Experiment no 1

Introduction to Python Programming

OBJECTIVES:

- To learn the basics of Python Programming.
- To learn datatypes, variables and some of the loops in python

1. INTRODUCTION:

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.
It is used for:

- Web development (server-side)
- Software development
- Mathematics
- System scripting

2. PYTHON ENVIRONMENT SETUP:

Download python from its official website using the link **https://www.python.org/downloads/** and then we will head towards downloading the basic environment used for Python programming. Environment link will be **https://www.anaconda.com/download/.** We will download the Anaconda environment and then we will use the Jupyter Notebook for the compilation of python programming. Jupyter Notebook will be used an IDE.

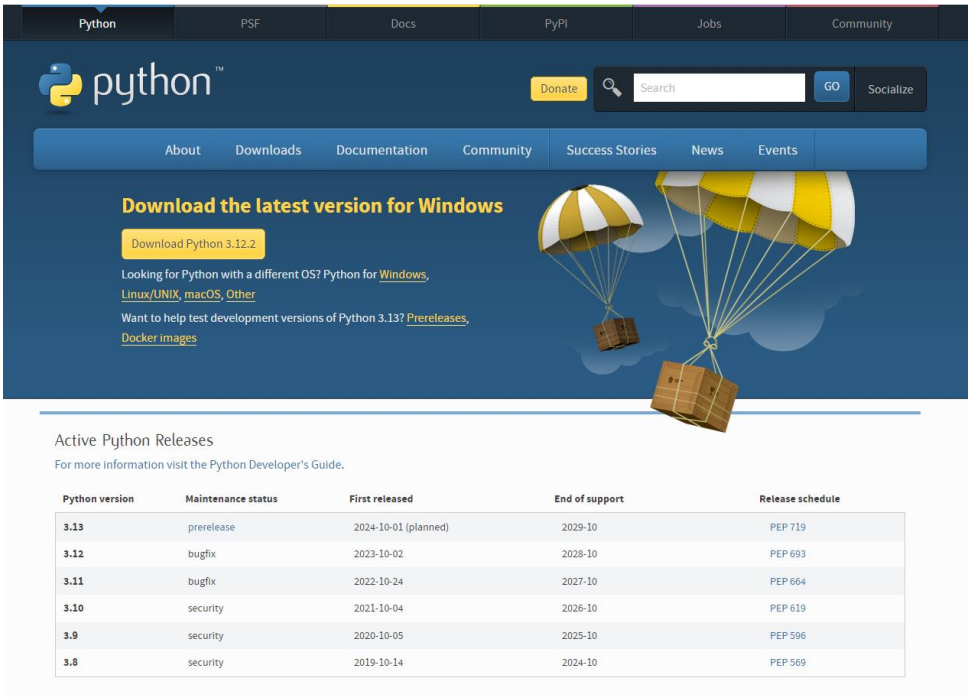

Figure 1: Python Download

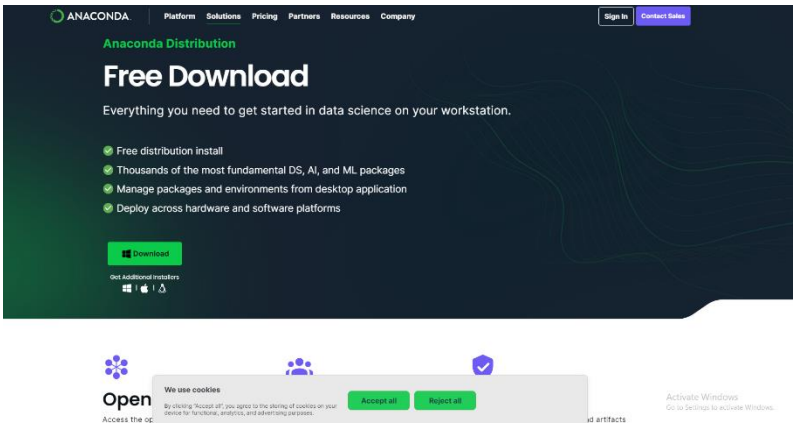

Figure 2: Anaconda Environment Download

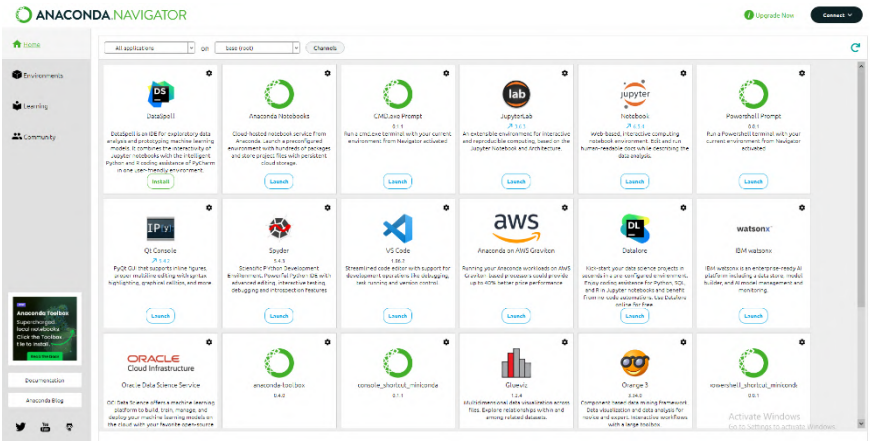

Figure 3: Anaconda Distribution

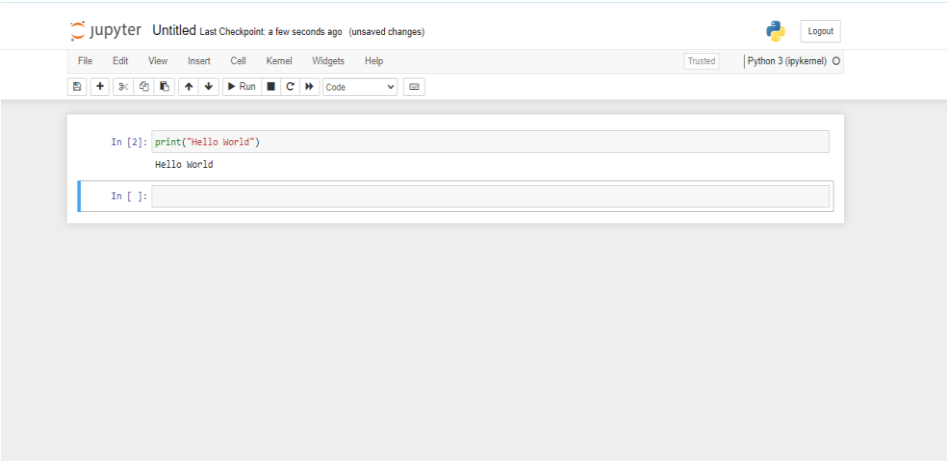

Figure 4: Jupyter Notebook

3. PYTHON BASICS:

3.1 Execute Python Syntax

Python syntax can be executed by writing directly in the Command Line:

```
>>> print("Hello, World!")
Hello, World!
```

Figure 5: Python Script (Shell Execution)

3.2 Python Indentation

Indentation refers to the spaces at the beginning of a code line. Where in other programming languages the indentation in code is for readability only, the indentation in Python is very important. Python uses indentation to indicate a block of code.

```
if 5 > 2:
  print("Five is greater than two!")
```

Figure 6: Python Indentation

3.3 Python Comments

Comments can be used to explain Python code. Comments can be used to make the code more readable. Comments can be used to prevent execution when testing code.

3.3.1 Single line Comment

Comments starts with a #, and Python will ignore them. Comments can be placed at the end of a line, and Python will ignore the rest of the line.

```
#This is a comment
print("Hello, World!")
```

Figure 7: Python Single Line Comment

3.3.2 Multi Line Comment

You can use a multiline string. Since Python will ignore string literals that are not assigned to a variable, you can add a multiline string (triple quotes) in your code, and place your comment inside it.

```
"""
This is a comment
written in
more than just one line
"""
print("Hello, World!")
```

Figure 8: Python Multi Line Comment

3.4 Python Variables

3.4.1 Creating Variables

Python has no command for declaring a variable. A variable is created the moment you first assign a value to it. Variables do not need to be declared with any particular type, and can even change type after they have been set. If you want to specify the data type of a variable, this can be done with casting.

```
x = 4       # x is of type int
x = "Sally" # x is now of type str
print(x)
```

Figure 9: Python Variables

```
x = str(3)    # x will be '3'
y = int(3)    # y will be 3
z = float(3)  # z will be 3.0
```

Figure 10: Casting Variables

3.4.2 Multiple Values Variable

Python allows you to assign values to multiple variables in one line. And you can assign the same value to multiple variables in one line. If you have a collection of values in a list, tuple etc. Python allows you to extract the values into variables. This is called **unpacking**.

```
x, y, z = "Orange", "Banana", "Cherry"
print(x)
print(y)
print(z)
```

Figure 11: Multiple Value Variables

```
x = y = z = "Orange"
print(x)
print(y)
print(z)
```

Figure 12: One value to multiple Variables

```
fruits = ["apple", "banana", "cherry"]
x, y, z = fruits
print(x)
print(y)
print(z)
```

Figure 13: Unpacking Variables

3.5 Python Datatypes

In programming, data type is an important concept. Variables can store data of different types, and different types can do different things. Python has the following data types built-in by default.

Example	Data Type
x = "Hello World"	str
x = 20	int
x = 20.5	float
x = 1j	complex
x = ["apple", "banana", "cherry"]	list
x = ("apple", "banana", "cherry")	tuple
x = range(6)	range
x = {"name" : "John", "age" : 36}	dict
x = {"apple", "banana", "cherry"}	set
x = frozenset({"apple", "banana", "cherry"})	frozenset
x = True	bool
x = b"Hello"	bytes
x = bytearray(5)	bytearray
x = memoryview(bytes(5))	memoryview
x = None	NoneType

Figure 14: Python Datatypes

3.6 Python Input

The input() function allows user to give input for the variable. If value is not defined and user need to enter a customized value then input() function with a variable is defined and then the compiler takes the input from the user for that variable.

```
print('Enter your name:')
x = input()
print('Hello, ' + x)
```

Figure 15: Input Function

4. PYTHON CONDITIONAL STATEMENTS:

4.1 if else Statement

Python supports the usual logical conditions from mathematics:

- Equals: a == b
- Not Equals: a != b
- Less than: a < b
- Less than or equal to: a <= b
- Greater than: a > b
- Greater than or equal to: a >= b

These conditions can be used in several ways, most commonly in "if statements" and loops. An "if statement" is written by using the if keyword.

```
a = 200
b = 33
if b > a:
  print("b is greater than a")
else:
  print("b is not greater than a")
```

Figure 16: If Else Statement

4.2 elif Statement

The elif keyword is Python's way of saying "if the previous conditions were not true, then try this condition".

```
a = 200
b = 33
if b > a:
  print("b is greater than a")
elif a == b:
  print("a and b are equal")
else:
  print("a is greater than b")
```

Figure 17: Elif Statement

5. PYTHON LOOPS:

5.1 For Loop

A for loop is used for iterating over a sequence (that is either a list, a tuple, a dictionary, a set, or a string). This is less like the for keyword in other programming languages, and works more like an iterator method as found in other object-orientated programming languages. With the for loop we can execute a set of statements, once for each item in a list, tuple, set etc.
To loop through a set of code a specified number of times, we can use the range() function.

```
for x in range(6):
  print(x)
```

Figure 18: For Loop

```
for x in range(2, 6):
  print(x)
```

Figure 19: range() Functions Parameter

5.2 While Loop

With the while loop we can execute a set of statements as long as a condition is true.

```
i = 1
while i < 6:
  print(i)
  i += 1
```

Figure 20: While Loop

6. ACTIVITIES:

A1) Write a program that uses a while loop to print the numbers from 1 to 10, each on a separate line.

A2) Write a program that uses for loop to calculate the sum of the first 5 even numbers (2, 4, 6, 8, 10). Print the final sum.

A3) Write a simple code snippet that demonstrates the importance of indentation in Python. Use Comments and tell about the indentation error in the code.

A4) Define and assign a variable for each of the following data types on the compiler and use comments to define the datatype.

IMPORTANT: All the activities` code should be attached to the manual before summary section.

Lab Exercise and Summary

Summary should cover Introduction, Procedure, Data Analysis and Evaluation.

Student's Signature:_____ **Date:**_____

LABORATORY SKILLS ASSESSMENT (Psychomotor)

Total Marks:100

Criteria (Max Marks)	Level 1 0% ≤ S < 50%	Level 2 50% ≤ S< 70%	Level 3 70% ≤ S< 90%	Level 4 90%≤ S ≤100%	Score (S)
Procedural Awareness (30)	Selects inappropriate skills and/or strategies Required by thetask.	Selects and applies appropriate skills and/or strategies required by the task with major errors.	Selects and applies the appropriate strategies and/or skills specific to the task without significant errors.	Selects and applies appropriate strategies and/or skills specific to the task without any error.	
Practical Implementation (30)	Makes major critical errors in applying procedural knowledge related to python basics.	Makes numerous critical errors in applying procedural knowledge related to python basics. Assigned roles.	Makes some non-critical errors in applying procedural knowledge related to python basics.	Applies the procedural knowledge in optimized ways related to python basics.	
Use of Tool/Equipment (30)	Uses tools, equipment and materials with limited competence.	Uses tools, equipment and materials with some competence.	Uses tools, equipment and materials with considerable competence.	Uses tools, equipment and materials with a high degree of competence.	
Safety (10)	Requires constant reminders to follow safety procedures.	Requires some reminders to follow safety procedures.	Follows safety procedures with only minimal reminders.	Routinely follows safety procedures.	
				Marks Obtained	

Instructor Name:_____ **Sign:** _____

LABORATORY SKILLS ASSESSMENT (Affective)

Total Marks: 40

Criteria (Max. Marks)	Level 1 0% ≤ S < 50%	Level 2 50% ≤ S < 70%	Level 3 70% ≤ S < 90%	Level 4 90% ≤ S ≤ 100%	Score
Introduction (5)	Very little background information provided or information is incorrect	Introduction is brief with some minor mistakes	Introduction is nearly complete, missing some minor points	Introduction complete and well-written; provides all necessary background principles for the experiment	
Procedure (5)	Many stages of the procedure are not entered on the lab report.	Many stages of the procedure are entered on the lab report.	The procedure could be more efficiently designed but most stages of the procedure are entered on the lab report.	The procedure is well designed and all stages of the procedure are entered on the lab report.	
Data Record (10)	Data is brief and missing significant pieces of information.	Data provides some significant information and has few critical mistakes.	Data is almost complete but has some minor mistakes.	Data is complete and relevant. Tables with units are provided. Graphs are labeled. All questions are answered correctly.	
Data Analysis (10)	Data is presented in very unclear manner.	Data is presented in ways that are not clear enough.	Data is presented in ways that can be understood and interpreted.	Data is presented in ways that best facilitate understanding and interpretation.	
Report Quality (10)	Report contains many errors.	Report is somewhat organized with some spelling or grammatical errors.	Report is well organized and cohesive but contains some grammatical errors.	Report is well organized and cohesive and contains no grammatical errors. Presentation seems polished.	
				Marks Obtained	

LABORATORY SKILLS ASSESSMENT (Cognitive)

Total Marks: 10

(If any) **Marks Obtained**	

Instructor's Signature:_____ **Date:**_____