1. Analyse the codes below and write the reason behind the error.

(a)

```
class A:
    def __init__(self):
        self.P=10
        self.__Q=20
    def getY(self):
        return self.__Q
a=A()
print(a.P)
print(a.__Q)
```

(b)

```
class A:
    def __init__(self,P):
        self.P=P
    def print(self):
        print(self.P)
a=A()
a.print()
```

(c) Write down the output:

```
class A:
    def __init__(self,P):
        self.P=P
    def print(self):
        print(self.P)
a=A("Welcome")
a.print()
```

2. Write a program to create a class named Demo. Define two methods Get_String() and Print_String(). Accept the string from the user and print it in upper case.
3. Create a class Circle. Use constructor to read an attribute radius and use proper functions to return the radius and calculate the area of circle.
4. Implement amrita grading system using inheritance.
5. Implement hierarchical inheritance based on the following details:
   Create class Shape. Derive two sub classes Rectangle and Triangle. Define the required attributes for each to calculate the area using constructors. Find the area of each using functions and objects.
6. Try the programs in your tutorial PPT
7. A class named *Computer* has the attributes *model and speed*. It derives another class *LaptopComputer*, which have an extra attribute *weight*. Derive the following output:

   NoteBook Pro15, 1500 MHz, 2 kg

8. Consider the class definitions for a ComputerGame and a GameWarehouse. A GameWarehouse object is used to store ComputerGame objects. Please familiarize yourself with these classes. Then define a new class named GameMuseum which inherits the GameWarehouse class. The GameMuseum class should *override* the list_games() method, so that it returns a list of only those games which were made before the year 1990. The new class should also have a constructor which *calls the constructor from the parent class GameWarehouse*. The constructor takes no arguments.

   You may use the following code to test your implementation:

```
museum = GameMuseum()
museum.add_game(ComputerGame("Pacman", "Namco", 1980))
museum.add_game(ComputerGame("GTA 2", "Rockstar", 1999))
museum.add_game(ComputerGame("Bubble Bobble", "Taito",1986))
for game in museum.list_games():
        print(game.name)
```

Sample output :

Pacman Bubble Bobble

9. Construct a class hierarchy for bank accounts and print the transaction details in proper format.
10. Construct a class hierarchy for people on a college. Include faculty, staff and students. What do they have in common? What distinguishes from one another? Define proper attributes for each classes and implement the various type of inheritance.