1. Define a Python function student(). Using function attributes display the names of all arguments.

def student(x):

    print(x)

name = input("Enter the students name")

student(name)

1. Write a Python function student\_data () that will print the ID of a student (student\_id). If the user passes an argument student\_name or student\_class the function will print the student name and class.

def student(student\_id, student\_name, student\_class):

    print(student\_id)

    print(student\_name)

    print(student\_class)

id = int(input("Enter the student's id number"))

name = input("Enter the students name")

studentClass = input("Enter the students class")

student(id, name, studentClass)

1. Write a simple Python class named Student and display its type. Also, display the \_\_dict\_\_ attribute keys and the value of the \_\_module\_\_ attribute of the Student class.

class Student:

    pass

print(type(Student))

print(Student.\_\_dict\_\_.keys())

print(Student.\_\_module\_\_)

1. Write a Python program to create two empty classes, Student and Marks. Now create some instances and check whether they are instances of the said classes or not. Also, check whether the said classes are subclasses of the built-in object class or not.

class Student:

    pass

class Marks:

    pass

student1 = Student()

mark1 = Marks()

print(isinstance(student1, Student))

print(isinstance(student1, Marks))

print(isinstance(mark1, Marks))

print(isinstance(mark1, Student))

print(issubclass(Student, object))

print(issubclass(Marks, object ))

1. Write a Python class named Student with two attributes student\_name, marks. Modify the attribute values of the said class and print the original and modified values of the said attributes.

class Student:

    student\_name = "John Brooke"

    marks = 96

m2201995 = Student()

print(f"Student name: {m2201995.student\_name}")

print(f"Marks: {m2201995.marks}")

m2201995.student\_name = "Barry Ballon"

m2201995.marks = 100

print(f"Student name: {m2201995.student\_name}")

print(f"Marks: {m2201995.marks}")

1. Write a Python class named Student with two attributes student\_id, student\_name. Add a new attribute student\_class and display the entire attribute and the values of the class. Now remove the student\_name attribute and display the entire attribute with values.

class Student:

    student\_id = "01"

    student\_name = "John Cole"

print("Orginal Values: ")

print(f"Student ID: {getattr(Student, 'student\_id')}")

print(f"Student name: {getattr(Student, 'student\_name')}")

print("After adding student class: ")

Student.student\_class = "Class B"

print(f"Student ID: {getattr(Student, 'student\_id')}")

print(f"Student name: {getattr(Student, 'student\_name')}")

print(f"Student class: {getattr(Student, 'student\_class')}")

print("After removing student name: ")

del Student.student\_name

print(f"Student ID: {getattr(Student, 'student\_id')}")

print(f"Student class: {getattr(Student, 'student\_class')}")

1. Write a Python class named Student with two attributes: student\_id, student\_name. Add a new attribute: student\_class. Create a function to display all attributes and their values in the Student class.

class Student:

    student\_id = "01"

    student\_name = "Harry Wilson"

    def displayAll():

        print(f"Student ID: {Student.student\_id}, Student Name: {Student.student\_name}")

print("Displaying all attributes and their values: ")

Student.displayAll()

1. Write a Python class named Student with two instances student1, student2 and assign values to the instances' attributes. Print all the attributes of the student1, student2 instances with their values in the given format.