**Week - 1**

**Lab Sessions - 1,2**

**Lab -1 - Part I**

**Part A**

**Lab Aim:** To study and understand Basic Python fundamentals and installation

**Outcome: At the end of the practical students will**

* **Understand and install different Python environments (Google Collab and IDLE)**
* **Implement programs using basic concepts of Python like Variables, Data Types and Operators**

**Task 1: Installation & setup of Google Collab and IDLE (For Python)**

1. **Google Colab**

Colab is a free notebook environment that runs entirely in the cloud. It lets you and your team members edit documents, the way you work with Google Docs. Colab supports many popular machine learning libraries which can be easily loaded in your notebook.

This tutorial gives an exhaustive coverage of all the features of Colab and makes you comfortable working on it with confidence.

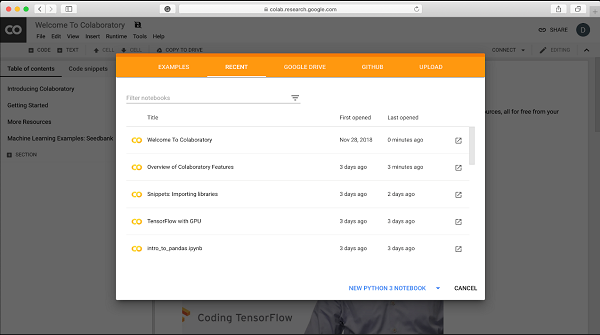
What Colab Offers You?

As a programmer, you can perform the following using Google Colab.

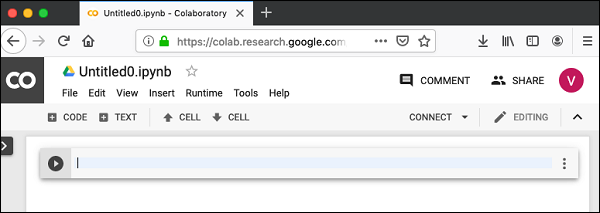
* Write and execute code in Python
* Document your code that supports mathematical equations
* Create/Upload/Share notebooks
* Import/Save notebooks from/to Google Drive
* Import/Publish notebooks from GitHub
* Import external datasets e.g. from Kaggle
* Integrate PyTorch, TensorFlow, Keras, OpenCV
* Free Cloud service with free GPU

Note − As Colab implicitly uses Google Drive for storing your notebooks, ensure that you are logged in to your Google Drive account before proceeding further.

Step 1 − Open the following URL in your browser − [https://colab.research.google.com](https://colab.research.google.com/notebooks/welcome.ipynb) Your browser would display the following screen (assuming that you are logged into your Google Drive) −



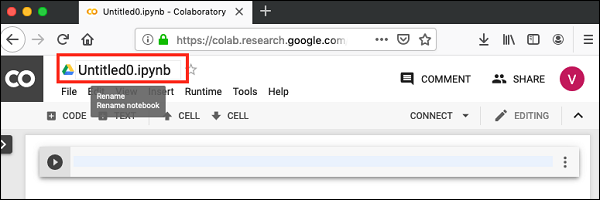
Step 2 − Click on the NEW PYTHON 3 NOTEBOOK link at the bottom of the screen. A new notebook would open up as shown in the screen below.



As you might have noticed, the notebook interface is quite similar to the one provided in Jupyter. There is a code window in which you would enter your Python code.

Setting Notebook Name

By default, the notebook uses the naming convention UntitledXX.ipynb. To rename the notebook, click on this name and type in the desired name in the edit box as shown here −



We will call this notebook as MyFirstColabNotebook. So type in this name in the edit box and hit ENTER. The notebook will acquire the name that you have given now.

Entering Code

You will now enter a trivial Python code in the code window and execute it.

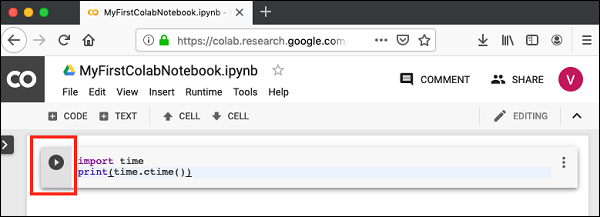
Enter the following two Python statements in the code window −

import time

print(time.ctime())

Executing Code

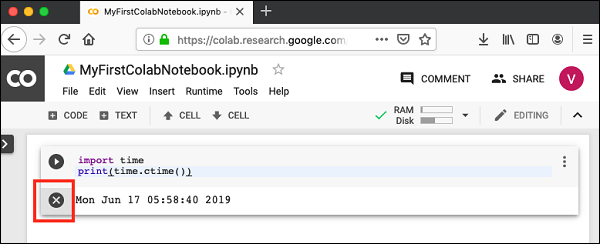
To execute the code, click on the arrow on the left side of the code window.



After a while, you will see the output underneath the code window, as shown here −

Mon Jun 17 05:58:40 2019

You can clear the output anytime by clicking the icon on the left side of the output display.

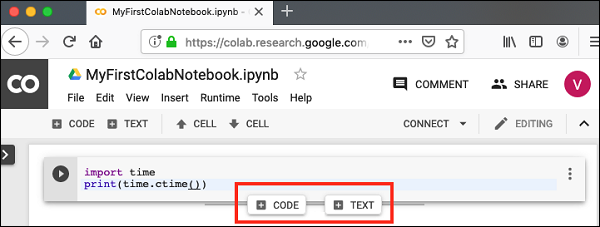


Adding Code Cells

To add more code to your notebook, select the following menu options −

Insert / Code Cell

Alternatively, just hover the mouse at the bottom center of the Code cell. When the CODE and TEXT buttons appear, click on the CODE to add a new cell. This is shown in the screenshot below −



A new code cell will be added underneath the current cell. Add the following two statements in the newly created code window −

time.sleep(5)

print (time.ctime())

Now, if you run this cell, you will see the following output −

Mon Jun 17 04:50:27 2019

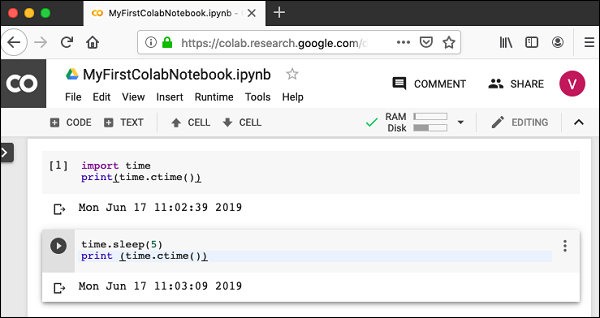
Certainly, the time difference between the two time strings is not 5 seconds. This is obvious as you did take some time to insert the new code. Colab allows you to run all code inside your notebook without an interruption.

Run All

To run the entire code in your notebook without an interruption, execute the following menu options −

Runtime / Reset and run all…

It will give you the output as shown below −



Note that the time difference between the two outputs is now exactly 5 seconds.

The above action can also be initiated by executing the following two menu options −

Runtime / Restart runtime…

or

Runtime / Restart all runtimes…

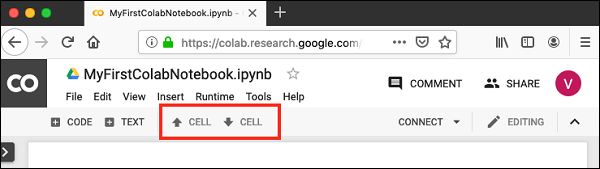
Followed by

Runtime / Run all

Study the different menu options under the Runtime menu to get yourself acquainted with the various options available to you for executing the notebook.

Changing Cell Order

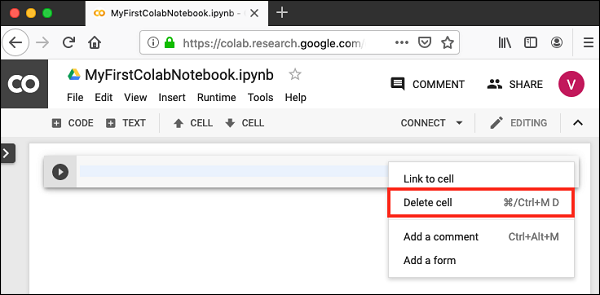
When your notebook contains a large number of code cells, you may come across situations where you would like to change the order of execution of these cells. You can do so by selecting the cell that you want to move and clicking the UP CELL or DOWN CELL buttons shown in the following screenshot −



You may click the buttons multiple times to move the cell for more than a single position.

Deleting Cell

During the development of your project, you may have introduced a few now-unwanted cells in your notebook. You can remove such cells from your project easily with a single click. Click on the vertical-dotted icon at the top right corner of your code cell.



Click on the Delete cell option and the current cell will be deleted.

Now, as you have learned how to run a trivial notebook, let us explore the other capabilities of Colab

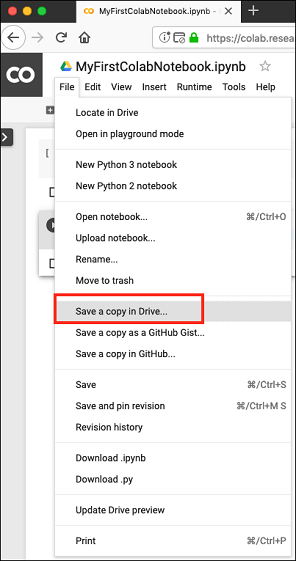
Colab allows you to save your work to Google Drive or even directly to your GitHub repository.

Saving to Google Drive

Colab allows you to save your work to your Google Drive. To save your notebook, select the following menu options −

File / Save a copy in Drive…

You will see the following screen −



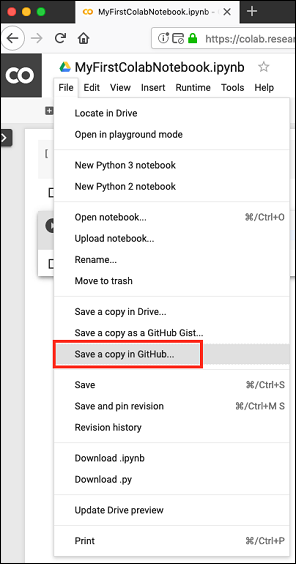
The action will create a copy of your notebook and save it to your drive. Later on you may rename the copy to your choice of name.

Saving to GitHub

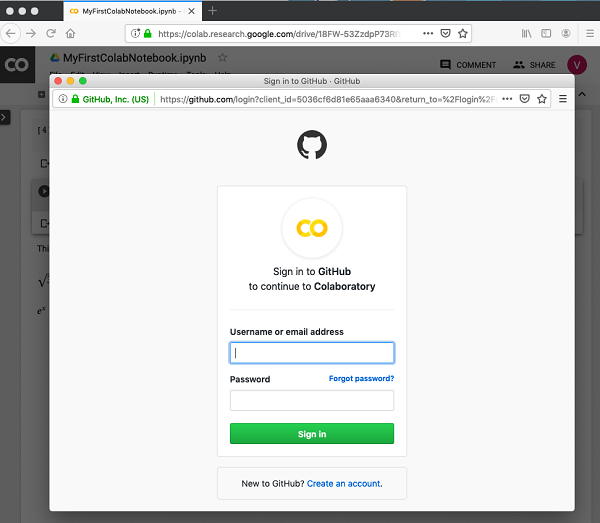
You may also save your work to your GitHub repository by selecting the following menu options −

File / Save a copy in GitHub...

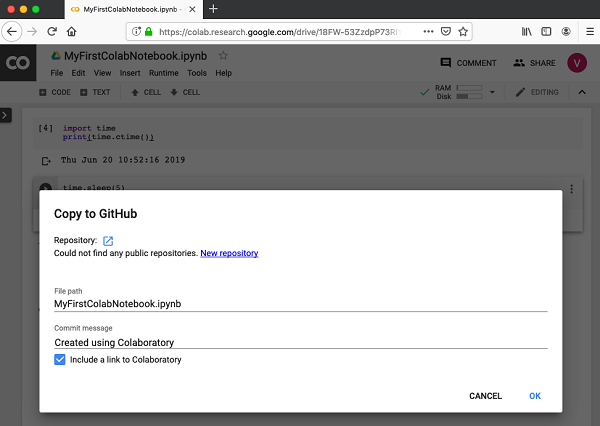
The menu selection is shown in the following screenshot for your quick reference −



You will have to wait until you see the login screen to GitHub.

a

Now, enter your credentials. If you do not have a repository, create a new one and save your project as shown in the screenshot below −

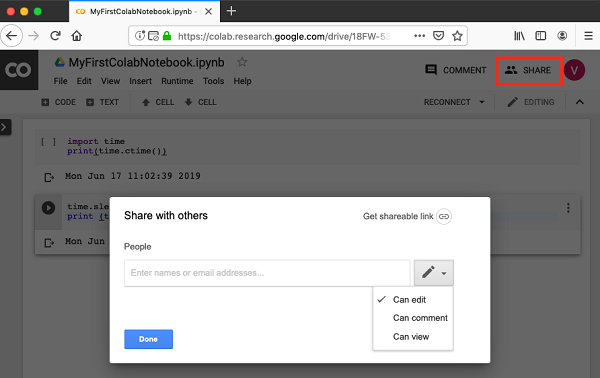


In the next chapter, we will learn how to share your work with others.

To share the notebook that you have created with other co-developers, you may share the copy that you have made in your Google Drive.

To publish the notebook to general audience, you may share it from your GitHub repository.

There is one more way to share your work and that is by clicking on the SHARE link at the top right hand corner of your Colab notebook. This will open the share box as shown here −



You may enter the email IDs of people with whom you would like to share the current document. You can set the kind of access by selecting from the three options shown in the above screen.

Click on the Get shareable link option to get the URL of your notebook. You will find options for whom to share as follows −

* Specified group of people
* Colleagues in your organisation
* Anyone with the link
* All public on the web

Now. you know how to create/execute/save/share a notebook. In the Code cell, we used Python so far. The code cell can also be used for invoking system commands. This is explained next.

1. **IDLE (For Python):**

Python – IDLE

IDLE (Integrated Development and Learning Environment) is an integrated development environment (IDE) for Python.

The Python installer for Windows contains the IDLE module by default.

IDLE is not available by default in Python distributions for Linux. It needs to be installed using the respective package managers. For example, in case of Ubuntu:

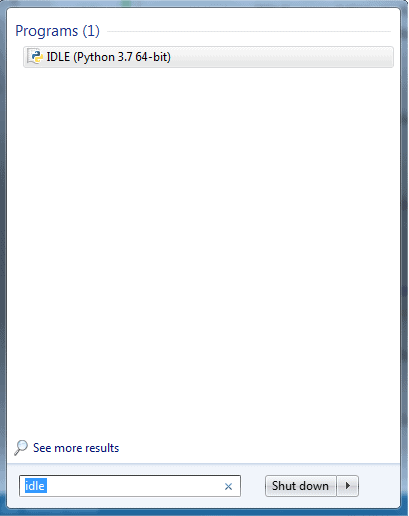
$ sudo apt-get install idle

IDLE can be used to execute a single statement just like Python Shell and also to create, modify and execute Python scripts.

IDLE provides a fully-featured text editor to create Python scripts that includes features like syntax highlighting, autocompletion and smart indent.

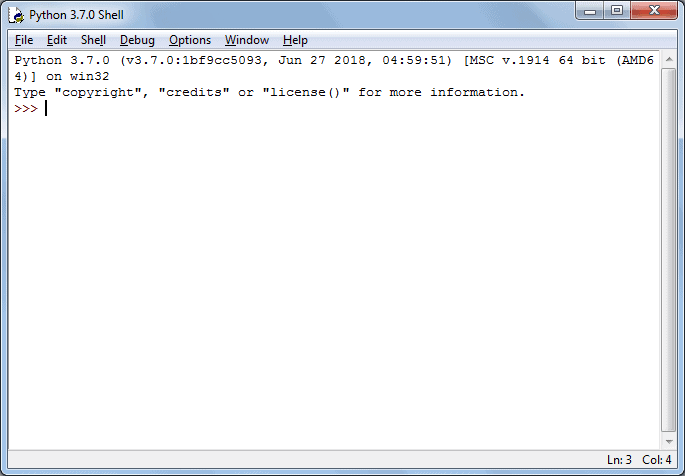
It also has a debugger with stepping and breakpoints features.

To start IDLE interactive shell, search for the IDLE icon in the start menu and double click on it.

[](https://www.tutorialsteacher.com/Content/images/python/open-idle.png)

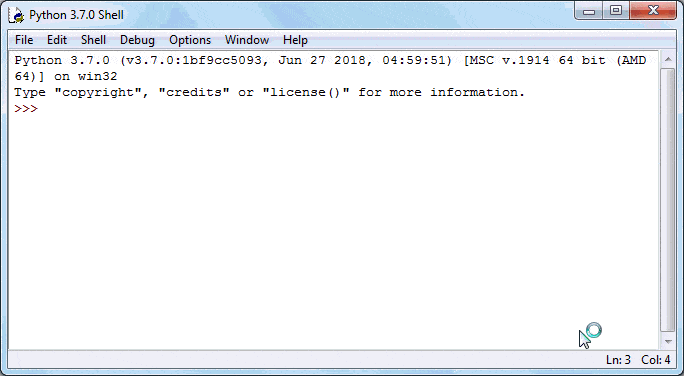
Python IDLE

This will open IDLE, where you can write Python code and execute it as shown below.

[](https://www.tutorialsteacher.com/Content/images/python/idle.png)

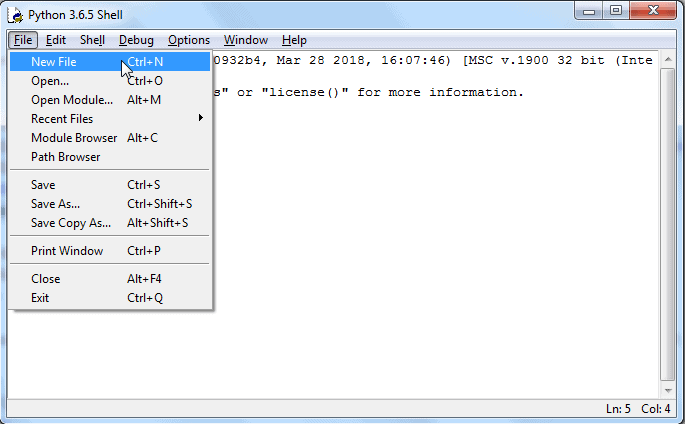
Python IDLE

Now, you can execute Python statements the same as in Python Shell as shown below.

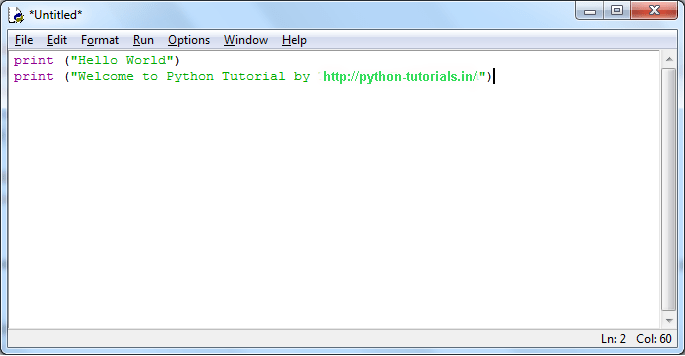
[](https://www.tutorialsteacher.com/Content/images/python/idle.gif)

Python IDLE

To execute a Python script, create a new file by selecting File -> New File from the menu.

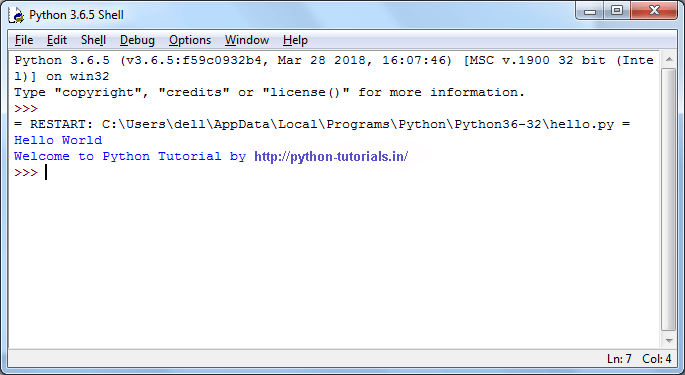
[](https://www.tutorialsteacher.com/Content/images/python/python-script-idle.png)

Enter multiple statements and save the file with extension .py using File -> Save. For example, save the following code as hello.py.

[](https://www.tutorialsteacher.com/Content/images/python/python-script-idle2.png)

Python Script in IDLE

Now, press F5 to run the script in the editor window. The IDLE shell will show the output.

[](https://www.tutorialsteacher.com/Content/images/python/python-script-idle3.png)

Python Script Execution Result in IDLE

**Task 2: Practice using interactive website - learnpython.org**

* [Hello, World!](https://www.learnpython.org/en/Hello%2C_World%21)
* [Variables and Types](https://www.learnpython.org/en/Variables_and_Types)
* [Basic Operators](https://www.learnpython.org/en/Basic_Operators)

**Task 3:**  **Solve the following questions:**

1. Variables in Python: Create a variable savings with the value 100.Check out this variable by typing print(savings) in the script **(Datacamp)**
2. Python as a calculator. (Perform all arithmetic operations on console/shell)
3. Suppose you have Rs.100, which you can invest with a 10% return each year. After one year, it's 100×1.1=110 rupees, and after two years it's 100×1.1×1.1=121.Add code to calculate how much money you end up with after 7 years, and print the result. **(Datacamp)**
4. Write a Python script that prints the result for 8958937768937 divided by 2851718461558. **(Hackinscience)**
5. Take two variables a and b and print three lines where:
   * The first line contains the sum of the two numbers.
   * The second line contains the difference of the two numbers (first - second).
   * The third line contains the product of the two numbers.
6. Take input from user in two variables and print the following:
   * Add logic to print two lines. The first line should contain the result of integer division
   * The second line should contain the result of float division **(Hackerrank)**
   * No rounding or formatting is necessary.
7. Take input from the user in variables first\_name ,last\_name and age. Print the full name of the user with age. Output should look like, e.g “Age of John Willam is 16”
8. Accept the radius of a circle and find its area and circumference.
9. Consider the following scenario:

Input :

Principle (amount): 1200

Time: 2

Rate: 5.4

Output : Compound Interest = 133.099243.

Calculate compound interest. Assume your own variables.

1. Write a program to accept 5 numbers from the user using individual statements. Print the sum, average of the numbers. Check if the sum is less than 100 and print the true false value
   * Take input from the user in two variables. Perform below operations
     + Find remainder
     + Perform floor division
     + Calculate a raise to b
2. Write a program to calculate the area of the circle.
3. Write a program to store integer value in a variable ‘a’. Perform below operations on the variable.
   * Bitwise shift left
   * Bitwise shift right

**Task 4: Simulate and Learn: (IIT VLABS)**

Simulate the following for understanding Arithmetic Operations in Python:

[https://python-iitk.vlabs.ac.in/exp/arithmetic-operations/simulation.ht](https://python-iitk.vlabs.ac.in/exp/arithmetic-operations/simulation.html)ml

**PART B**

**(PART B: TO BE COMPLETED BY STUDENTS)**

**(Students must submit the soft copy as per following segments within two hours of the practical. The soft copy must be uploaded on the Teams or emailed to the concerned lab in charge faculties at the end of the practical in case the there is no Black board access available)**

|  |  |
| --- | --- |
| Roll No. C136 | Name: Mishika Shah |
| Program: BTI | Division: D |
| Semester: IV | Batch: D1 |
| Date of Experiment: | Date of Submission: |
| Grade: |  |

**B.1 Software Code written by student:**

Task 3:

Q a)

saving = 100

print(saving)

Q b)

a = 6

b = 3

print("The addition is ", a + b)

print("The subtraction is ", a - b)

print("The multiplication is ", a \* b)

print("The division is ", a / b)

print("The power is ", a \*\* b)

print("The float division is ", a // b)

Q c)

a = 100

b = 100 \* (1.1 \*\* 7)

print("The money we end up after 7 years is",b )

Q d)

a = 8958937768937

b = 2851718461558

print("The division of the two numbers is ",a / b)

Q e)

a = 5

b = 3

print("The sum of the two numbers is ", a + b)

print("The difference between the two numbers is ", a - b)

print("The product of the two numbers is ", a \* b)

Q f)

number1 = int(input("Enter the first number : "))

number2 = int(input("Enter the second number : "))

print("The integer division between the two numbers are : ", number1 // number2)

print("The float division between the two numbers are : ", number1 / number2)

Q g)

first\_name = input("Enter your first name: ")

last\_name = input("Enter your last name: ")

age = int(input("Enter your age: "))

print("Age of ", first\_name, last\_name," is ", age)

Q h)

radius = int(input("Enter the radius of the circle: "))

pi = 3.14

print("The area of the circle is ", pi \* radius \* radius)

print("The circumference of the circle is ", 2 \* pi \* radius)

Q i)

principle\_amount = float(input("Enter the principle amount: "))

rate = float(input("Enter the rate: "))

time = float(input("Enter the time: "))

print("The compound interest is ", principle\_amount \* ((1 + (rate / 100)) \*\* time))

Q j) 1)

num1 = int(input("Enter the 1st number: "))

num2 = int(input("Enter the 2nd number: "))

num3 = int(input("Enter the 3rd number: "))

num4 = int(input("Enter the 4th number: "))

num5 = int(input("Enter the 5th number: "))

sum = num1 + num2 + num3 + num4 + num5

average = (num1 + num2 + num3 + num4 + num5) / 5

print("The sum of all the numbers is ", sum)

print("The average of all the number is ", average)

if sum < 100 :

print("The sum is less than 100, so TRUE")

else :

print("The sum is greater than 100, so FALSE")

Q j) 2)

num1 = int(input("Enter the 1st number: "))

num2 = int(input("Enter the 2nd number: "))

print("The remainder is ", num1 % num2)

print("The floor division is ", num1 // num2)

print("The answer for the 1st number raised to 2nd number is ", num1 \*\* num2)

Q k)

radius = int(input("Enter the radius of the circle : "))

pi = 3.14

print("The area of the circle is ", pi \* radius \* radius)

B.2 Input and Output:

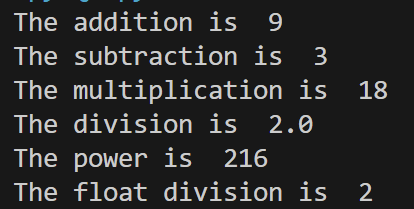
*(Paste your program input and output in following format. If there is error then paste the specific error in the output part. In case of error with due permission of the faculty extension can be given to submit the error free code with output in due course of time. Students will be graded accordingly.)*

Task 3:

Q a):



Q b):



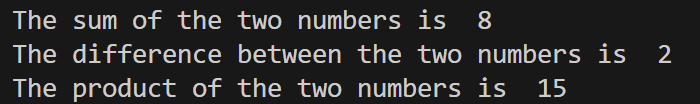
Q c):



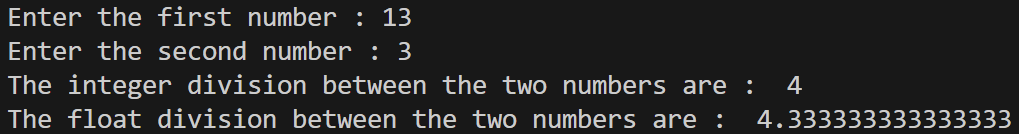
Q d):



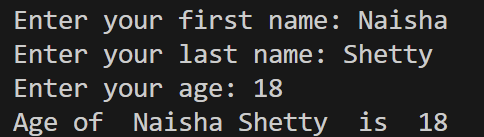
Q e):



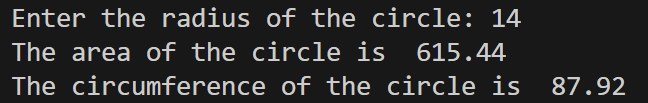
Q f):



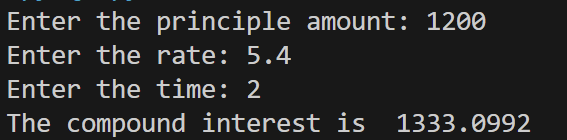
Q g):



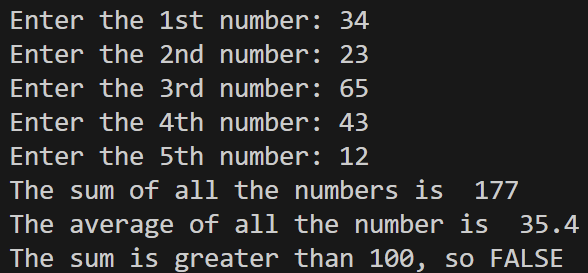
Q h):



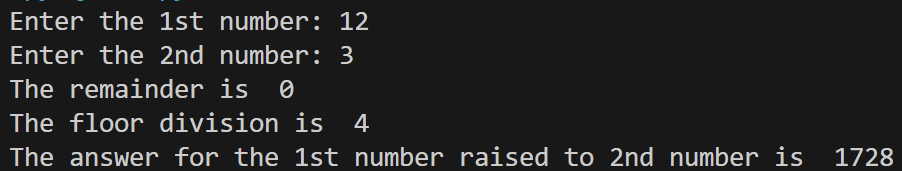
Q i):



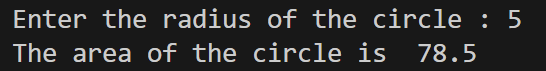
Q j)1):



Q j)2):



Q k):



B.3 Conclusion:

I learnt the basics of Python programming.