# S. Y. B. Tech (ECE)

**Trimester: VI Subject: Linux Based Python Laboratory (CET2005A)**

# Name: Shreerang Mhatre Class: Electrical and Computer

**Roll No.: 29 Batch: A2**

# Experiment – 03 Title: Introduction to Fundamentals of Python Performed on: 02/11/2022

**Marks**

**Teacher’s Signature with date**

**Submitted on: 02/11/2022**

**Aim**: Introduction to Fundamentals of Python

# Objective:

* 1. To know the Fundamentals of Python.
  2. To implement the basic Python programs.

# Theory:

Python has the following pedagogical benefits:

* Python has simple, conventional syntax. Python statements are very close to those of pseudocode algorithms, and Python expressions use the conventional notation found in algebra. Thus, you can spend less time dealing with the syntax of a programming language and more time learning to solve interesting problems.
* Python has safe semantics. Any expression or statement whose meaning violates the definition of the language produces an error message.
* Python scales well. It is easy for beginners to write simple programs in Python. Python also includes all the advanced features of a modern programming language, such as support for data structures and object-oriented software development, for use when they become necessary.
* Python is highly interactive. You can enter expressions and statements at an interpreter’s prompts to try out experimental code and receive immediate feedback. You can also compose longer code segments and save them in script files to be loaded and run as modules or standalone applications.
* Python is general purpose. In today’s context, this means that the language includes resources for contemporary applications, including media computing and web services.
* Python is free and is in widespread use in the industry. You can download Python to run on a variety of devices. There is a large Python user community, and expertise in Python programming has great resume value.

To summarize these benefits, Python is a comfortable and flexible vehicle for expressing ideas about computation, both for beginners and for experts.

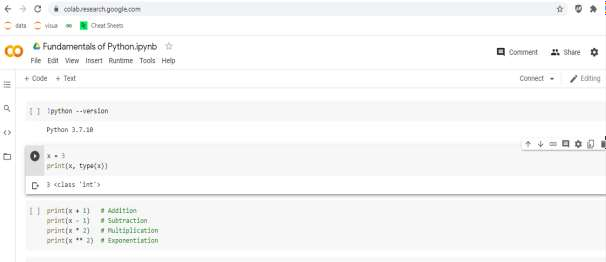
Python Applications:

* Data Science
* Date Mining
* Desktop Applications
* Console-based Applications
* Mobile Applications
* Software Development
* Artificial Intelligence
* Web Applications
* Enterprise Applications
* 3D CAD Applications
* Machine Learning
* Computer Vision or Image Processing Applications.
* Speech Recognitions

# Python Documentation:

**Execution of Python code using Google Colaboratory:**

* Colaboratory, or “Colab” for short, is a product from Google Research.
* Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education.
* More technically, Colab is a hosted Jupyter notebook service that requires no setup to use, while providing free access to computing resources
* including GPUs.

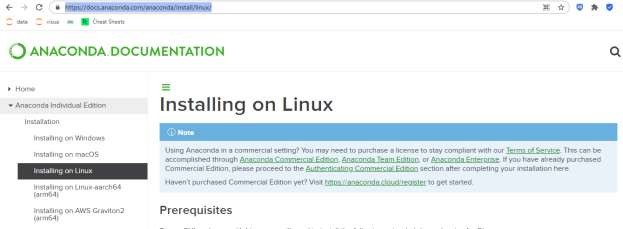


# Execution of Python code using Anaconda Navigator:

* Anaconda Navigator is a desktop GUI that comes with Anaconda Individual Edition.
* It makes it easy to launch applications and manage packages and environments without using command-line commands

Install Anaconda by following steps given in the link below:

[https://docs.google.com/document/d/16eQKQBbaI\_XlEHqHV9N7UKOEaMEQYpRoXOykYU-](https://docs.google.com/document/d/16eQKQBbaI_XlEHqHV9N7UKOEaMEQYpRoXOykYU-ZJQA/edit?usp=sharing) [ZJQA/edit?usp=sharing](https://docs.google.com/document/d/16eQKQBbaI_XlEHqHV9N7UKOEaMEQYpRoXOykYU-ZJQA/edit?usp=sharing)



# Python Data Types:

|  |  |
| --- | --- |
| Text Type: | str |
| Numeric Types: | int, float, complex |
| Sequence Types: | list, tuple, range |
| Mapping Type: | dict |
| Set Types: | set, frozenset |
| Boolean Type: | bool |
| Binary Types: | bytes, bytearray, memoryview |

**Basic Python Commands:**

Checking Version:

!python --version

Printing:

x = 4

print(x, type(x))

Arithmetic Operations:

x=5

print(x + 1) # Addition print(x - 1) # Subtraction print(x \* 2) # Multiplication print(x \*\* 2) # Exponentiation

String:

hello = 'hello' # String literals can use single quotes world = "world" # or double quotes; it does not matter print(hello, len(hello))

**Input**: Pyhton Commands **Output:** Output of each command **Conclusion:**

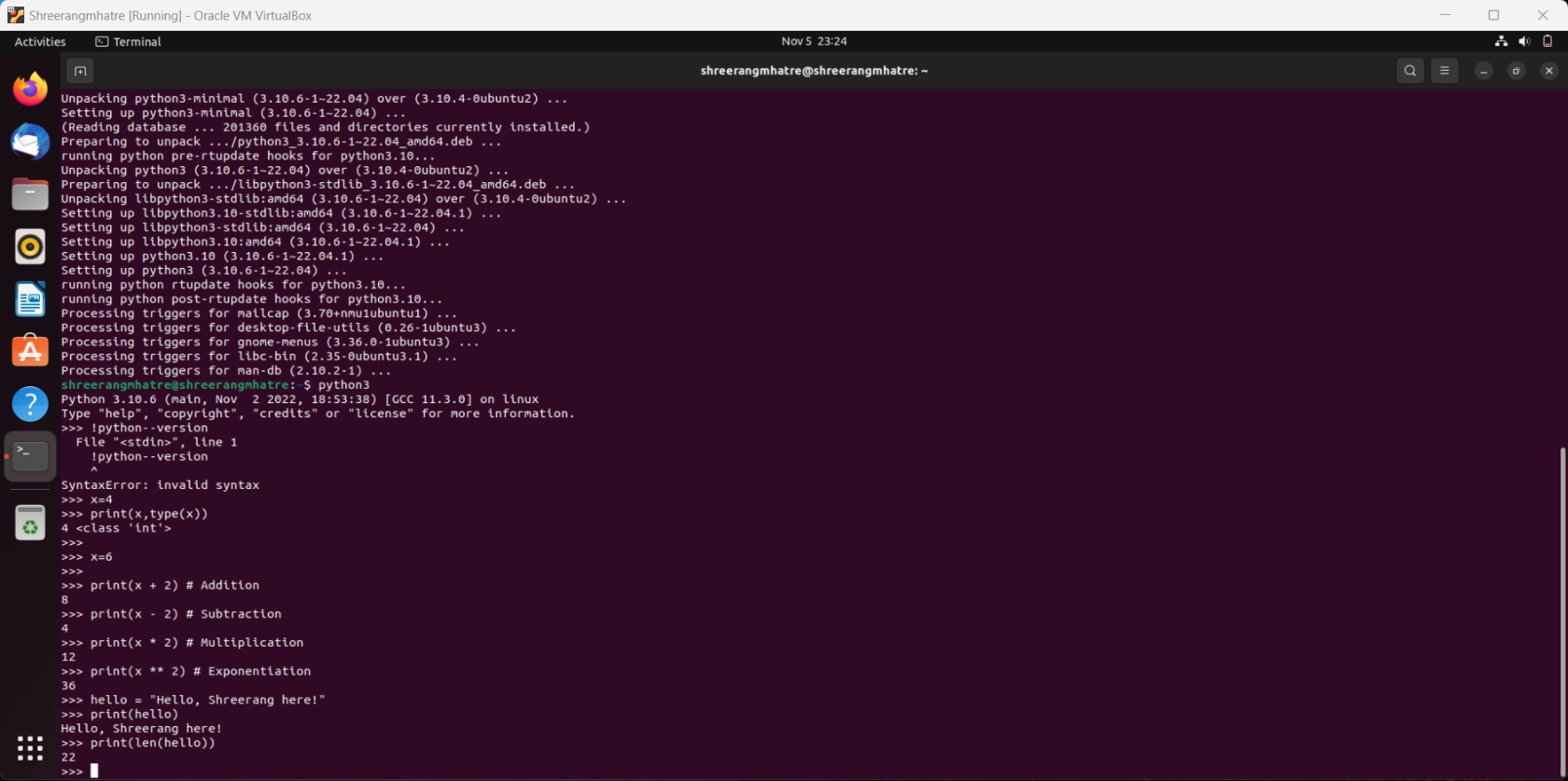
Thus, we have installed python in Ubuntu using Oracle VirtualBox and executed all basic python

commands.

# Additional Reference Links:

1. [https://www.python.org](https://www.python.org/)
2. [https://www.tutorialspoint.com](https://www.tutorialspoint.com/)
3. <https://www.programiz.com/python-programming>

# Procedure:



# Post Lab Questions:

1. What are basic Python program elements?

Ans) Basic Python program elements are:

strings (text), numbers (integers and floating point numbers), tuples (simple sequences), lists (more

flexible sequences), and dictionaries

1. Write a Python Program to Swap Two Variables.

Ans)

Program: -

# Python program to swap two variables

x = 5

y = 10

# create a temporary variable and swap the values

temp = x

x = y

y = temp

print('The value of x after swapping: {}'.format(x))

print('The value of y after swapping: {}'.format(y))

1. Write a Python Program using if-elif-else.

Ans)

Program: -

num = 3.4

if num > 0:

print("Positive number")

elif num == 0:

print("Zero")

else:

print("Negative number")

1. Write a Python Program to Find the Sum of Natural Numbers

Ans)

Program: -

num = int(input("Enter a number: "))

if num < 0:

print("Enter a positive number")

else:

sum = 0

# use while loop to iterate un till zero

while(num > 0):

sum += num

num -= 1

print("The sum is",sum)