Vidyavardhini’s College of Engineering & Technology

Department of Computer Engineering

Experiment No. 4
Creating functions, classes and objects using python
Date of Performance:
Date of Submission:

Experiment No. 4

Title: Creating functions, classes and objects using python

Aim: To study and create functions, classes and objects using python

Objective: To introduce functions, classes and objects in python

Theory:

A function is a block of code which only runs when it is called.

You can pass data, known as parameters, into a function.

A function can return data as a result.

A class is a user-defined blueprint or prototype from which objects are created. Classes provide a means of bundling data and functionality together. Creating a new class creates a new type of object, allowing new instances of that type to be made. Each class instance can have attributes attached to it for maintaining its state. Class instances can also have methods (defined by their class) for modifying their state.

To understand the need for creating a class let's consider an example, let's say you wanted to track the number of dogs that may have different attributes like breed, age. If a list is used, the first element could be the dog's breed while the second element could represent its age. Let's suppose there are 100 different dogs, then how would you know which element is supposed to be which? What if you wanted to add other properties to these dogs? This lacks organization and it's the exact need for classes.

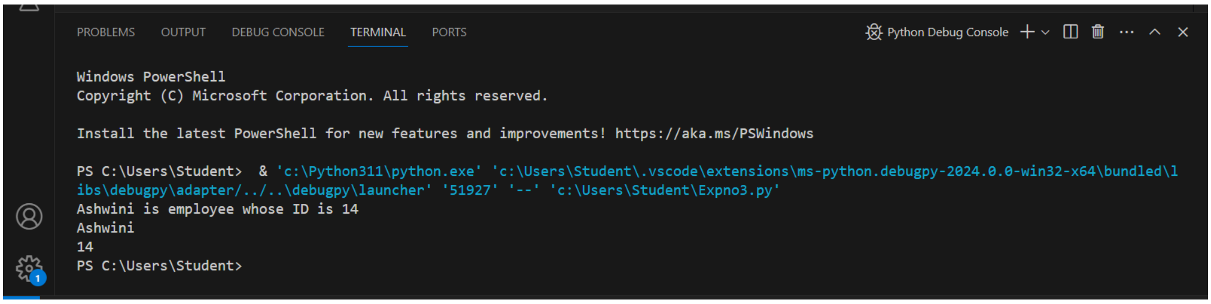
Class creates a user-defined data structure, which holds its own data members and member functions, which can be accessed and used by creating an instance of that class. A class is like a blueprint for an object.

Code:

1)class Employee:

```
    def __init__(self,name,ID):
      self.name = name
      self.ID = ID
    def show(self):
        print(f"{self.name} is employee whose ID is {self.ID}")
e1=Employee("Ashwini",14)
e1.show()
print(e1.name)
print(e1.ID)
```

Output:



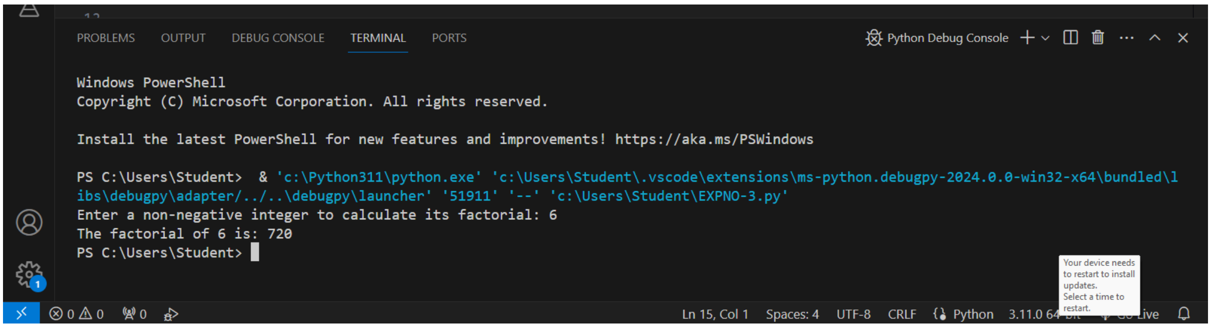
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Student>  & 'c:\Python311\python.exe' 'c:\Users\Student\.vscode\extensions\ms-python.debugpy-2024.0.0-win32-x64\bundled\l
ibs\debugpy\adapter/../..\debugpy\launcher' '51927' '--' 'c:\Users\Student\Expno3.py'
Ashwini is employee whose ID is 14
Ashwini
14
PS C:\Users\Student>
```

Code:

```
2) def factorial(n):
    if n < 0:
        return "Factorial is not defined for negative numbers"
    elif n == 0:
        return 1
    else:
```

```
    fact = 1
    for i in range(1, n + 1):
        fact *= i
    return fact
num = int(input("Enter a non-negative integer to calculate its factorial: "))
result = factorial(num)
print(f"The factorial of {num} is: {result}")
```

Output:



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Student>  & 'c:\Python311\python.exe' 'c:\Users\Student\.vscode\extensions\ms-python.debugpy-2024.0.0-win32-x64\bundled\l
ibs\debugpy\adapter/../..\debugpy\launcher' '51911' '--' 'c:\Users\Student\EXPNO-3.py'
Enter a non-negative integer to calculate its factorial: 6
The factorial of 6 is: 720
PS C:\Users\Student> 
```

Conclusion:

Classes object and functions have been implemented.