Name: Muhammad Abdullah

Roll no: 016

Section: 3A

Department: BSDSM

Task #1: Artificial Intelligence Lab

Q#1 Create a class User that keeps track of how many user objects have been created.

Requirements:

● Use a class variable total_users

● Use a class method get_total_users() to return the total number of users

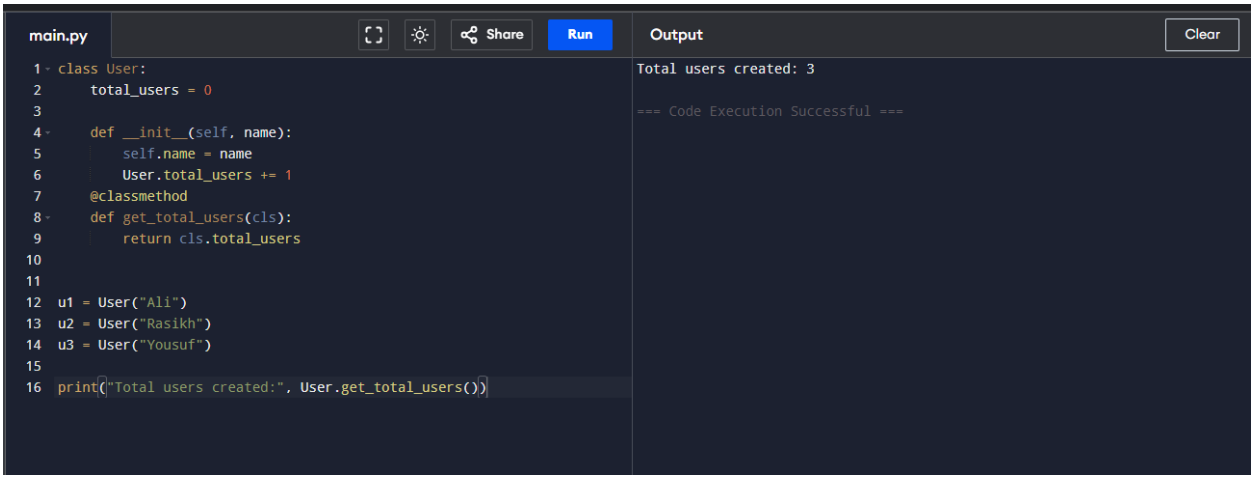
● Increment the counter every time a new object is created

Task:

Write a program that:

1. Creates at least 3 user objects

2. Prints the total number of users using the class method



Q2. Create a class Product with:

● A class variable tax_rate = 0.15

Requirements:

● Use a class method update_tax_rate(new_rate) to update the

tax rate

● Use a static method calculate_tax(price) that returns the tax for

a given price

● Demonstrate that changing the tax rate affects all products

main.py Share Run Output

```python
class Product:
    tax_rate = 0.15
    def __init__(self, name, price):
        self.name = name
        self.price = price

    @classmethod
    def update_tax_rate(cls, new_rate):
        cls.tax_rate = new_rate
    @staticmethod
    def calculate_tax(price):
        return price * Product.tax_rate
p1 = Product("Laptop", 1000)
p2 = Product("Phone", 500)

print("Old Tax Rate:", Product.tax_rate)
print("Tax on Laptop:", Product.calculate_tax(p1.price))
print("Tax on Phone:", Product.calculate_tax(p2.price))

Product.update_tax_rate(0.20)

print("\nNew Tax Rate:", Product.tax_rate)
print("Tax on Laptop:", Product.calculate_tax(p1.price))
print("Tax on Phone:", Product.calculate_tax(p2.price))
```

```
Old Tax Rate: 0.15
Tax on Laptop: 150.0
Tax on Phone: 75.0

New Tax Rate: 0.2
Tax on Laptop: 200.0
Tax on Phone: 100.0

=== Code Execution Successful ===
```

Q3. Create a class Validator.

Requirements:

● Write a static method is_valid_email(email) that:

○ Returns True if the email contains @ and .

○ Otherwise returns False

● Do NOT use any instance or class variables

Task:

Test the method with at least 3 email addresses.

main.py Share Run Output

```python
class Validator:

    @staticmethod
    def is_valid_email(email):
        if "@" in email and "." in email:
            return True
        else:
            return False

print(Validator.is_valid_email("rasikh@gmail.com"))
print(Validator.is_valid_email("ali@yahoo"))
print(Validator.is_valid_email("jefferyepstein@domain.org"))
```

```
True
False
True

=== Code Execution Successful ===
```