**Practical 9**

**Aim:** Consider an example of declaring the examination result. Design three classes: Student, Exam, and Result. The Student class has data members such as those representing rollNumber, Name, etc. Create the class Exam by inheriting Student class. The Exam class adds fields representing the marks scored in six subjects. Derive Result from the Exam class, and it has its own fields such as total\_marks. Write an interactive program to model this relationship.

**Code:**

class Student:

    def \_\_init\_\_(self, rollNo, name):

        self.rollNo = rollNo

        self.name = name

    def display(self):

        print(f'Student Roll No: {self.rollNo}')

        print(f'Student Name: {self.name}')

class Exam(Student):

    def \_\_init\_\_(self, rollNo, name, subject):

        super().\_\_init\_\_(rollNo, name)

        self.subject = subject

    def display(self):

        super().display()

        for i in range(len(self.subject)):

            print(f'Subject {i} Marks: {self.subject[i]}')

class Result(Exam):

    total\_marks = 0

    def \_\_init\_\_(self, rollNo, name, subject):

        super().\_\_init\_\_(rollNo, name, subject)

        self.total\_marks = sum(subject)

    def display(self):

        super().display()

        print(f'Total Marks: {self.total\_marks}')

if \_\_name\_\_ == '\_\_main\_\_':

    student = Student(1, 'Raj')

    student.display()

    print()

    exam = Exam(2, 'Deep', [10, 20, 30])

    exam.display()

    print()

    result = Result(3, 'Khushi', [40, 50, 60])

    result.display()

    print()

# Github Repo Link: https://github.com/s-shubham-22/20CE123\_CE259\_PIP

**Output:**

