

PROGRAMMING FUNDAMENTALS

Lab Experiment #02

---

**OBJECTIVE:**

Installation of Python, understanding basic input and output, introduction with data types

**TOOLS REQUIRED:**

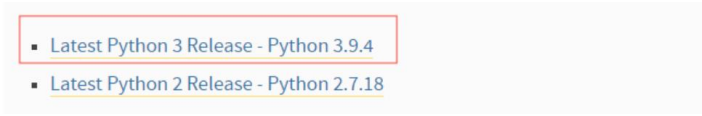
Personal computer with internet connection

**DESCRIPTION:**

In this lab experiment, you will first install Python development interface (IDLE) on the lab computer and become familiar with the IDLE environment. Later, you will write your first Python program and practice with the basic data types.

**LAB TASK:**

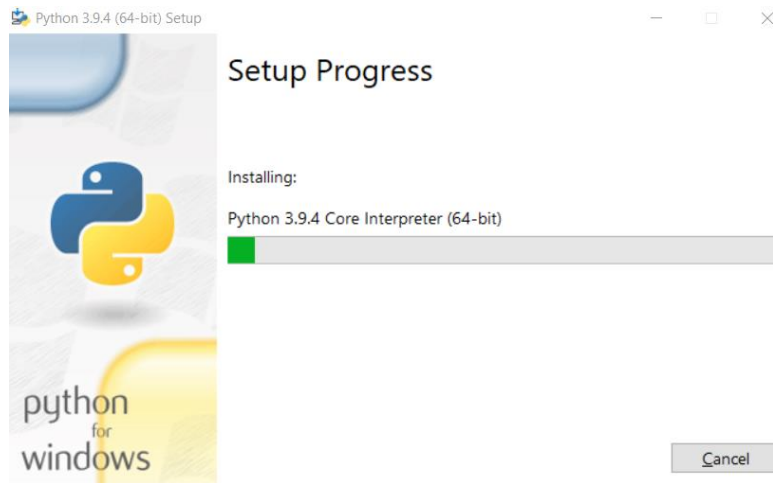
1. Go to <https://www.python.org/downloads/windows/> and select a version of Python to install (This lab shows the installation of version 3.9.4). Download the executable installer.

- 
- Latest Python 3 Release - Python 3.9.4
  - Latest Python 2 Release - Python 2.7.18

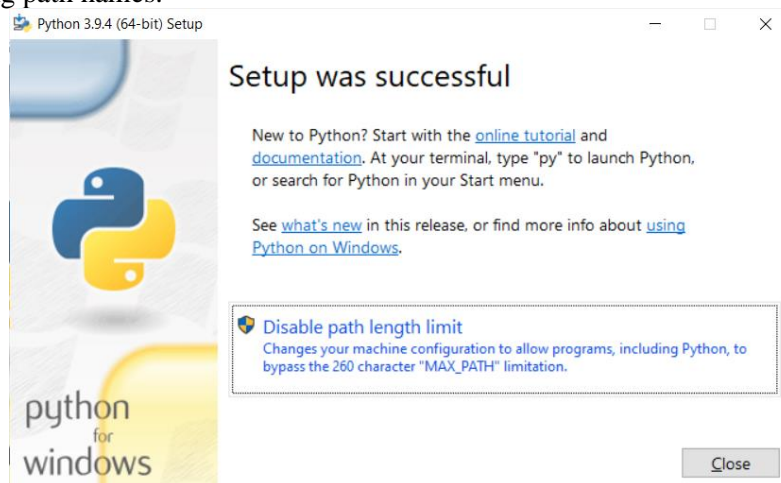
2. Double-click the executable file, which is downloaded; the following window will open. Click on the Add Path check box, it will set the Python path automatically.



3. Run the Python Installer once downloaded. (In this example, we have downloaded Python 3.9.4)
4. Make sure you select the “Install launcher for all users” and “Add Python 3.9 to PATH” checkboxes.
5. Select Install Now — the recommended installation options.
6. Wait for it to complete the installation process



7. The next dialog will prompt you to select whether to Disable the path length limit. Choosing this option will allow Python to bypass the 260-character MAX\_PATH limit. Effectively, it will enable Python to use long path names.



8. To check if Python 3.9.4 has been successfully installed in our system, open *cmd* prompt in your system and run `"Python -V"`
9. To open Python IDLE environment, open windows search, and type "IDLE" and open it.
10. From the "File" menu, select "New file" and create a new blank python script file
11. Save the script file as "lab2\_1.py" at appropriate location on the disk
12. Modify "lab2\_1.py"
  - a. Use two string variables named "first\_name" and "last\_name". Assign values (first\_name="Ali", last\_name="Khan") to them and then print the names on the screen using "print" function. The output should be like the following.
 

```
>>> print('First name is: Ali')
      print('Last name is: Khan')
      >>>
```
  - b. Modify the program to join first name and last name and show them on one line, like the following
 

```
>>> print('First name is: Ali and last name is: Khan')
      >>>
```
  - c. Use `input()` function to ask the user the first name and last name and then display on the output. The output should be similar to the following:

```

Enter first name:Ali
Enter last name:Khan
Welcome:  Ali   Khan
>>>

```

- d. Further modify the program to find and display the length of the name entered by the user. Use *len()* to find the length of the names and then display using *print()* function. Your output should match the following:

```

Enter first name:Ali
Enter last name:Khan
Welcome:  Ali   Khan
Your first name contains 3 characters, and your last name contains 4 characters
>>>

```

- e. Modify the program to show the user entered name in small, capital and title case.

13. Write a program “lab2\_2.py” that inputs two numbers and shows the results of addition, subtraction, multiplication, and division operations on both numbers. The output should match the following:

```

Input the first number:5
Input the second number:3
A+B 8
A-B 2
A*B 15
A/B 1.6666666666666667
>>>

```

14. Write a program “lab2\_3.py” that takes an input from the user and tells its type. Use *type()* function.

```

Input someting:a
You entered <class 'str'>
>>>

```

15. Write a program “lab2\_4.py” to learn about typecasting i.e., setting specific data types (*str()*, *float()*, *int()*).

## QUESTIONS:

Q # 1: Consider the following code snippet in Python?

```

a = 1.0
b = "1"
c = "1.1"

```

Now, write the output of the following lines

```

1. a + float(b)
2. float(b) + float(c)
3. a + int(c)
4. a + int(float(c))
5. int(a) + int(float(c))
6. 2.0 * b

```

Ans.

1. \_\_\_\_\_  
 2. \_\_\_\_\_  
 3. \_\_\_\_\_  
 4. \_\_\_\_\_  
 5. \_\_\_\_\_  
 6. \_\_\_\_\_

Q # 2: What would be the output of the following Python code?

a)

```
a,b='red','blue'  
a,b=b,a  
print(a,b)
```

b)

```
a='10'  
print(a+a)
```

```
str = 'Hello World!'  
  
print (str)  
print (str[0])  
print (str[2:5])  
print (str[2:])  
print (str * 2)  
print (str + "TEST")
```

**Name:** \_\_\_\_\_

**Roll #:** \_\_\_\_\_

**Date:** \_\_\_\_\_

**Subject Teacher**

**Remarks:**