

## Wyzwalacz (inaczej procedura wyzwalana)

Wyzwalacze (trigger) - wykonuje akcję na pewne zdefiniowane zdarzenie, które jest związane z konkretną tabelą i co się z tym wiąże również bazą danych. Wyzwalacze są uruchamiane automatycznie.

Składnia wyzwalacza:

```
CREATE TRIGGER nazwa_wyzwalacza
{BEFORE | AFTER }
{INSERT | UPDATE | DELETE }
ON nazwa_tabeli
FOR EACH ROW
Instrukcja SQL;
```

- nazwa - musi być zgodna z zasadami jakie obowiązują dla wszystkich obiektów dla MySQL - należy korzystać ze znaków alfanumerycznych, podkreślenia, nie używać spacji ani słów kluczowych itp.
- czas - czyli czy wyzwalacz ma być wywołany przed (BEFORE), czy po (AFTER) wykonaniu zdarzenia
- zdarzenie - odpowiada typowi akcji wykonywanej, czyli w momencie dodania rekordu(INSERT), uaktualnienia(UPDATE) lub jego usunięcia(DELETE)
- instrukcja - kod jest podobny do procedur i zawiera się w blokach BEGIN i END
- FOR EACH ROW – określenie wyzwalacza z poziomu wierszy

Identyfikatory :old i :new

Zdefiniowano dwa słowa: OLD oraz NEW, które jasno definiują, o które dane nam chodzi. W przypadku poleceń INSERT możemy używać jedynie słowa NEW, w przypadku poleceń DELETE jedynie słowa OLD, natomiast w przypadku komend UPDATE, zarówno OLD jak i NEW.

- W wyzwalaczu z poziomu wiersza można uzyskać dostęp do danych tego wiersza,
- W tym celu używamy dwóch identyfikatorów korelacji old i new, które są specjalnymi zmiennymi powiązanymi,
- Są one traktowane jako rekord tabeli z którą powiązany jest wyzwalacz,
- Są one nazywane pseudorekordami, bo są one rekordami tylko w składni, nie można przypisywać ich jako całych rekordów.

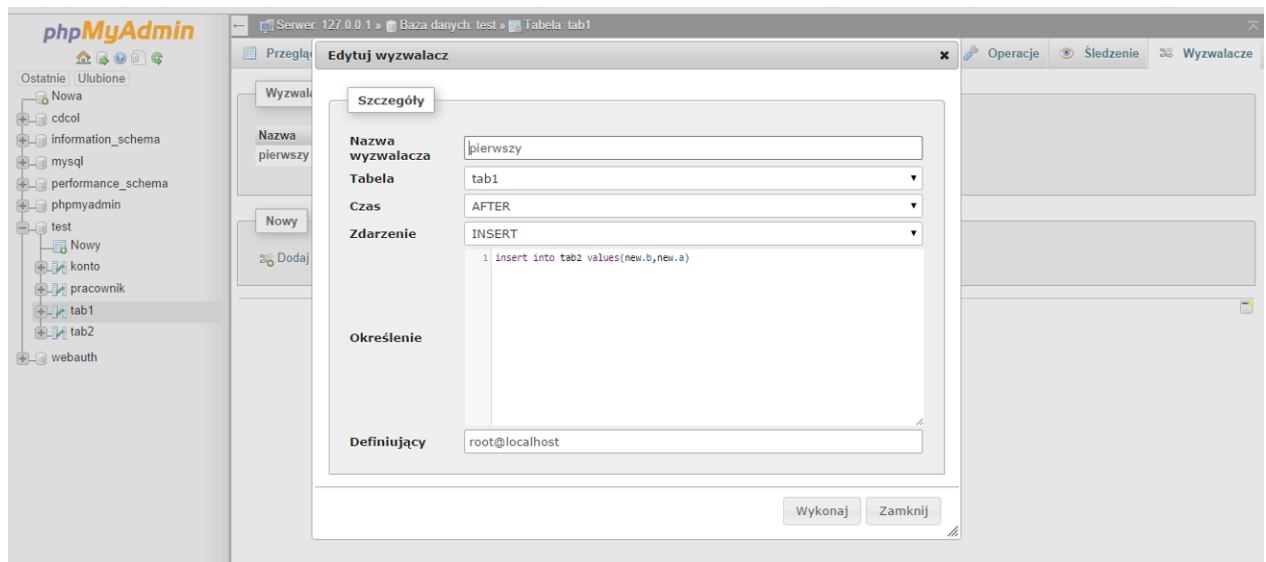
## Zadanie 1

Stwórz tabelki:

```
CREATE TABLE tab1 (a int, b CHAR(10));
CREATE TABLE tab2 (c CHAR(10), d int);
```

Stwórz wyzwalacz, który będzie po wprowadzeniu danych do tabelki tab1 będzie automatycznie wprowadzał te same dane do tabeli tab2.

```
CREATE TRIGGER `pierwszy` AFTER INSERT ON `tab1`
FOR EACH ROW insert into tab2 values(new.b,new.a)
```



## Ćwiczenie 1

Stwórz tabelki:

test1(a1 INT);

test4( a4 INT, b4 INT);

Stwórz wyzwalacz, który po wprowadzeniu wartości do tabelki test1 w tabelce test4 zmodyfikuje  $b4 = b4 + 10$  jeżeli a4 będzie równe wprowadzanej wartości.

Aby wprowadzić działanie wyzwalacza wprowadź do tabelki test4 wartości:

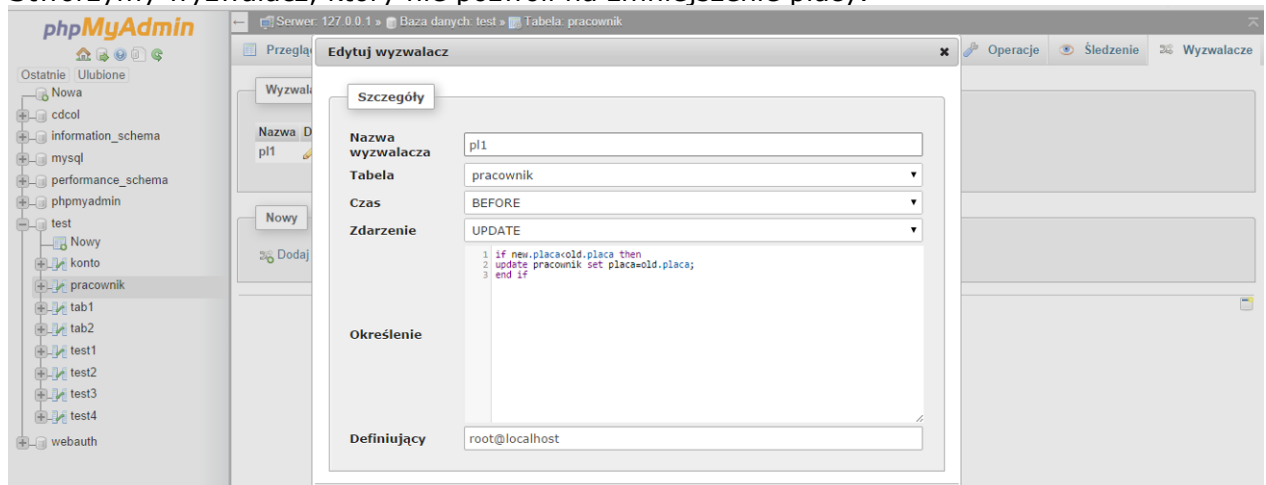
| a4 | b4 |
|----|----|
| 1  | 10 |
| 2  | 20 |
| 3  | 30 |

A teraz wprowadź do tabelki test1 wartość 1, sprawdź czy w tabelce test4 zmieniła się wartość b4 dla rekordu pierwszego.

## Zadanie 2

Stwórz tabelkę pracownicy(id, imie, nazwisko, placa). Wprowadź dowolne 3 rekordy.

Stworzymy wyzwalacz, który nie pozwoli na zmniejszenie płacy.



## Zadanie 3

Stwórz tabelkę towar(id, nazwa, cena), wprowadź 3 rekordy.

Stwórz tabelkę towar\_arch(id, nazwa, cena, data).

Stwórz wyzwalacz, który będzie wprowadzał stare dane oraz dzisiejszą datę do tabelki towar\_arch, jeżeli w tabelce towar zostaną dane zmodyfikowane. Sprawdź działanie wyzwalacza.

```
CREATE TRIGGER `wyz`  
BEFORE UPDATE ON `towar`  
FOR EACH ROW  
    insert into towar_arch values(old.id, old.nazwa, old.cena, sysdate())
```

#### Zadanie 4

Stwórz tabelkę konto(nr, saldo), wprowadź 3 rekordy.

Zdefiniujemy trigger, który będzie sprawdzał wprowadzane zmiany w tablicy. Jeśli ktoś wykona polecenie UPDATE na danych w tablicy zdefiniowanej poniżej, wprowadzając dowolne zmiany, wówczas za każdym razem (dla każdego wiersza FOR EACH ROW) nowa wartość będzie sprawdzana i dopiero kiedy pomyślnie przejdzie testy zostanie zaakceptowana, w przeciwnym wypadku zostanie wprowadzona wartość z zakresu od 0 do 100.

```
CREATE TRIGGER `zmiana` BEFORE UPDATE ON `konto`  
FOR EACH ROW  
IF NEW.saldo < 0 THEN  
    SET NEW.saldo = 0;  
ELSEIF NEW.saldo > 100 THEN  
    SET NEW.saldo = 100;  
END IF
```

Sprawdź działanie wyzwalacza.

#### Zadanie 5

Stwórz tabelę zamówienie(id, produkt, data\_zamowienia). Wprowadź 3 rekordy. Stworzymy wyzwalacz, który pozwoli zdefiniować dla pola data datę aktualną.

```
CREATE TRIGGER ustaw_date  
BEFORE INSERT  
ON zamówienie  
FOR EACH ROW  
BEGIN  
SET new.data_zamowienia = now();  
END;
```

#### Zadanie 6

Stwórz tabelę rachunki\_szczegolowe, w której umieszczysz pola (id\_klienta, woda, gaz, prad). Wprowadz do tej tabeli przykładowe dane.

Stwórz tabelę rachunki\_podsumowanie, w której umieszczysz pola (id\_klienta, suma)

Następnie stwórz trigger, który po wpisaniu dowolnych danych do tabeli rachunki\_szczegolowe generuje nowy wiersz w tabeli rachunki\_podsumowanie, gdzie suma=woda + gaz + prad

```
CREATE TRIGGER `sumator` AFTER INSERT ON `rachunki_szczegolowe`  
FOR EACH  
ROW INSERT INTO rachunki_podsumowanie  
VALUES (  
new.id, new.woda + new.gaz + new.prad  
);
```

#### Zadanie 7

Stwórz tabelę pensje\_pracownikow (id, pensja) oraz podwyzki (id, kwota)

Do tych tabel wprowadz dane, tak, aby tylko niektórzy pracownicy otrzymali podwyżki.

Stworz trigger, który po wpisaniu kwoty podwyżki dowolnemu pracownikowi będzie powodował aktualizację danych w tabeli pensje\_pracownikow tak, że pensja będzie automatycznie się zwiększała o kwotę podwyżki

```
CREATE TRIGGER `premiator` AFTER INSERT ON `podwyzki`  
FOR EACH  
ROW UPDATE pensje_pracownikow SET pensja = pensja + new.kwota WHERE id =  
new.id
```

### Zadanie 8

Stwórz tabelę pracownicy, w której wpiszesz umieszczysz pola: id, imię, nazwisko, płeć. Utwórz trigger, który jeśli zostanie złe wprowadzona płeć dla kobiety, to zmieni ją automatycznie.

```
CREATE TRIGGER `plcio_rozpoznawacz` BEFORE INSERT ON `pracownicy`  
FOR EACH  
ROW BEGIN  
IF new.imie LIKE "%a"  
THEN  
SET new.plec = "K";  
END IF;
```

### Ćwiczenie 2

Zaprojektuj bazę danych biblioteka, składającą się z tabel:

Osoba (id, imię, nazwisko, wiek)

Książka (nr, tytuł, autor)

Wypożyczenia (nr, id\_osoby, nr\_książki, data\_wyp, data\_odd)

Wypożyczenie\_archiwum (nr, id\_osoby, nr\_książki, data\_odd)

Dodatkowo zaprojektuj wyzwalacze:

Na tabelce Osoba wyzwalacz, który pozwoli stworzyć wiek w przedziale (10,90)

Na tabelce Wypożyczenia wyzwalacz, który po zmodyfikowaniu pola data\_odd będzie archiwizował dane do tabelki Wypożyczenie\_archiwum