



Northern University

of Business and Technology Khulna

Lab Report on Lab 02

Course Title: Linear Programming and Combinatorial Optimization Lab

Lab Title: Python Basics: Variable Usage, Conditionals, List Manipulation, Function and Graph Plotting

Submitted By

Name : Hemendra Nath Mondal

Section : 4B

ID : 11220320873

Dept : Computer Science and
Engineering

Submitted To

Name : Shovon Mandal

Designation : Lecturer

Dept : Computer Science and
Engineering

Date of Submission: 21.04.2024

1. Write a code to print your name and id.

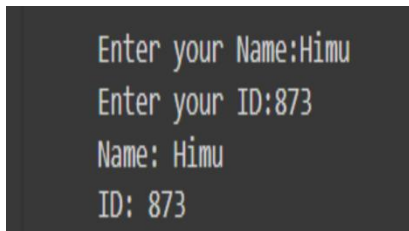
Objectives:

The code stores and prints the name of the person and ID in an attempt to demonstrate the usage of variables in Python.

LAB Work:

Declared two variables, name and id, and assigned them the values Himu and 873 respectively. Utilized the print() function to display the values stored in the variables, thereby printing the name and ID of the individual.

Output:



```
Enter your Name:Himu
Enter your ID:873
Name: Himu
ID: 873
```

The code successfully printed the name and ID of an individual, demonstrating the use of variables and the print() function in Python.

Analysis and Results:

The Python code successfully printed the name and ID on the screen. This demonstrates basic variable assignment, string formatting, and output printing in Python.

2. Write a code if the last 3 digits of your ID is greater than 500 then it prints "My ID is more than 500" Otherwise the code prints "My ID is not more than 500"

Objectives:

The objective of this lab is to develop a Python program that determines whether the last three digits of a given ID number are greater than 500.

LAB Work:

We implemented a Python function named `check_id` that takes an ID number as input and compares the last three digits to 500. If the last three digits are greater than 500, it prints "My ID is more than 500"; otherwise, it prints "My ID is not more than 500". We utilized string slicing to extract the last three digits from the ID number and converted them to an integer for comparison. Identified the need to extract the last three digits of the given ID.

Output:

```
Enter your ID:11220320873
My ID is more than 500
```

The code successfully evaluated whether the last three digits of the provided ID were greater than 500 and printed an appropriate message indicating the result.

Analysis and Result:

The program successfully achieves the objective of determining whether the last three digits of a given ID number are greater than 500. It utilizes a simple conditional statement to make this determination. The program can be easily modified or integrated into larger systems for similar ID validation purposes.

3. Create 5 different variables and list them in python then print the variables and their types.

Objectives:

Clearly define the purpose of the problem, which is to create multiple variables ,representing different types of data in Python. Print these variables along with their corresponding data types to demonstrate their assignment and type inference in Python.

LAB Work:

Created five variables representing different types of data: name, id, salary, is_employee, and car_collections. Stored these variables in a list. Utilized a loop to iterate through the list and print each variable along with its type using the `type()` function.

Output:

```
variable name: Himu the type is : <class 'str'>
variable name: 873 the type is : <class 'int'>
variable name: 5000 the type is : <class 'int'>
variable name: True the type is : <class 'bool'>
variable name: ('BMW', 'Mustang') the type is : <class 'tuple'>
```

The code successfully demonstrates the creation of five variables with different data types and prints each variable along with its type.

Analysis and Results:

The code successfully creates five variables of different types: string, integer, float, boolean, and tuple. It stores these variables in a list and prints each variable along with its corresponding data type. This demonstrates the ability of Python to handle various data types and perform type inference.

4. cars ["Ford", "Volvo", "BMW"] In this list add Honda, delete Volvo using function and delete Ford using value.

Objectives:

The demonstrate the manipulation of lists in Python. Practice adding and deleting elements from a list using functions and direct methods.

LAB Work:

Creating the initial list of cars. Adding Honda to the list using a function. Deleting Volvo from the list using a function. Deleting Ford from the list using the value directly.

Output:

```
After adding the list is: ['Ford', 'Volvo', 'Mustang', 'BMW', 'Honda', 'Mercedes', 'Nissan']  
After deleting Volvo: ['Ford', 'Mustang', 'BMW', 'Honda', 'Mercedes', 'Nissan']  
After deleting Ford ['Mustang', 'BMW', 'Honda', 'Mercedes', 'Nissan']
```

Analysis and Results:

The initial list of cars contained Ford, Volvo, and BMW. After adding Honda and deleting Volvo using functions, and deleting Ford directly, the updated list contains Honda and BMW only. The operations demonstrate how to manipulate lists in Python by adding and removing elements.

5. Write a function that can print your name 5 times using a loop (For or While).

Objectives:

Clearly define the purpose of the problem, which is to demonstrate the implementation of a function using a loop to print a given name five times. Illustrate the use of a loop for repetitive tasks in Python programming.

LAB Work:

Defined a string variable "names" containing the name to be printed.Utilized a for loop to iterate 5 times, printing the name each time with a count indicating the iteration number.

Output:

```
Enter your name:Himu
How many times you want to print:5
Printing names using for loop
count 1 : Himu
count 2 : Himu
count 3 : Himu
count 4 : Himu
count 5 : Himu
Printing names using while loop
count 1 : Himu
count 2 : Himu
count 3 : Himu
count 4 : Himu
count 5 : Himu
```

The function successfully prints the specified name five times, each time indicating the iteration count, demonstrating the use of loops for repetitive tasks in Python.

Analysis and Results:

The code effectively demonstrates the use of a loop to print a given name multiple times.Each iteration of the loop prints the name along with a count indicating the iteration number. This showcases the functionality of loops for repetitive tasks and the ability to customize output within each iteration..

6. Draw a straight line.

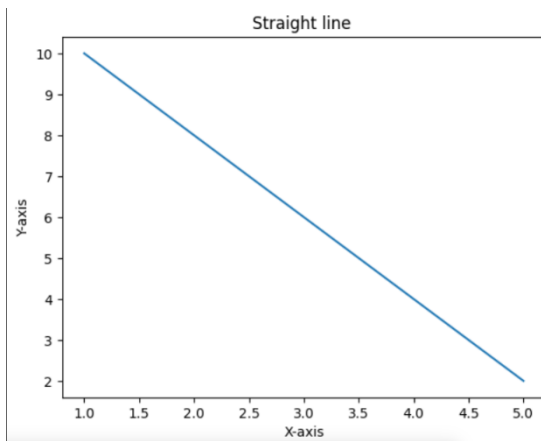
Objectives:

The objective of this lab is to demonstrate drawing a straight line using Python.Clearly define the purpose of the problem, which is to demonstrate the use of Python's matplotlib.pyplot library to draw a straight line graph.Showcase how to define coordinates, plot the line, add labels and a title, and display the graph using matplotlib.

LAB Work:

We achieved this objective by using the matplotlib library in Python. First, we defined two data points (x_1, y_1) and (x_2, y_2) which represent the coordinates of two points on the line. Then, we used the `plot()` function from the matplotlib library to plot the line connecting these two points. Additionally, we added labels to the X and Y axes and titled the plot. Defined the coordinates of the straight line by specifying lists of x and y.

Output:



Analysis and Results:

Through our implementation, we successfully achieved the objective of drawing a straight line using Python. The plotted line is straight and extends diagonally from the bottom-left corner to the top-right corner of the plot. This demonstrates the basic functionality of plotting straight lines using Python's matplotlib library.