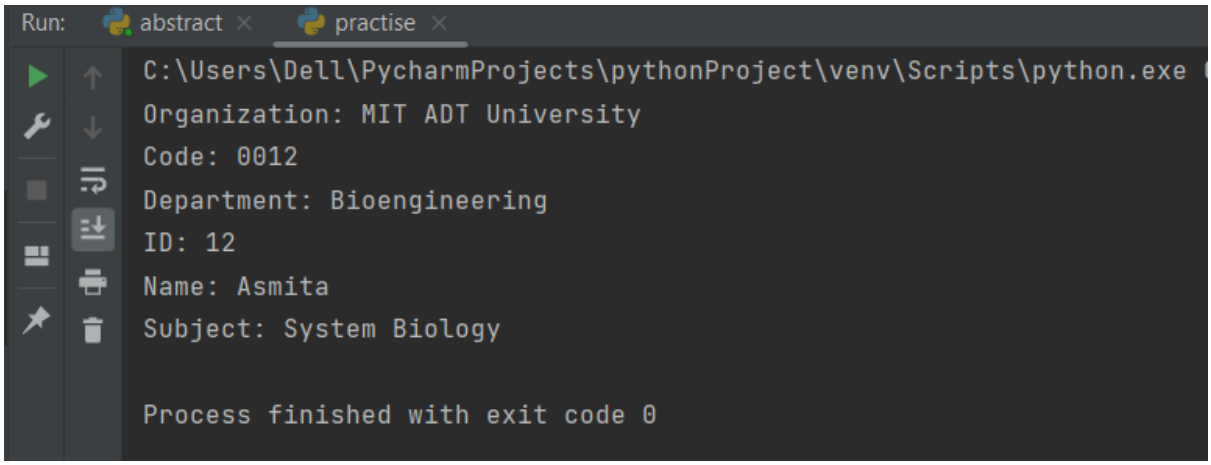
Practical No:1

Asmita Jagadale

MITU19IMBI0012

Inheritance:

```
class Person:
    def set_data(self, org, code):
        self.org = org
        self.code = code

class Department(Person):
    def get_data(self,dept,id):
        self.dept = dept
        self.id = id

class Student(Department):
    def set_name(self,name,sub):
        self.name = name
        self.sub = sub

    def display(self):
        print(f'Organization: {self.org} \nCode: {self.code} \nDepartment:
{self.dept} \nID: {self.id} \nName: {self.name} \nSubject: {self.sub}')

obj = Student()
obj.set_data("MIT ADT University","0012")
obj.get_data("Bioengineering","12")
obj.set_name("Asmita", "System Biology")
obj.display()
```

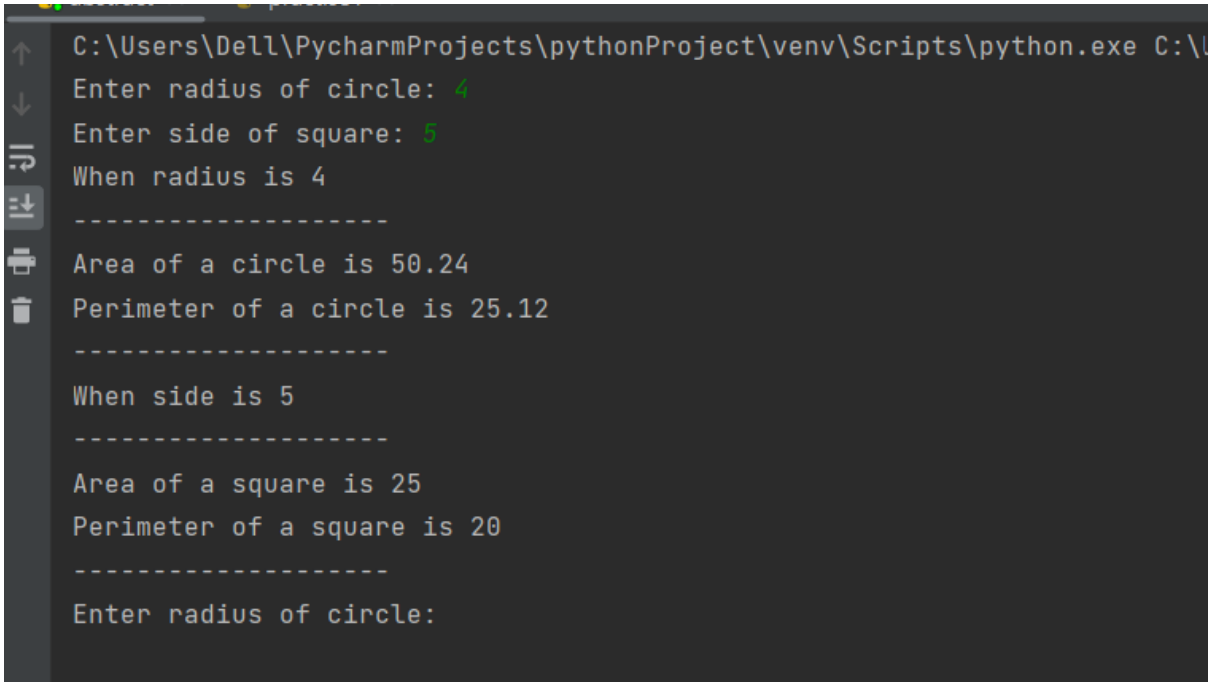


```python
class Student():
    def set_name(self,name,id):
        self.name = name
        self.id = id

class Sports(Student):
    def set_data(self,sports,marks):
        self.sports = sports
        self.marks = marks

class Test(Student):
    def get_name(self,sub,score):
        self.sub = sub
        self.score = score

class Result(Sports,Test):
    def get_data(self,code, dept):
        self.code = code
        self.dept = dept

    def display(self):
        print(f'Student name: {self.name} \nStudent Id: {self.id}')
        print(f'Sports marks: {self.sports} - {self.marks}')
        print(f'Test marks: {self.sub} - {self.score}')
        print(f'Department: {self.dept}')
        print(f'Code: {self.code}')

obj = Result()
obj.get_data("0012", "Bioengineering")
obj.get_name("Python","25/40")
obj.set_data("Cricket","35/40")
obj.set_name("Joe","01")
obj.display()
```

```
C:\Users\Dell\PycharmProjects\pythonProject\venv\Scripts\p
Student name: Joe
Student Id: 01
Sports marks: Cricket - 35/40
Test marks: Python - 25/40
Department: Bioengineering
Code: 0012

Process finished with exit code 0
```

Abstraction:

```
from abc import ABC, abstractmethod
class Shape(ABC):
    @abstractmethod
    def area(self):
        pass
    def perimeter(self):
        pass
class Circle(Shape):
    def __init__(self, radius):
        self.radius = radius
    def area(self):
        return 3.14 * self.radius ** 2
    def perimeter(self):
        return 2 * 3.14 * self.radius
class Square(Shape):
    def __init__(self, side):
        self.side = side
    def area(self):
        return self.side ** 2
    def perimeter(self):
        return self.side * 4
# Initializing the loop
for i in range(0, 100):
    rad = int(input("Enter radius of circle: "))
    s = int(input("Enter side of square: "))
    circle = Circle(rad)
    square = Square(s)
    print("When radius is", rad)
    print("--------------------")
    print(f"Area of a circle is {circle.area()}")
    print(f"Perimeter of a circle is {circle.perimeter()}")
    print("--------------------")
    print("When side is", s)
    print("--------------------")
    print(f"Area of a square is {square.area()}")
    print(f"Perimeter of a square is {square.perimeter()}")
    print("--------------------")
```


C:\Users\Dell\PycharmProjects\pythonProject\venv\Scripts\python.exe C:\
Enter radius of circle: 4
Enter side of square: 5
When radius is 4

Area of a circle is 50.24
Perimeter of a circle is 25.12

When side is 5

Area of a square is 25
Perimeter of a square is 20

Enter radius of circle: