Green University Of Bangladesh

Department Of Computer Science and Engineering (CSE)

Faculty of Sciences and Engineering

Semester: (Summer, Year: 2023), B.Sc. in CSE (DAY)

LAB REPORT NO - 01

Course Title: Data Mining Lab

Course Code: CSE-436 **Section:** DB

Lab Experiment Name: Python Problem Solve

Student Details

	Name	ID
1	Sk. Nahid	201902073

Lab Date : 09/10/2023

Submission Date : 15/10/2023

Course Teacher Name : Rezwanul Haque

Lab Report Status

Mark:..... **Signature:**.....

Comments:..... **Date:**.....

1 INTRODUCTION

Python is a high-level computer programming language. Using this we can solve problems.

2 OBJECTIVE

Here we have four tasks to do.

- 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.
- Create a Python class named Student with two attributes studentName, marks. Modifing the attribute values of the said class and print the original and modified values of the said attributes.
- Create e a Python class named Student with two attributes studentId, studentName. Adding a new attribute studentClass and displaying the entire attribute and their values of the said class. Then remove the studentName attribute and display the entire attribute with values.
- Create a Python class named Student with two attributes studentId, studentName. Adding a new attribute studentClass. Then create a function to display the entire attribute and their values in Student class.

3 PROCEDURE

Here for solving these problem we will use VSCode platform.

INSTANCE OR NOT: Said classes is instance or not can be justify by using isinstance() Function. It is a pre-built function of python. The isinstance() function returns True if the specified object is of the specified type, otherwise False. If the type parameter is a tuple, this function will return True if the object is one of the types in the tuple.

MODIFY THE VALUE OF ATTRIBUTE: It works like rewrite the value of attribute modify the previous value to new value.

ADD & REMOVE ATTRIBUTE: Writing the function name with new attribute name with the value can easily add new attribute with value. For remove using of del can remove any attribute.

DISPLAY ATTRIBUTE WITH VALUE USING FUNCTION: Below class function we will define another function where we pass the argument and print all of them.

4 IMPLEMENTATION

INSTANCE OR NOT

```python
# Question_01
class Student1:
    pass
class Marks:
    pass

sk_nahid = Student1()
passed = Marks()
check_student = isinstance(sk_nahid, Student1)
print("sk_nahid is a instance of class Student--", check_student)
check_marks = isinstance(sk_nahid, Marks)
print("sk_nahid is a instance of class Marks--", check_marks)
check_student = isinstance(passed, Student1)
print("passed is a instance of class Student--", check_student)
check_marks = isinstance(passed, Marks)
print("passed is a instance of class Marks--", check_marks)
check_student_subclass = isinstance(Student1, object)
print("Student class is a subclass of object--", check_student_subclass)
check_marks_subclass = isinstance(Marks, object)
print("Marks class is a subclass of object--", check_marks_subclass)
```

Listing 1: Check instance and subclasses.py

MODIFY THE VALUE OF ATTRIBUTE

```python
# Question_02
class Student2:
    def __init__(self, name, marks):
        self.name = name
        self.marks = marks

e1 = Student2("Sk. Nahid", 22)
print(e1.name, e1.marks)
e1.name = "Ahmed"
e1.marks = 23
print("modified value:", e1.name, e1.marks)
```

Listing 2: Converting array to ordinary list.py

ADD & REMOVE ATTRIBUTE

```python
class Student3:
    def __init__(self, ID, Name):
        self.ID = ID
        self.Name = Name

e1 = Student3("201902073", "Sk. Nahid")
print(e1.ID, e1.Name)
# adding new attribute to this class
e1.Marks = 82
```

```
print("New attribute=> Marks: ",e1.Marks)
# deleting attribute
del e1.Marks
print("Deleted attribute=> Marks: ",e1.Marks)
```

Listing 3: Add & Remove attribute

DISPLAY ATTRIBUTE WITH VALUE USING FUNCTION

```
class Student4:
    def __init__(self, StudentID, StudentName):
        self.StudentID = StudentID
        self.StudentName = StudentName
    def function(self):
        print(f'Student name is {self.StudentName} & ID is {self.StudentID}')

e1 = Student4("201902073", "Sk. Nahid")
e1.function()
```

Listing 4: Display attribute value using function

5 OUTPUT

INSTANCE OR NOT

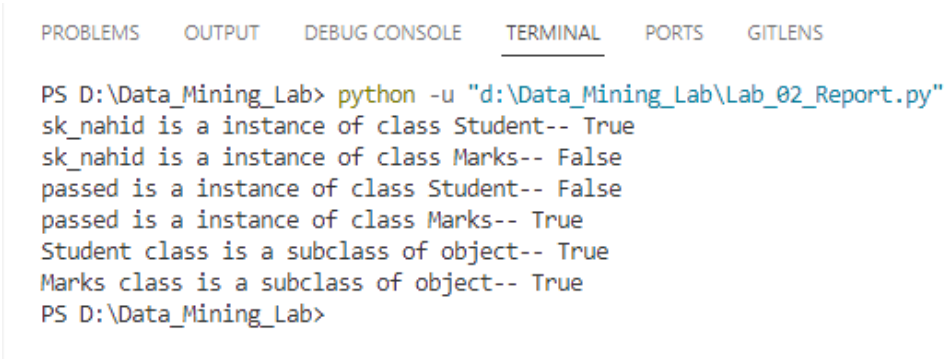

Figure 1: Check instance and sub-classes

MODIFY THE VALUE OF ATTRIBUTE ADD & REMOVE ATTRIBUTE

Figure 2: Converting array to list

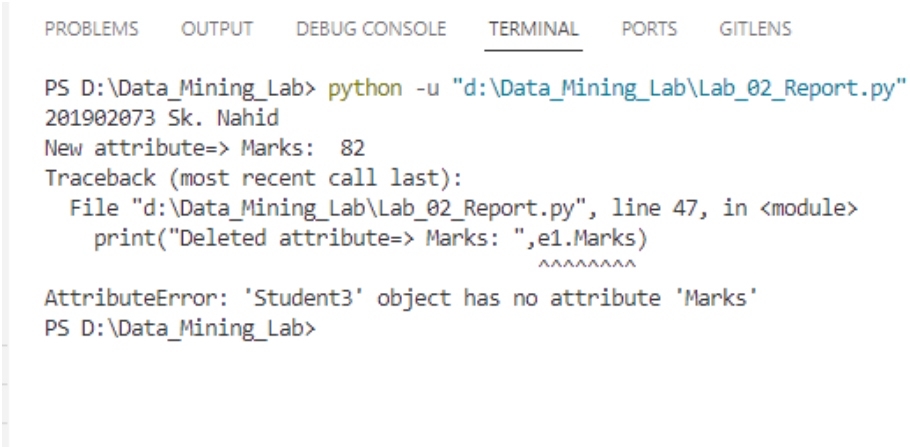

Figure 3: Remove duplicate value from dictionary

DISPLAY ATTRIBUTE WITH VALUE USING FUNCTION

Figure 4: Remove duplicate value from dictionary

6 DISCUSSION & ANALYSIS

For solving this problem here we use python language. Here we use classes, instance etc. Adding new attribute was just like mention a new attribute with value and deleting was too easy, just use del for delete one.

7 SUMMARY

Though python was not new for us, it was very interesting to doing code in python. Also, this lab report completed using LaTeX.