# Assignment 1 Solutions

Muhammad Turab

18CS45

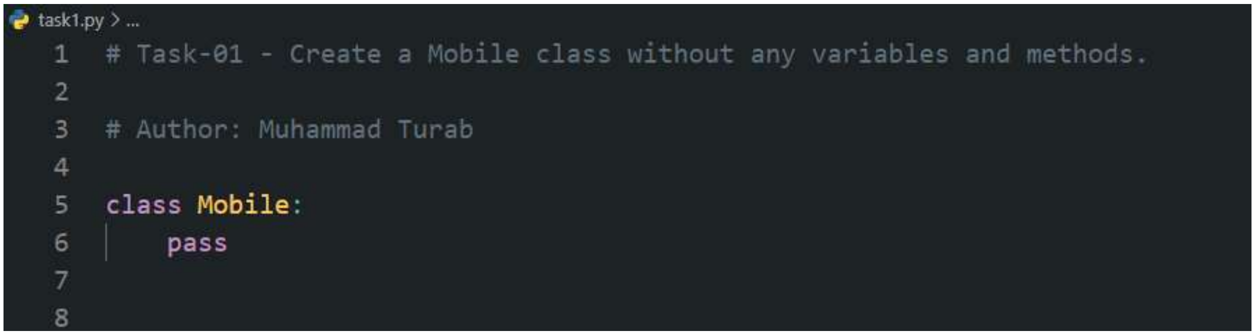
Muhammad Turab

Task-01 - Create a Mobile class without any variables and methods.

Code:

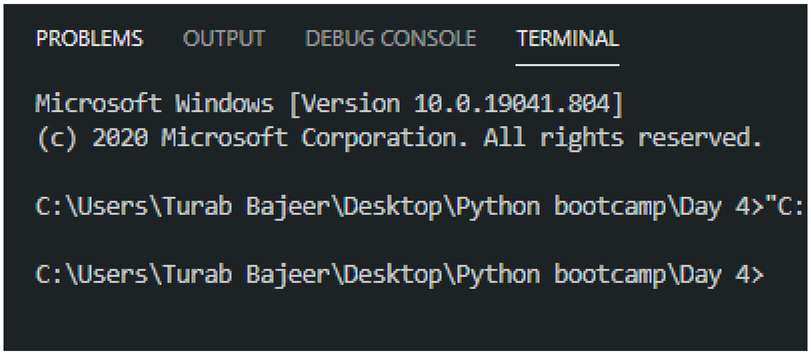
I didn’t write the constructor because as the question says without any variables and methods. So, constructor is basically a method

It was giving error so I wrote the pass keyword.



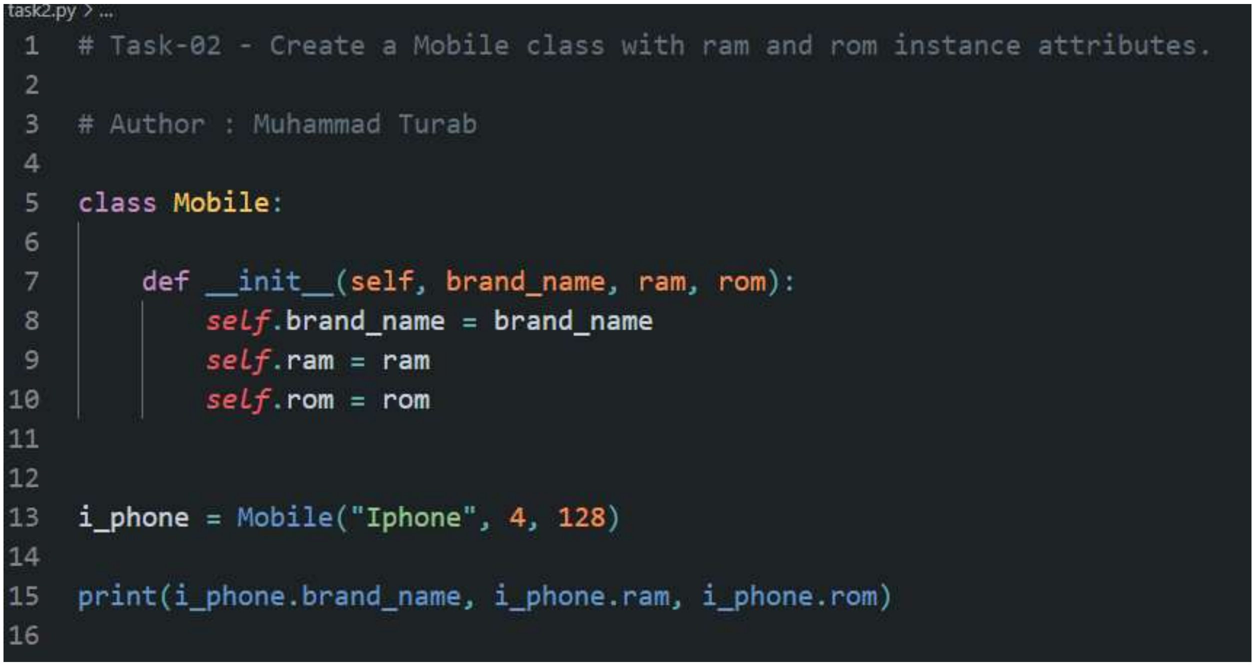
```python
# Task-01 - Create a Mobile class without any variables and methods.

# Author: Muhammad Turab

class Mobile:
    pass

```

Output:



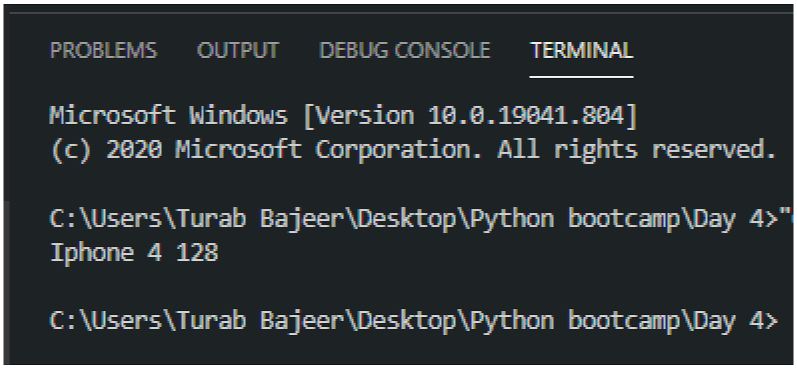
```
Microsoft Windows [Version 10.0.19041.804]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Turab Bajeer\Desktop\Python bootcamp\Day 4>"C:

C:\Users\Turab Bajeer\Desktop\Python bootcamp\Day 4>
```

Task-02 - Create a Mobile class with ram and rom instance attributes.



```python
# Task-02 - Create a Mobile class with ram and rom instance attributes.

# Author : Muhammad Turab

class Mobile:

    def __init__(self, brand_name, ram, rom):
        self.brand_name = brand_name
        self.ram = ram
        self.rom = rom


i_phone = Mobile("Iphone", 4, 128)

print(i_phone.brand_name, i_phone.ram, i_phone.rom)

```

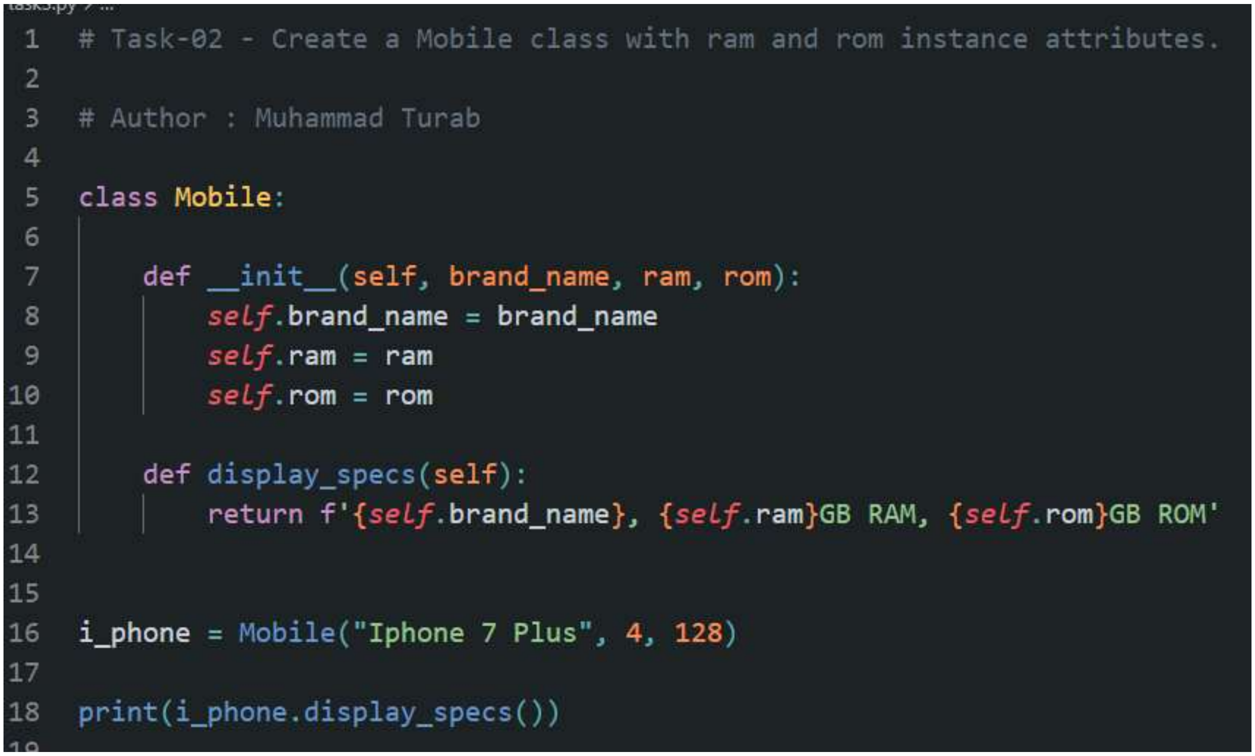
Output:



```
Microsoft Windows [Version 10.0.19041.804]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Turab Bajeer\Desktop\Python bootcamp\Day 4>"
Iphone 4 128

C:\Users\Turab Bajeer\Desktop\Python bootcamp\Day 4>
```

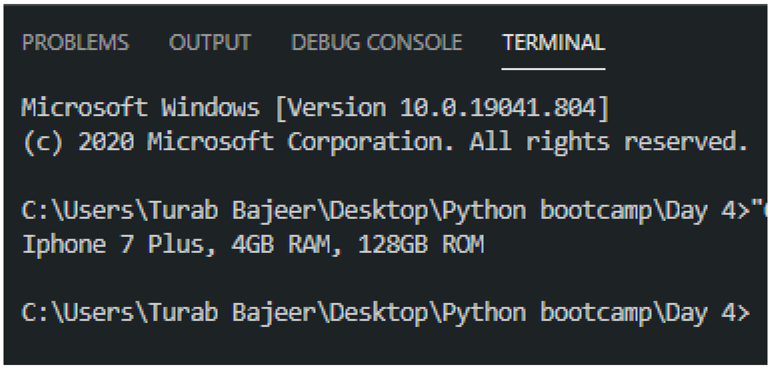
Task-03 - Using solution of (2) add a method display_specs to the class Mobile

Code:



```python
# Task-02 - Create a Mobile class with ram and rom instance attributes.

# Author : Muhammad Turab

class Mobile:

    def __init__(self, brand_name, ram, rom):
        self.brand_name = brand_name
        self.ram = ram
        self.rom = rom

    def display_specs(self):
        return f'{self.brand_name}, {self.ram}GB RAM, {self.rom}GB ROM'


i_phone = Mobile("Iphone 7 Plus", 4, 128)

print(i_phone.display_specs())
```

Output:



```
Microsoft Windows [Version 10.0.19041.804]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Turab Bajeer\Desktop\Python bootcamp\Day 4>"
Iphone 7 Plus, 4GB RAM, 128GB ROM

C:\Users\Turab Bajeer\Desktop\Python bootcamp\Day 4>
```

Task-04 Difference between Instance and Object.

Explanation:

The basic concept of Object Oriented Programming (OOP) revolves around two things, Class, and Object. The class is the blueprint. The Object is an actual thing that is made up using that 'blueprint' (like the car example given above). You cannot see the instances, all you see is code, which is your class. Object or instance are created at run-time and they are created in a specific memory area called heap memory.

Each instance consumes some memory depending upon how much and what value you store. For example "Java" is an instance of String class and holds memory required to represent those characters and to store some metadata. You might have also heard about class method vs instance methods, right? look we hardly call object method.

There is no harm calling instance as an object but if you are following rest of Java convention then why not this one. Do you call Java method function? No right, then there is no point calling instance as an object, it will just create confusion nothing more.

If you are senior Java developer or a trainer then it's your responsibility to pass right terminology to junior developers. Whatever they will hear from you, it will go a long way, so make sure you feed clear and concise information.