

Practise Questions 2

Topic: OOPs

1. Determine if School_bus is also an instance of the Vehicle class

```
class Vehicle:
    def __init__(self, name, mileage, capacity):
        self.name = name
        self.mileage = mileage
        self.capacity = capacity

class Bus(Vehicle):
    pass

School_bus = Bus("School Volvo", 12, 50)
```

2. Create a class Person with attributes: name and age.
 - a. Create a **display()** method that displays the name and age of an object created via the Person class.Create a child class Student which inherits from the Person class and which has a name, age and section attribute.
 - a. Create a method **displayStudent()** that displays the name, age and section of an object created via the Student class.

3. Create a Bus child class that inherits from the Vehicle class. The default fare charge of any vehicle is seating capacity * 100. If Vehicle is Bus instance, we need to add an extra 10% on full fare as a maintenance charge. So total fare for bus instances will become the final amount = total fare + 10% of the total fare.

Use the following code for your parent Vehicle class. You need to override the fare() method of a Vehicle class in Bus class.

```
class Vehicle:
    def __init__(self, name, mileage, capacity):
        self.name = name
        self.mileage = mileage
        self.capacity = capacity

    def fare(self):
        return self.capacity * 100

class Bus(Vehicle):
    pass
```

```
School_bus = Bus("School Volvo", 12, 50)
print("Total Bus fare is:", School_bus.fare())
```

4. Create a class Rectangle and initialize it with length and width.
 - a. Make a **Perimeter()** method to calculate the perimeter of the rectangle and an **Area()** method to calculate the area of the rectangle.
 - b. Make a method **display()** that displays the length, width, perimeter and area of an object created using an instantiation on rectangle class.

Create a Parallelepiped child class inheriting from the Rectangle class and initialize it with length, width and height.

- a. Make a **Volume()** method to calculate the volume of the Parallelepiped
5. Create a Computation class with a default constructor (without arguments) allowing to perform various calculations on integer numbers.
 - a. Create a method called **Factorial()** which takes an integer n as an argument and calculates the factorial of n.
 - b. Create a method called **Sum()** which takes an integer n as an argument and calculates the sum of the first n integers $1 + 2 + 3 + \dots + n$.
 - c. Create a method called **testPrime()** which takes an integer n as an argument and checks whether n is prime or not.
 - d. Create a **listDiv()** method that takes an integer n as an argument and returns a list of all the divisors of n.
 - e. Create a **listDivPrime()** method that takes an integer n as an argument and returns a list of all the prime divisors of n.