

#InfyTQ Python Previous Year Slots Questions: - PART-3

Final Preparation Series

Q1.

Consider the Code Given Below:

```
class PoliceMan:
    __counter = 20
    def __init__(self,name):
        self.name = name
        self.dept = None

police_man1 = PoliceMan('Roger')           #Line1
police_man2 = PoliceMan('Gabriel')
police_man3 = police_man2
police_man2 = police_man1
police_man3 = police_man2                 #Line2
```

How many reference variable(s) refer to the object created in Line1 at the end of Line2?

Online Learning

Options: -

1. 3

2. 0

3. 2

4. 1

Q2.

Which of the following Function signatures are INCORRECT?

```
def func1(arg1, *arg2, *arg3):  
    pass
```

```
def func2(arg4=10, arg5):  
    pass
```

```
def func3(arg6,arg7,arg8=3):  
    pass
```

Options: -

1. func1 only

2. func2 only

This content is made by online learning team don't use without permission

3. func2 and func3 only

4. func1 and fun2 only

Q3.

What is the Output of the code given below?

```
input_list=[0,2,1,3,0]
val=0
while(len(input_list)!=1):
    input_list.pop(1)
    input_list.remove(input_list[0])
    input_list.append(val+1)
    val+=1

print(input_list)
```

Options: -

1.[4]

2.[3]

3.[1]

4. 0

Q4.

What is the Output of this Code given below?

```
def retrieve_num(number, value):  
    if(number<(value//2)):  
        return 1  
    elif(number%2==0):  
        return retrieve_num(number-1, value-1)  
    else:  
        return (value-1)*retrieve_num(number-1, value-1)  
  
print(retrieve_num(7,4))
```

Options: -

1. 315
 2. Runtime Error
 3. 0
 4. -1
-

Q5.

The “Three-Utilities Problem” is well known mathematical puzzle which is as follows:

“There are three houses on a planar surface and three utilities – gas, water & electricity. Each House needs to be connected to each of the three utilities.”

Identify the most suitable data structure to model the “Three-Utilities Problem”?

Options: -

1. Stack
 2. Graph
 3. Tree
 4. Hash Table
-

Q6.

Consider the python code below.

```
class Employee(metaclass=ABCMeta):
    def __init__(self,emp_id):
        self.__emp_id=emp_id
    @abstractmethod
    def monthly_salary(self):
        pass

class Trainee(Employee):
    def __init__(self,emp_id,batch_name):
        super().__init__(emp_id)
        self.__batch_name=batch_name
```

Which of the following statement(s) is/are TRUE?

- (i). Class Employee cannot be instantiated as it is an abstract class
- (ii). Class Trainee cannot be instantiated as it does not implement monthly_salary () method

Online Learning

Options: -

1. Only (i) is True
 2. Only (ii) is True
 3. Both (i) & (ii) are True
 4. Neither (i) nor (ii) is True
-

Q7.

What is the output of the Given Code Below?

```
values = ["823", "863"]
num = values[0][0:]
for row in range(0, len(values)):
    for column in range(0, len(values[row])):
        if num > values[row][column:]:
            num = values[row][column:]

print(num)
```

Options: -

1. 2

2. 3

3. 23

4. 323

Q8.

What is the Output of given below code?

```
data = 50
try:
    data = data/10
except ZeroDivisionError:
    print('Cannot divide by 0 ', end = '')
finally:
    print('Infy', end = '')
else:
    print('Division successful ', end = '')
```

Online Learning

Options:-

1. Syntax Error
2. Division Successful
3. Infy
4. Both 2 & 3.

