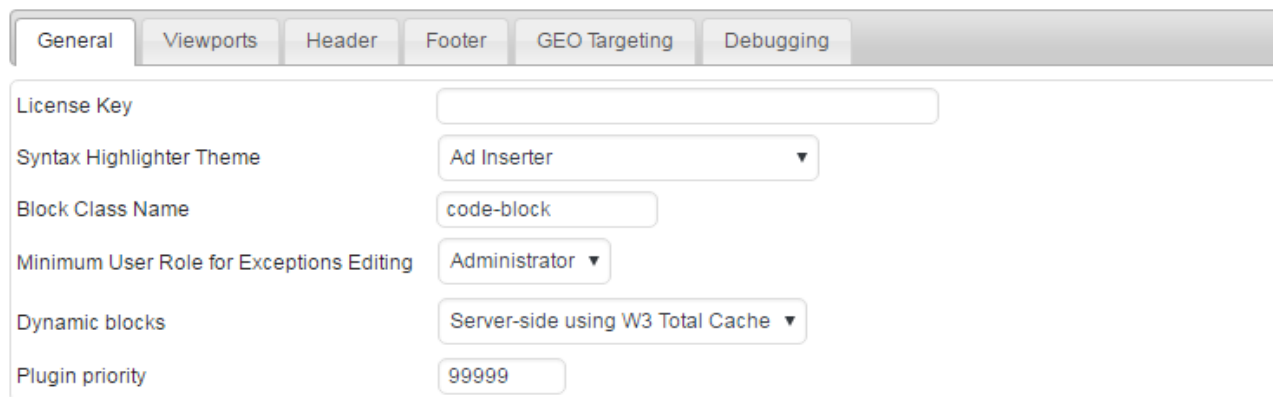
**PLEASE NOTE:** If you are using **W3 Total Cache** for caching then you can still use server-side ad rotation, referer and cookie check, IP address and country detection even when the pages are cached as Ad Inserter supports special features of this caching plugin: **Server-side using W3 Total Cache**. Check section **W3 Total Cache** for details.

**WARNING:** If you are using caching, the inserted code may not appear immediately on the page. Make sure you disable caching when you are testing or debugging. Some caching plugins like [WP Super Cache](#) have an option to disable caching for known users.

**PLEASE NOTE:** There is also an option to **disable caching** on specific pages (location on the misc section tab Insertion - button **Misc**). This option will prevent caching of pages where code blocks with this option enabled are inserted. This option works with WP Super Cache, W3 Total Cache and WP Rocket caching plugins. After you enable this option make sure you flush the cache.

## W3 Total Cache

If you are using **W3 Total Cache** for caching then you can still use server-side ad rotation, IP address and country detection even when the pages are cached as Ad Inserter supports special features of this caching plugin. To enable this mode go to Ad Inserter plugin settings tab \* / tab General and set Dynamic blocks to **Server-side with W3 Total Cache** and configure W3 Total Cache for dynamic content.



The screenshot shows the 'General' tab of the Ad Inserter settings. The 'Dynamic blocks' dropdown menu is set to 'Server-side using W3 Total Cache'. Other settings include: License Key (empty), Syntax Highlighter Theme (Ad Inserter), Block Class Name (code-block), Minimum User Role for Exceptions Editing (Administrator), and Plugin priority (99999).

When Dynamic blocks is set to **Server-side with W3 Total Cache** and you are using ad rotation, IP address or country detection, Ad Inserter inserts short PHP code with special tags in place of code block. When W3 Total Cache plugin loads cached page **it executes this code before it serves the pages**. So the page is still served from the cache but the Ad Inserter blocks are dynamically created just before the page is served. Therefore, pages are served from the cache and the Ad Inserter code is still generated on the server-side.

### Configure W3 Total Cache

How to fully set up W3 Total Cache is beyond the scope of this page. Check the W3 Total Cache manuals or contact its developer for help with that if you need it.

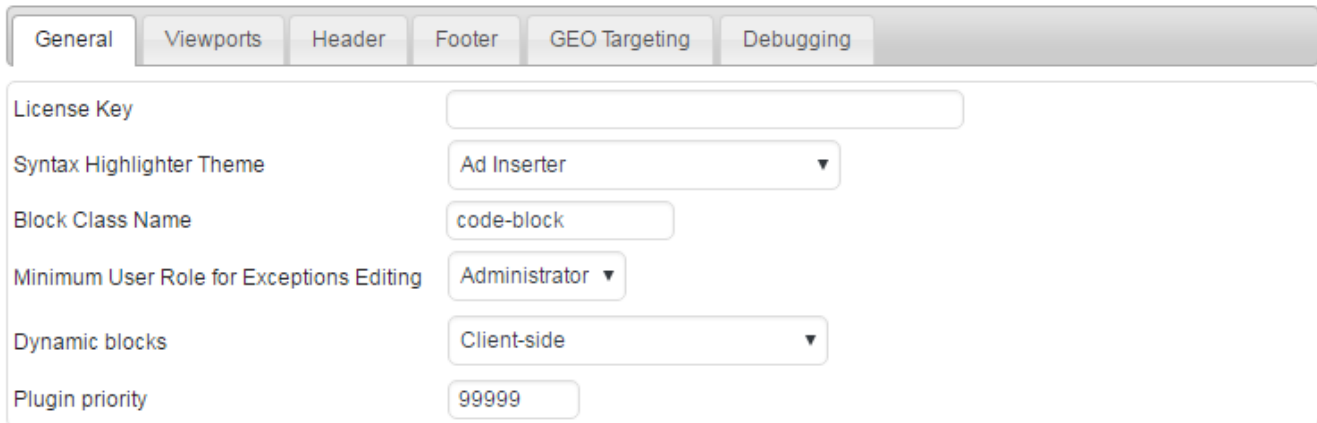
- In **General Settings** set the Page Caching to **Disk: Basic**.
- In **Page Cache** under Advanced enable **Late Initialization**.
- If you use the **Minify** option add `mfunc` to the **ignored comment stem** field.

You need to flush the cache after making these changes. Please note also the following:

- Mixing static cached and dynamic content is a tricky thing to do and often causes all kinds of issues. Ad Inserter follows the guidelines from W3 Total Cache on this. If it doesn't work it's very likely you didn't configure W3 Total Cache correctly. Alternatively, you can decide to use **client-side** option for dynamic blocks – your adverts will show up just fine and rotation and GEO targeting will still work.
- In order to use dynamic content with W3 Total Cache, `W3TC_DYNAMIC_SECURITY` PHP constant needs to be defined. Usually it is located in the `wp-config.php` file. This is a security string used when generating PHP code for dynamic content. If this constant is not defined, Ad Inserter will define and use it.
- Do not use debugging functions when caching is activated.
- When you make changes in settings of the W3 Total Cache **purge all caches** before you check page.
- If you deactivate W3 Total Cache plugin, Ad Inserter will use **Server-side** instead of **Server-side with W3 Total Cache** setting for Dynamic blocks.
- After you activate W3 Total Cache plugin go to **General Settings** and **Save all settings**.

## Ad rotation

To use client-side rotation you have to enable it first. Go to Ad Inserter plugin settings tab \* / tab General and set Dynamic blocks to **Client-side**.



The screenshot shows the Ad Inserter plugin settings interface. The 'General' tab is selected. The 'Dynamic blocks' dropdown menu is set to 'Client-side'. Other settings include: License Key (empty), Syntax Highlighter Theme (Ad Inserter), Block Class Name (code-block), Minimum User Role for Exceptions Editing (Administrator), and Plugin priority (99999).

Configure blocks normally and separate different versions with `[ADINSERTER ROTATE]` shortcode. When the page will be created and Ad Inserter called, it will generate hidden code for all block versions. When the page will be loaded in the browser, one randomly chosen version of the code will be displayed. For example, to rotate 3 images:

```

[ADINSERTER ROTATE]

[ADINSERTER ROTATE]

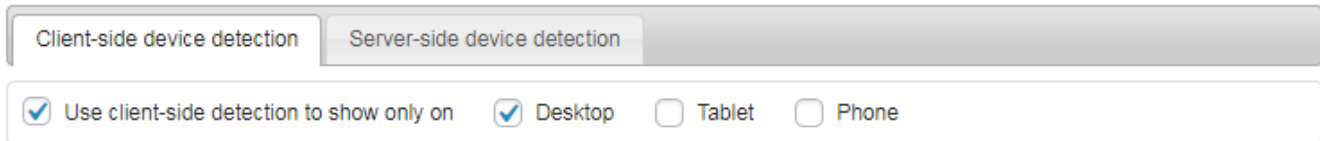
```

Please note the following:

- Client-side rotation works fine with banners, text ads and many ad networks including Amazon Associates. However, it should not be used with some ad networks including AdSense (hiding ad units is **not allowed**) because of the method used for client-side rotation: all options are present on the page but are hidden until the page is loaded and one option is made visible. Please check Terms of Service for your ad network.
- Ad Inserter reserves space for code block based on the height of the first block version. If all the versions have the same height then you won't notice anything. However, if block versions have different heights then it makes sense to define fixed block height with Custom CSS to avoid content jumping when block option with different or initially unknown height is displayed. If the first option is some Javascript based ad (e.g. Amazon widget) you should wrap it with a div and define height, for example `<div style="height: 300px;"> ... </div>`
- Since the code (block option) is displayed only when the page is loaded, you may notice a very short delay (few 100 ms) before the code (or ad) is displayed.
- Client-side rotation uses div elements to wrap the code(s). This is not important unless you use **No wrapping** style.

## Device detection

Ad Inserter supports two types of device detection: **server side** and **client-side**. Each block has its own settings.



The screenshot shows a settings panel for device detection. At the top, there are two tabs: "Client-side device detection" (which is selected) and "Server-side device detection". Below the tabs, there is a checkbox labeled "Use client-side detection to show only on" which is checked. To the right of this checkbox are three radio buttons: "Desktop" (checked), "Tablet" (unchecked), and "Phone" (unchecked).

**Client-side** detection of mobile/desktop devices works always as it is done in visitor's browser. CSS media queries and viewport (browser's screen) width are used to determine device type and show or hide Ad Inserter code blocks.

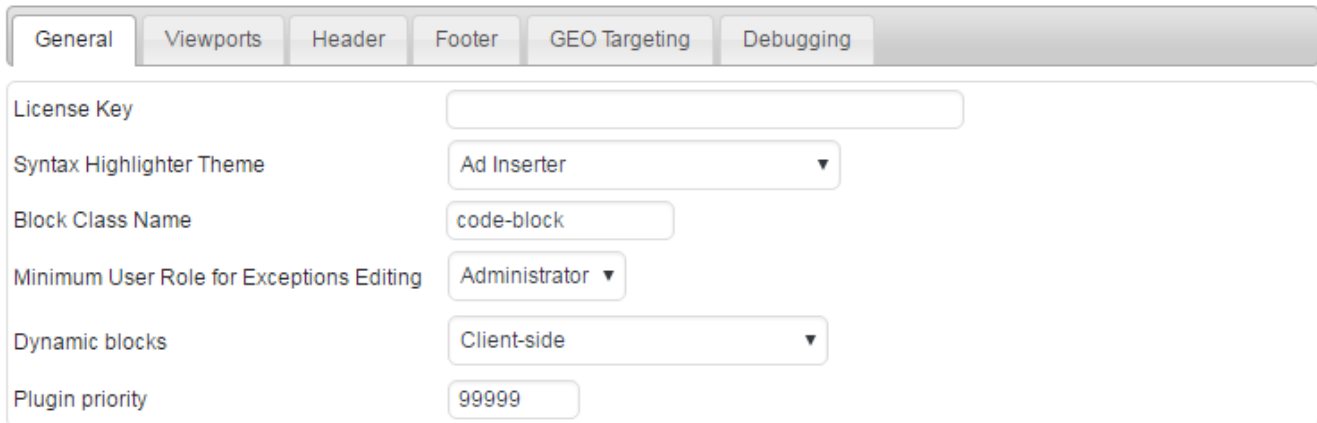
In most cases you should use **ONLY client-side** detection type. Works perfectly with responsive designs as they use CSS media queries.

Please note the following:

- Client-side device detection works fine with all ads. However, it should be used with care as some ad networks like AdSense allow hiding of ad unit only if you're implementing a [responsive ad unit](#). When client-side device detection is used the code block is always inserted on the page but the block is hidden unless it is enabled for the device (more precisely: for the viewport width). This means that the ads are always inserted (loaded and counted) for all devices, but for unwanted devices they are hidden by the browser using CSS media queries based on viewport widths.
- Client-side rotation uses div elements to wrap the code(s). This is not important unless you use **No wrapping** style.
- If you are using **No Wrapping** style and need to hide code on some devices using client-side detection (CSS Media Queries) then you need to add appropriate class to your CSS code ([ai-viewport-1](#), [ai-viewport-2](#), [ai-viewport-3](#)). This doesn't apply to widgets as they always contain a wrapping div.

## IP address and country detection (Pro only)

To use client-side IP address or country detection you have to enable it first. Go to Ad Inserter plugin settings tab \* / tab General and set Dynamic blocks to **Client-side**.



The screenshot shows the Ad Inserter plugin settings interface. At the top, there are tabs for 'General', 'Viewports', 'Header', 'Footer', 'GEO Targeting', and 'Debugging'. The 'GEO Targeting' tab is currently selected. Below the tabs, there are several settings:

- License Key:
- Syntax Highlighter Theme:
- Block Class Name:
- Minimum User Role for Exceptions Editing:
- Dynamic blocks:
- Plugin priority:

Configure blocks normally and define lists of IP addresses or countries. When the page will be created and Ad Inserter Pro called, it will generate hidden block. When the page will be loaded in the browser, IP address and/or country of the visitor will be checked and if allowed the block will be displayed. Check also Button Lists - Black and White Lists.

Please note the following:

- Client-side IP address and country detection works fine, however, there are two modes for Dynamic blocks: **Client-side show** and **Client-side insert**. Both generate Javascript code to run in the browser. **Client-side show** uses CSS to hide unwanted code while **Client-side insert** uses Javascript to insert code when needed or enabled. **Client-side show** is fine with banners as they are all loaded and only shown when needed but some adverts might not work properly or can't be used this way and you might violate their Terms of Service – use **Client-side insert** mode in such case.
- Page needs to access </wp-admin/admin-ajax.php> page on your web server to get IP address and country. Make sure it is accessible and not password protected.
- Since the code block (when allowed) is displayed only when the page is loaded, you may notice a very short delay (few 100 ms) before the code (or ad) is displayed.
- When Ad Inserter Pro (using client-side detection of IP address and/or country) makes the block visible, the content below shifts downwards.
- Client-side IP address and country detection uses div element to wrap the code. This is not important unless you use **No wrapping** style.



# Common Settings

---

## Ad before the first paragraph on all posts

- Automatic Insertion: Before Paragraph
- Posts checked, other page types unchecked
- Paragraph number: 1

Automatic Insertion:  Alignment and Style:

Posts  Homepage  Category pages  
  Static pages  Search pages  Tag / Archive pages

Paragraph number

## Ad before the last paragraph on all posts

- Automatic Insertion: Before Paragraph
- Posts checked, other page types unchecked
- Paragraph number: 1
- Count: From Bottom

Automatic Insertion:  Alignment and Style:

Posts  Homepage  Category pages  
  Static pages  Search pages  Tag / Archive pages

Paragraph number(s)

Count  paragraphs with tags  that have between  and  words  
and   Minimum number of paragraphs   
 Count inside special elements Minimum number of words in paragraphs above

## Contextual Amazon Native Shopping Ads

- Automatic Insertion: After Content
- Posts checked, other page types unchecked

Automatic Insertion: After Content ▼ Alignment and Style: Default ▼ Show

---

Posts ▼  
 Static pages ▼

Homepage  Category pages  
 Search pages  Tag / Archive pages









Code:

```
<script type="text/javascript">
amzn_assoc_placement = "adunit0";
amzn_assoc_search_bar = "true";
amzn_assoc_tracking_id = "ad-inserter-20";
amzn_assoc_search_bar_position = "top";
amzn_assoc_ad_mode = "search";
amzn_assoc_ad_type = "smart";
amzn_assoc_marketplace = "amazon";
amzn_assoc_region = "US";
amzn_assoc_title = "Search Results from Amazon";
amzn_assoc_default_search_phrase = "{smart_tag}";
amzn_assoc_default_category = "All";
amzn_assoc_linkid = "cf1873f027a57f63cede634cfd444bea";
</script>
<script src="//z-na.amazon-adsystem.com/widgets/onejs?MarketPlace=US"></script>
```

Use your own tracking id! Please note `{smart_tag}` as default search phrase. This makes this widget contextual. It will show products related to the tags for this post. You can also try with `{tag}`, `{short_title}`, `{category}` or `{short_category}`. This is how Amazon Native Shopping Ads look like:

Search Results from Amazon

All ▼
blogging
Go

 <p style="font-size: 0.8em;">Blogging: A Practical Guide to Plan Your Blog: Start Your Pr...</p> <p><b>\$0.00</b></p> <p>★★★★☆ (29)</p>	 <p style="font-size: 0.8em;">Blogging: The Ultimate Guide On How To Replace Your ...</p> <p><b>\$0.99</b></p> <p>★★★★☆ (61)</p>	 <p style="font-size: 0.8em;">How To Make Money Blogging: How I Replaced My Day J...</p> <p><b>\$5.38</b> ✓prime</p> <p>★★★★☆ (821)</p>	 <p style="font-size: 0.8em;">BLOGGING GETTING TO \$2,000 A MONTH IN 90 DAYS (Blogg...</p> <p><b>\$0.99</b></p> <p>★★★★☆ (16)</p>
 <p style="font-size: 0.8em;">Blogging: The Best Dam Little Guide To Starting A Profitab...</p> <p><b>\$17.98</b> <del>\$24.00</del> ✓prime</p> <p>★★★★☆ (10)</p>	 <p style="font-size: 0.8em;">Blogging For Dummies</p> <p><b>\$16.90</b> <del>\$24.00</del> ✓prime</p> <p>★★★★☆ (12)</p>	 <p style="font-size: 0.8em;">Blogging: A Practical Guide to Find Your Niche: Start Yo...</p> <p><b>\$0.00</b></p> <p>★★★★☆ (9)</p>	 <p style="font-size: 0.8em;">Blog, Inc.: Blogging for Passion, Profit, and to C...</p> <p><b>\$9.52</b> <del>\$18.06</del> ✓prime</p> <p>★★★★☆ (130)</p>

Ads by Amazon ▶

## Ad wrapped with text of the third paragraph on all posts

- Automatic Insertion: Before Paragraph
- Posts checked, other pages unchecked
- Block Alignment and Style: Float Left
- Paragraph number: 3

## Ad above the first and third post excerpts on the homepage

- Automatic Insertion: Before Excerpt
- Homepage checked, other pages unchecked
- Filter: 1, 3

## Ad between the second, third and fourth comment on posts

- Automatic Insertion: Between Comments
- Posts checked, other pages unchecked
- Filter: 2, 3, 4

## Ad before the second paragraph on all posts, homepage ads in the first three post

- Automatic Insertion: Before Paragraph
- Posts checked, Homepage checked, other pages unchecked
- Paragraph number: 2
- Filter: 1, 2, 3 (in some cases 2, 4, 6 and may need separate block - depends on the theme and needs some testing)

Automatic Insertion: <input type="text" value="Before Paragraph"/>	Alignment and Style: <input type="text" value="Default"/>	<input type="button" value="Show"/>
<input checked="" type="checkbox"/> <input type="text" value=""/> Posts	<input checked="" type="checkbox"/> Homepage	<input type="checkbox"/> Category pages
<input type="checkbox"/> <input type="text" value=""/> Static pages	<input type="checkbox"/> Search pages	<input type="checkbox"/> Tag / Archive pages
Paragraph number(s) <input type="text" value="2"/>	<input type="button" value="Counting"/>	<input type="button" value="Clearance"/>
Post/Static page must have between <input type="text" value=""/> and <input type="text" value=""/> words		
<b>Insertion</b>   Filter   Scheduling   Ad Blocking   General		
Filter insertions <input type="text" value="1, 2, 3"/>	using <input type="text" value="Auto"/>	<input type="checkbox"/> Invert filter

## Centered ad in the middle of post paragraphs

- Automatic Insertion: Before Paragraph
- Posts checked, other pages unchecked
- Block Alignment and Style: Center
- Paragraph number: 0.5

Automatic Insertion:	Before Paragraph ▼	Alignment and Style:	Center ▼	Show	
<input checked="" type="checkbox"/>	▼ Posts	<input type="checkbox"/>	Homepage	<input type="checkbox"/>	Category pages
<input type="checkbox"/>	▼ Static pages	<input type="checkbox"/>	Search pages	<input type="checkbox"/>	Tag / Archive pages
Paragraph number(s)	0.5	Counting	Clearance		

## Ad above post excerpts on the Insurance category page

- Automatic Insertion: Before Post
- Category pages checked, other pages unchecked
- Click Lists button to display lists
- Categories: [insurance](#), White List checked

Automatic Insertion:	Before Post ▼	Alignment and Style:	Default ▼	Show	
<input type="checkbox"/>	▼ Posts	<input type="checkbox"/>	Homepage	<input checked="" type="checkbox"/>	Category pages
<input type="checkbox"/>	▼ Static pages	<input type="checkbox"/>	Search pages	<input type="checkbox"/>	Tag / Archive pages
Categories	insurance	<input type="checkbox"/>	Black List	<input checked="" type="checkbox"/>	White List
Tags		<input checked="" type="checkbox"/>	Black List	<input type="checkbox"/>	White List
Post IDs		<input checked="" type="checkbox"/>	Black List	<input type="checkbox"/>	White List
Urls		<input checked="" type="checkbox"/>	Black List	<input type="checkbox"/>	White List
Url parameters		<input checked="" type="checkbox"/>	Black List	<input type="checkbox"/>	White List
Referers		<input checked="" type="checkbox"/>	Black List	<input type="checkbox"/>	White List

## Ad above the post excerpts on the Cars tag page

- Automatic Insertion: Before Post
- Archive pages checked, other pages unchecked
- Click Lists button to display lists
- Tags: cars, White List checked

Automatic Insertion:	<input type="text" value="Before Post"/>	Alignment and Style:	<input type="text" value="Default"/>	<input type="button" value="Show"/>		
<input type="checkbox"/>	<input type="text"/>	Posts	<input type="checkbox"/>	Homepage	<input type="checkbox"/>	Category pages
<input type="checkbox"/>	<input type="text"/>	Static pages	<input type="checkbox"/>	Search pages	<input checked="" type="checkbox"/>	Tag / Archive pages
Categories	<input type="text"/>	<input checked="" type="radio"/>	Black List	<input type="radio"/>	White List	
Tags	<input type="text" value="cars"/>	<input type="radio"/>	Black List	<input checked="" type="radio"/>	White List	
Post IDs	<input type="text"/>	<input checked="" type="radio"/>	Black List	<input type="radio"/>	White List	
Urls	<input type="text"/>	<input checked="" type="radio"/>	Black List	<input type="radio"/>	White List	
Url parameters	<input type="text"/>	<input checked="" type="radio"/>	Black List	<input type="radio"/>	White List	
Referers	<input type="text"/>	<input checked="" type="radio"/>	Black List	<input type="radio"/>	White List	

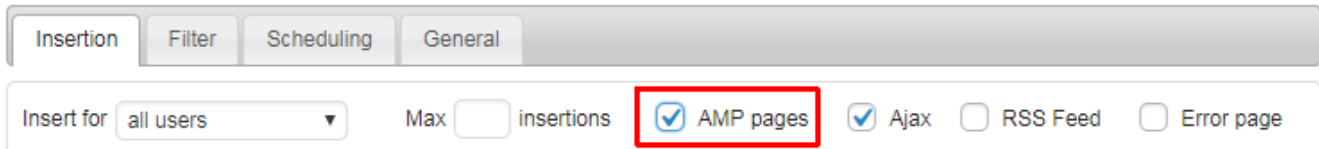
## Settings for ads on AMP pages

The following AMP plugins have been tested with Ad Inserter:

- [AMP for WP – Accelerated Mobile Pages](#) (free)
- [AMP](#) (free)
- [WP AMP Ninja](#) (free)
- [WP AMP - Accelerated Mobile Pages for WordPress](#) (paid)

Normally the code will not be inserted on AMP pages. There are two settings that can be used to insert code on AMP pages:

1. **AMP pages** checkbox in **Misc / Insertion** section - If checked it enables insertion also on AMP pages. **Use this approach only if you need to insert the same code on normal and AMP pages**. To insert different codes on AMP pages (e.g. for AdSense) **leave AMP pages unchecked** and use [\[ADINSERTER AMP\]](#) separator as described below.



The screenshot shows the Ad Inserter settings interface. The 'Insertion' tab is active. The 'AMP pages' checkbox is checked and highlighted with a red box. Other options include 'Ajax', 'RSS Feed', and 'Error page'.

2. Separate the codes with [\[ADINSERTER AMP\]](#) separator - the code above the separator will be inserted on normal pages, the code below the separator will be inserted on AMP pages. This separator can be used also for **Header** and **Footer** code. Below is an example for AdSense ads on normal and AMP pages - check also <https://support.google.com/adsense/answer/7183212?hl=en>

You can use [AdSense code generator](#) to generate AdSense AMP code described below.

```
<script type="text/javascript" src="//pagead2.googlesyndication.com/pagead/js/adsbygoogle.js" async="">
</script>
<!-- Normal page -->
<ins class="adsbygoogle"
style="display: block;"
data-ad-client="ca-pub-3492974262926419"
data-ad-slot="2337275628"
data-ad-format="auto"></ins>
<script type="text/javascript">// 
(adsbygoogle = window.adsbygoogle || []).push({});
// ]&gt;&lt;/script&gt;

[ADINSERTER AMP]

&lt;amp-ad
layout="fixed-height"
height=100
type="adsense"
data-ad-client="ca-pub-3492974262926419"
data-ad-slot="2337275628"&gt;
&lt;/amp-ad&gt;</pre></div><div data-bbox="451 909 571 926" data-label="Page-Footer"><p>AdInserter.Pro</p></div><div data-bbox="831 909 916 927" data-label="Page-Footer"><p>Page 142</p></div>
```

## Generating AdSense code for AMP pages

Ad Inserter has a simple code generator for AdSense ads which can be used also to generate AdSense code for AMP pages.

Paste normal AdSense code into the code window, click on **Code tools** button, then on Code editor button and then click on the **Import code button** to import the AdSense data from it.

**AdSense Post**

Banner AdSense

Publisher ID: pub-3232934728723453  
Ad Slot ID: 2384529483  
Ad Type: Standard  Responsive  
AMP Ad: Disabled

```
<script async src="//pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
<!-- Moje Delo Page Bottom -->
<ins class="adsbygoogle"
  style="display:block"
  data-ad-client="ca-pub-3232934728723453"
  data-ad-slot="2384529483"
  data-ad-format="auto"></ins>
<script>
(adsbygoogle = window.adsbygoogle || []).push({});
</script>
```

Once you have the **Publisher ID** and **Ad Slot ID** imported select ad type for AMP pages. Then click on the generate button and the the plugin will generate Javascript code for the AdSense ad in the code window of the current block.

**AdSense Post**

Publisher ID:

Ad Slot ID:

Ad Type:   Responsive

AMP Ad:

```
<script async src="//pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
<ins class="adsbygoogle"
  style="display: block;"
  data-ad-client="ca-pub-3232934728723453"
  data-ad-slot="2384529483"
  data-ad-format="auto"></ins>
<script>
(adsbygoogle = window.adsbygoogle || []).push({});
</script>
[ADINSERTER AMP]
<amp-ad
  layout="fixed-height"
  height=100
  type="adsense"
  data-ad-client="ca-pub-3232934728723453"
  data-ad-slot="2384529483">
</amp-ad>
```

AdSense ads need also a script in the header.

```
<script async custom-element="amp-ad" src="https://cdn.ampproject.org/v0/amp-ad-0.1.js"></script>
```

This same separator can be used also for **Header** and **Footer** code. The following code will be inserted in header only on AMP pages:

**HTML Page Header Code**

Code in the `<head></head>` section of the HTML page

```
[ADINSERTER AMP]
<script async custom-element="amp-ad" src="https://cdn.ampproject.org/v0/amp-ad-0.1.js"></script>
```



## Support

---

If you have a valid license for Ad Inserter Pro you can ask for help via email. Please use [contact form](#) to send a question or ask for help.

If you experience problems with the free Ad Inserter plugin you can ask for help on the [support forum](#).

However, before you ask for help please use debugging functions described above. In almost all cases it is possible to determine the nature of the problem just by checking the debugging data. Check also source code of the page. Some code for ads may not display anything, either because of errors in the ad code or because of ad network issues. In order to be able to diagnose the problem and suggest actions or fix a bug, please do the following:

1. Enable Remote debugging (located on the Ad Inserter Settings tab ⚙️ / tab Debugging).
2. You may need to disable caching (when instructed) otherwise debugging may not work properly
3. Clearly describe the problem. Describe what does not work as expected.
4. Describe the settings and code blocks used.
5. Provide web addresses (links) of the pages where the code from the settings above is not inserted properly.

Unless you provide the items listed above nobody can check your website, can't reproduce the problem and consequently can't help. Once the problem is fixed you can disable Remote debugging. Thank you very much for understanding.

Ad Inserter is completely free! Please support plugin development if you like it:

If you are using Ad Inserter and you like it, then please write about it and spread the word on the [review page](#).

- Write a nice [review](#)
- [Donate](#)
- [Go Pro](#)