LBYCPA1

Programming Logic and Design Laboratory

Laboratory Module 1

Introduction to Python Programming

By

John Carlo Theo S. Dela Cruz || LBYCPA1-EQ1

INTRODUCTION

The 1st module of LBYCPA1 is an introduction and utilization of ***Python, Jupyter Notebook*** and how can we utilize these programs to take this with us in our journey to learning Python Programming. We were first introduced with Jupyter Notebook, in this web-based platform we could perform Python live codes and equations as well as taking notes during discussions. This platform will be the playground for Computer Engineering / Science students, in which we conduct the experiment as well as documenting every analyzation regarding our modules.

The primary job of this Lab Report is for us to understand, coding and programming works. how Python is on the top programming languages, The reason why it is one of the top programming languages because it emphasizes readability, it makes coding easy in the language of Python, and it is more object-oriented. Python was introduced to us as an interactive shell and python script that we can use as a calculator, and it gives us the idea how arithmetic function works, and basics of Input and Output. The ***primary goal*** of starting something new is by being knowledgeable and practice the basics of coding and programming, this module is about getting to know about the language, what are the operations used in python? What is the function of Input and Outputs? How can we solve the following problems with the use of arithmetic equations only? Every problem has an answer and solution once the sets are finished.

(a) Objectives

1. To understand basic knowledge about how programming works.
2. To implement of programming knowledge to real life application.
3. To develop a program using Python.
4. To create a strong foundation in programming.
5. To be able to learn the different syntax to use with Python.
6. To learn how to use the Jupyter Notebook.
7. To learn about arithmetic operation in Python.
8. To understand the concept of Algorithms, Pseudocodes and Flowchart.
9. To solve various problems using Python.
10. To diagram our own flowcharts about our code.

(b) Materials and Tools

1. Instructor's lecture notes
2. Modules
3. Jupyter Notebook
4. Flowchart Software – Diagrams.net
5. Google

PROCEDURES (*Individual*) / EXPERIMENTAL PLAN

Before proceeding to write and run the code, it is important to read and understand the problem before initializing the code. A plan must be properly prepared and looked on before coding, like preparing before an attack. The process of making algorithms, pseudocodes, and flowcharts which depend on the interpretation and output.

Familiarization Exercise 1: 'aboutme.py' – In this exercise, we will be familiarizing on how to execute a script in Anaconda Prompt.

a. Copy the text/code given in the Jupyter Notebook (Preliminary Report 1).

b. Open a blank notepad.

   i. Paste the code

   ii. Modify the required text

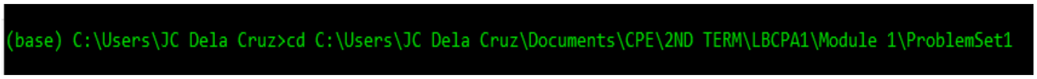
   iii. Save as 'aboutme.py', make sure to end the file with the '.py' to format it into a python file

c. Open CMD.exe prompt on the Anaconda Navigator

d. Find the location of your 'aboutme.py' file.

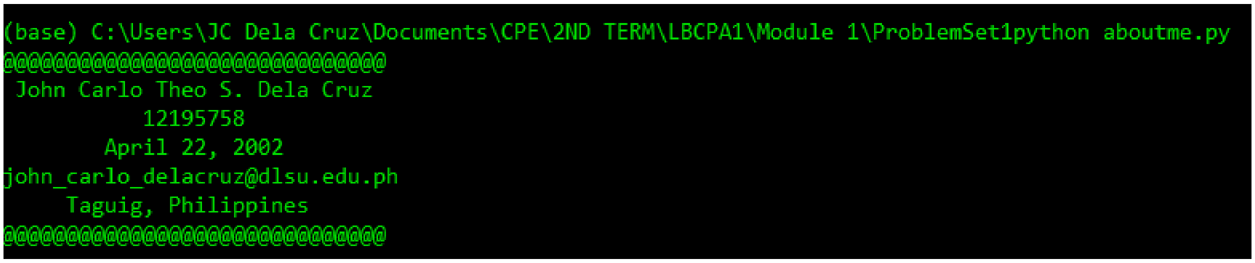
   i. Copy the directory

e. Use the command cd to change the directory and paste the directory of where the 'aboutme.py' is located.



```
(base) C:\Users\JC Dela Cruz>cd C:\Users\JC Dela Cruz\Documents\CPE\2ND TERM\LBCPA1\Module 1\ProblemSet1
```

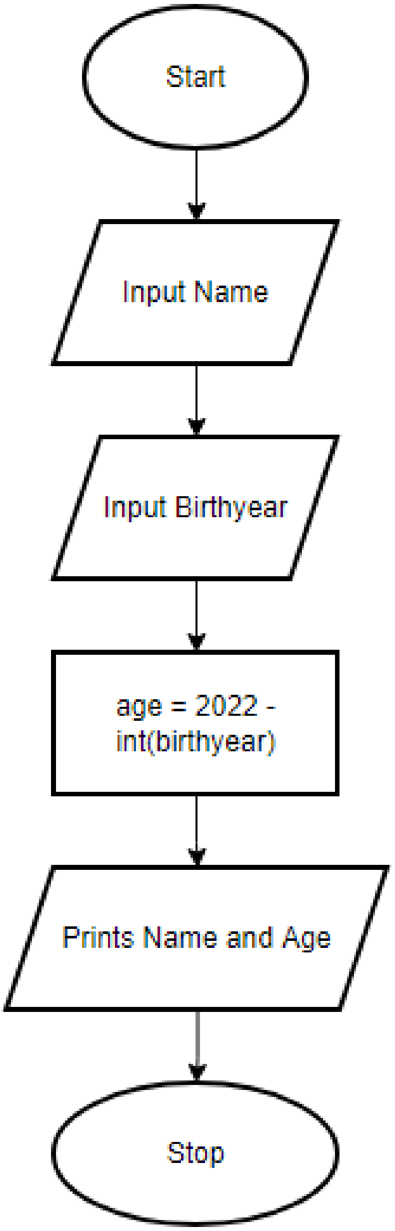
f. Type "python" and the file name "aboutme.py"



```
(base) C:\Users\JC Dela Cruz\Documents\CPE\2ND TERM\LBCPA1\Module 1\ProblemSet1python aboutme.py
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
 John Carlo Theo S. Dela Cruz
           12195758
        April 22, 2002
john_carlo_delacruz@dlsu.edu.ph
     Taguig, Philippines
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
```

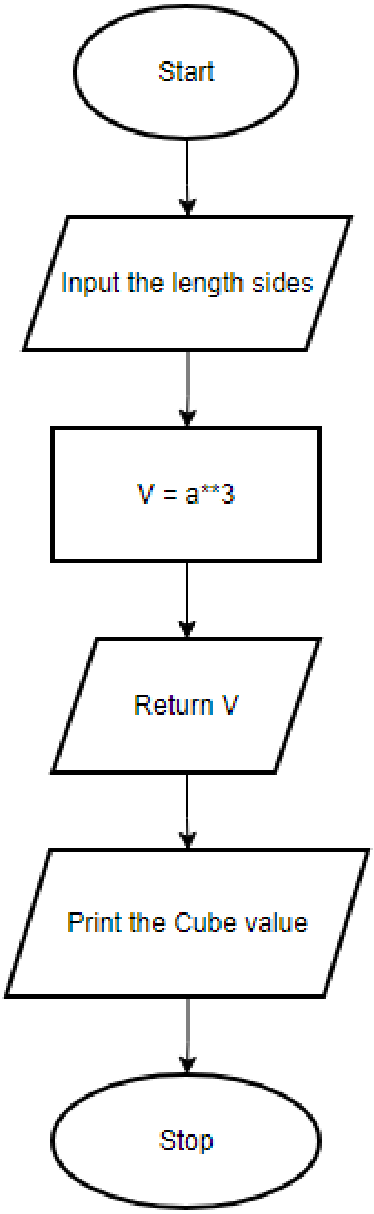
2. **Familiarization Exercise 2: Input and Output**

   a. Creating a simple input and output program that requires the user to input their name, birthyear and the program prints out the name that the user inputted and their age.

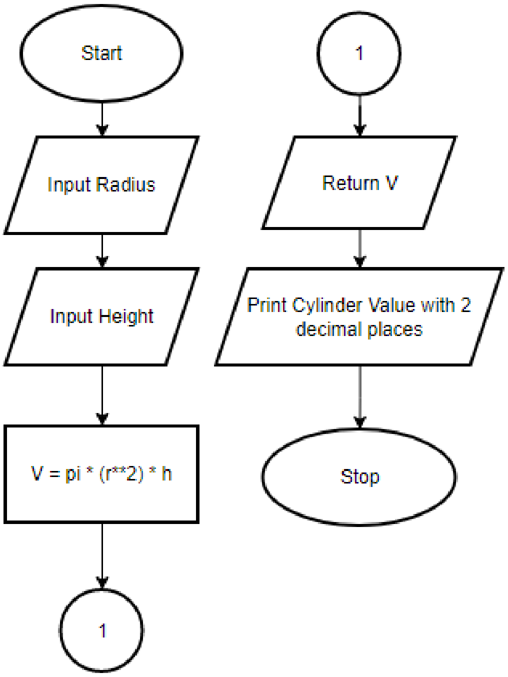


3. **Familiarization Exercise 3:** $Cube\ V = a^3$

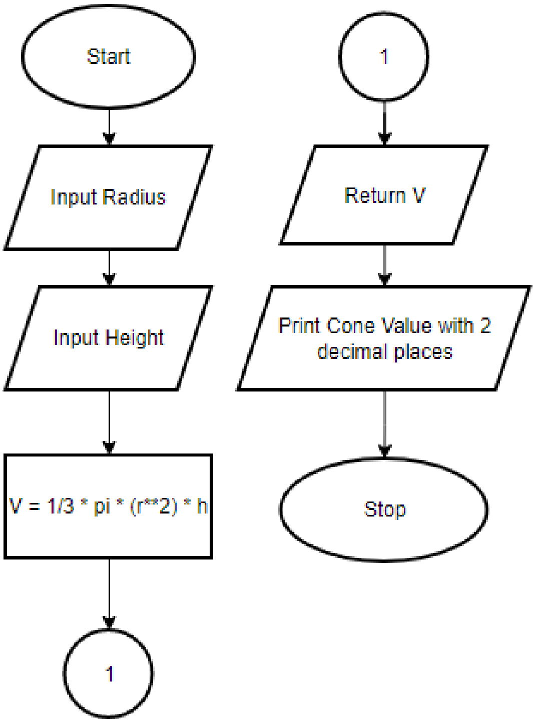
   a. Creating a program that calculates the various Mathematical Equations; Cube – Cylinder – Cone – Sphere.



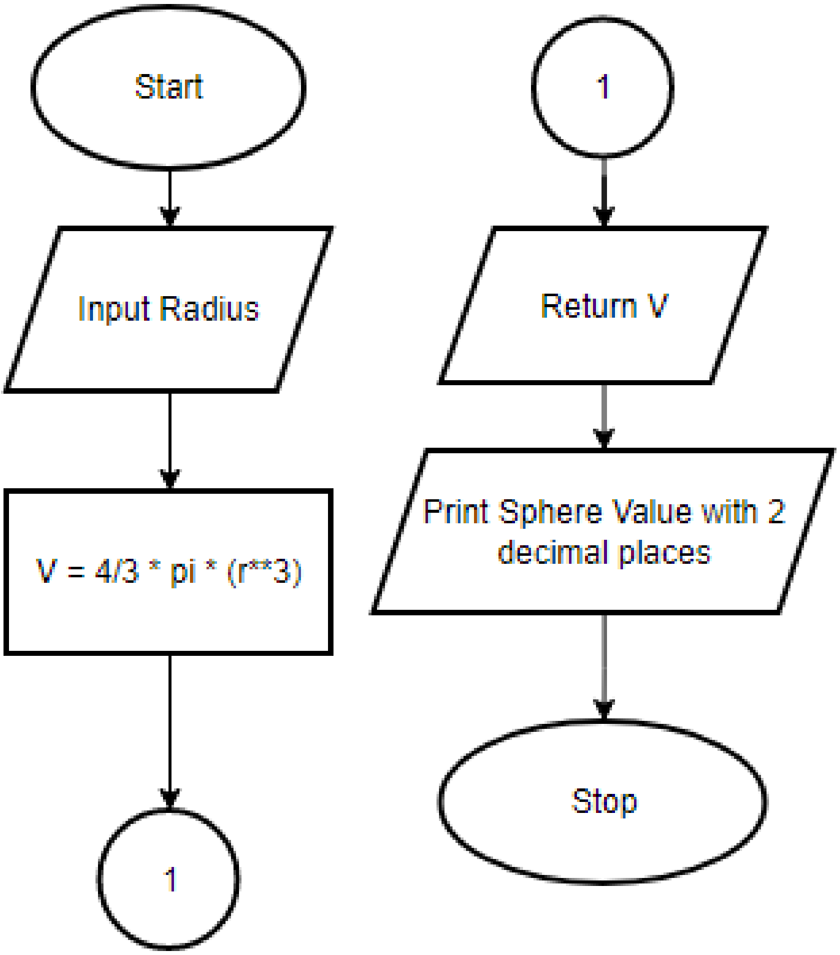
4. **Familiarization Exercise 4:** $Cylinder\ V = \pi r^2 h$



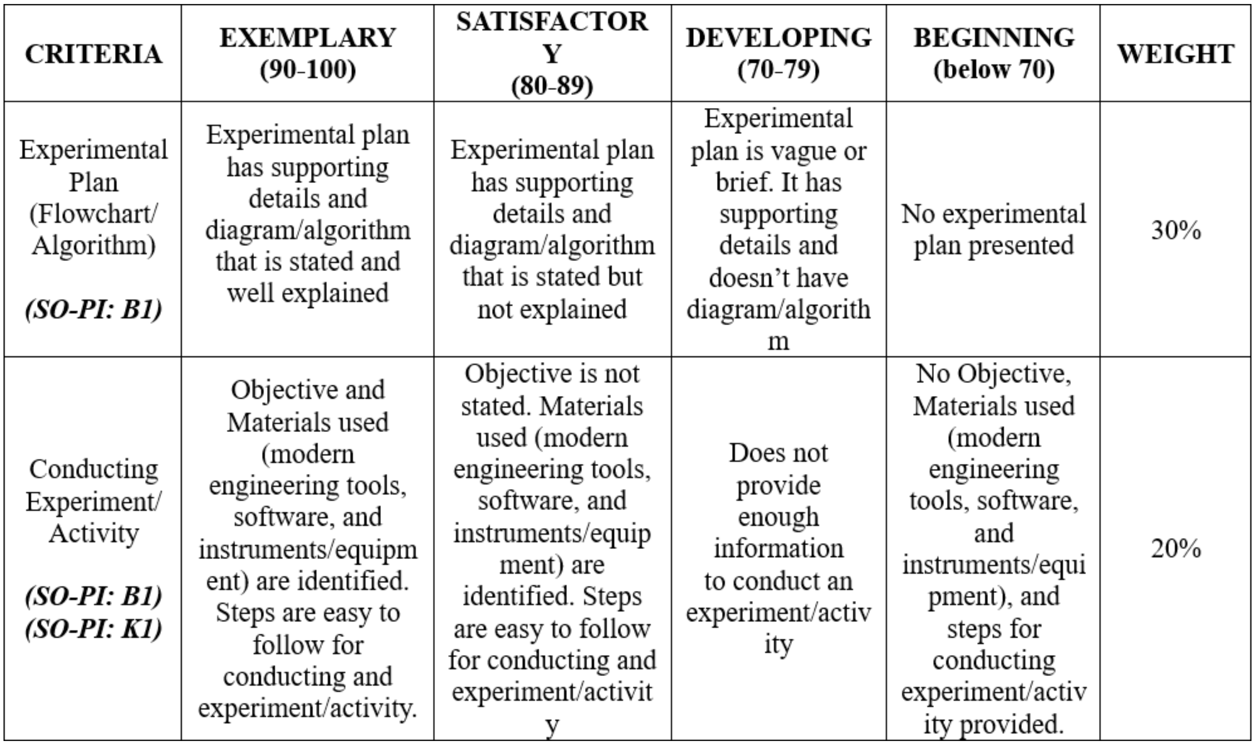
5. **Familiarization Exercise 5:** $Cone\ V = 1/3\pi r^2 h$


Start
Input Radius
Input Height
V = 1/3 * pi * (r**2) * h
1
1
Return V
Print Cone Value with 2 decimal places
Stop

6. **Familiarization Exercise 6: $Sphere\ V = 4/3\pi r^3$**



The Introduction together with individual Procedures and plan comprises the Experimental Plan and Conducting Experiment/ Activity criteria in the Final Laboratory Report Rubric:



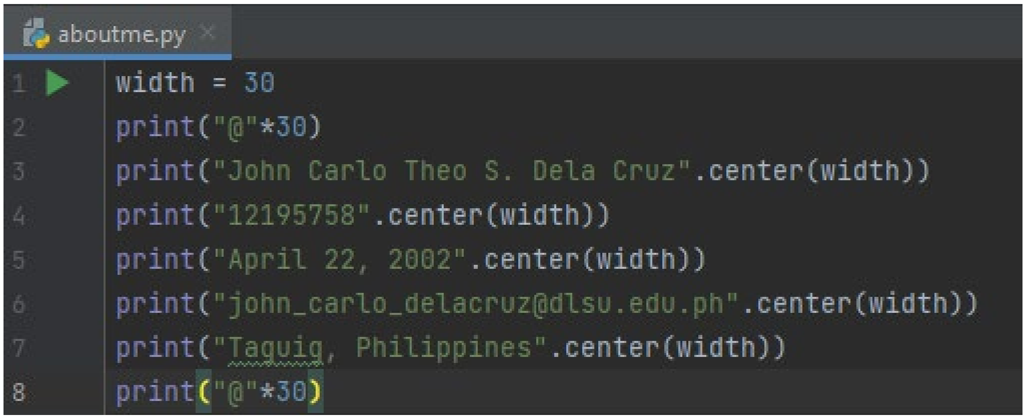
CRITERIA	EXEMPLARY (90-100)	SATISFACTORY (80-89)	DEVELOPING (70-79)	BEGINNING (below 70)	WEIGHT
Experimental Plan (Flowchart/ Algorithm) ***(SO-PI: B1)***	Experimental plan has supporting details and diagram/algorithm that is stated and well explained	Experimental plan has supporting details and diagram/algorithm that is stated but not explained	Experimental plan is vague or brief. It has supporting details and doesn't have diagram/algorithm	No experimental plan presented	30%
Conducting Experiment/ Activity ***(SO-PI: B1)*** ***(SO-PI: K1)***	Objective and Materials used (modern engineering tools, software, and instruments/equipment) are identified. Steps are easy to follow for conducting and experiment/activity.	Objective is not stated. Materials used (modern engineering tools, software, and instruments/equipment) are identified. Steps are easy to follow for conducting and experiment/activity	Does not provide enough information to conduct an experiment/activity	No Objective, Materials used (modern engineering tools, software, and instruments/equipment), and steps for conducting experiment/activity provided.	20%

RESULTS AND DISCUSSION/COMPUTATIONS (*Include the program output screenshots, and discussions per problem solution*)

1. Familiarization Exercise 1 Result: 'aboutme.py'

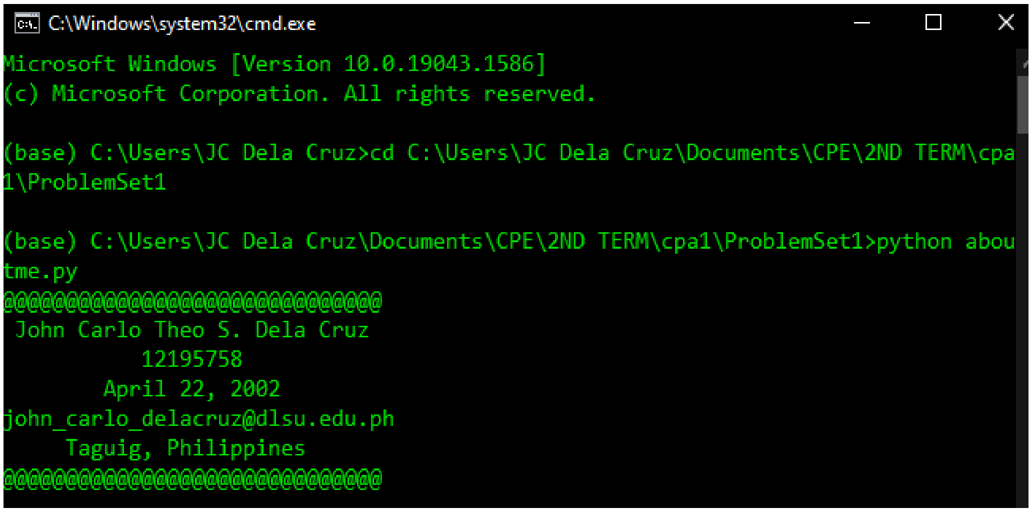
- The program's goal is to print out my personal details and execute that script in the Anaconda Prompt. First thing to do is to copy the Python code and paste it into a blank notepad page. Once you have pasted the code, modify the required text and I inputted my personal information inside the quotation marks. After inputting that information, I saved that file into 'aboutme.py'. We need to indicate the '.py' to convert it into a python file. Find the location of the 'aboutme.py' and I copied the directory in the address bar of my file explorer. Launch the CMD.exe prompt on the Anaconda Navigator and a black box will appear and all I need to do is to type 'cd' to change the directory and paste the directory that we copied a while ago. After pasting the directory, type "python 'aboutme.py'" to enable the python version and execute the file in the Command Prompt.

- **Code:**



```
width = 30
print("@"*30)
print("John Carlo Theo S. Dela Cruz".center(width))
print("12195758".center(width))
print("April 22, 2002".center(width))
print("john_carlo_delacruz@dlsu.edu.ph".center(width))
print("Taguig, Philippines".center(width))
print("@"*30)
```

- **Result:**

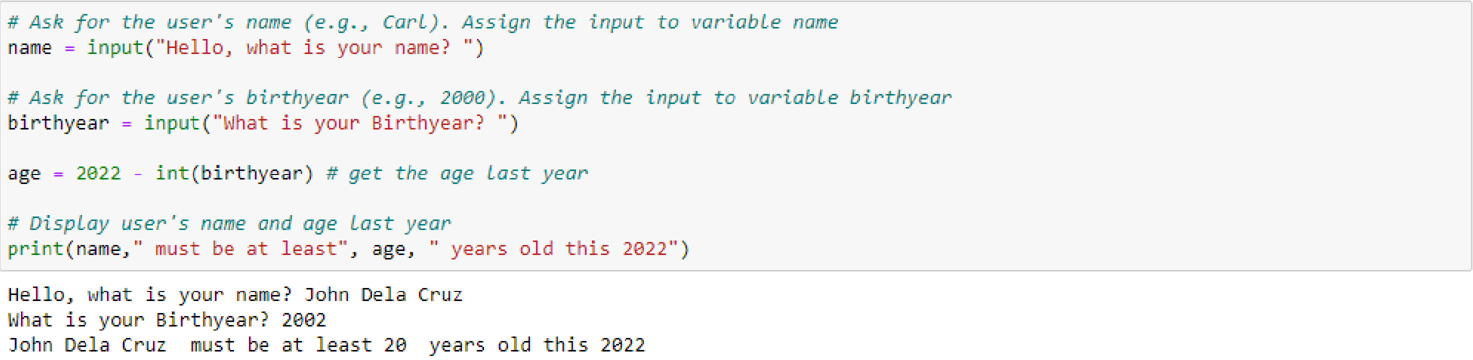


```
Microsoft Windows [Version 10.0.19043.1586]
(c) Microsoft Corporation. All rights reserved.

(base) C:\Users\JC Dela Cruz>cd C:\Users\JC Dela Cruz\Documents\CPE\2ND TERM\cpa
1\ProblemSet1

(base) C:\Users\JC Dela Cruz\Documents\CPE\2ND TERM\cpa1\ProblemSet1>python abou
tme.py
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
 John Carlo Theo S. Dela Cruz
           12195758
        April 22, 2002
john_carlo_delacruz@dlsu.edu.ph
     Taguig, Philippines
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
```

Familiarization Exercise 2 Result: Input and Output

- This is an example of a basic input-output program, where in the user is required to input their respective names and birthyear. The main variables are ‘name’, ‘birthyear’, and ‘age’, and these variables has their own unique functions, both ‘name’ and ‘birthyear’ requires the user to input the required fields while the ‘age’ function is dependent to the variable for ‘birthyear’. The find the age is just simply subtracting the year today which is 2022, to the inputted birthyear. The last section of the code is to display the user’s name and age. In printing a unique set of characters, we should always enclose them with quotation marks, because it indicates that they are not variables.

- Code and Result:

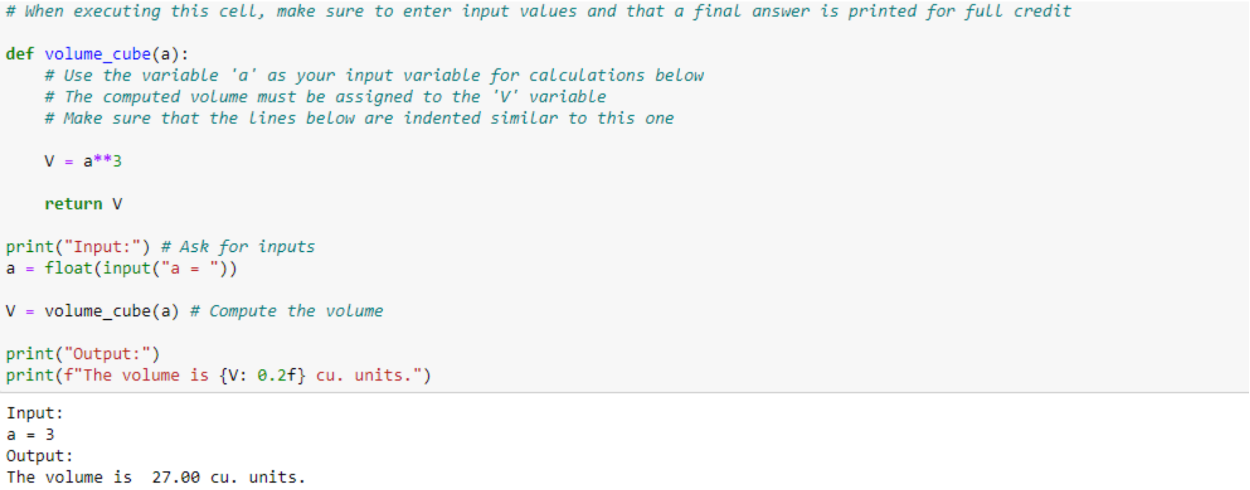


```
# Ask for the user's name (e.g., Carl). Assign the input to variable name
name = input("Hello, what is your name? ")

# Ask for the user's birthyear (e.g., 2000). Assign the input to variable birthyear
birthyear = input("What is your Birthyear? ")

age = 2022 - int(birthyear) # get the age last year

# Display user's name and age last year
print(name," must be at least", age, " years old this 2022")
```

```
Hello, what is your name? John Dela Cruz
What is your Birthyear? 2002
John Dela Cruz  must be at least 20  years old this 2022
```

Familiarization Exercise 3: Cube Equation

- This problem is also an input-output set, but it is in a form of a mathematical problem. Formula should be precise for this program to work. In this program, the user is required input a value of the side of the cube which is written as variable ‘a’. The variable a is set as an input variable for the calculations, after defining the ‘volume_cube’ below the line is where we set the variable V in which it will calculate the value of the cube.
    - The command return V will return the said variable as the computed volume is assigned.
    - Command a = float(input(“a = “)), it will require the user to input a number, and that input is set as a float. A float represents a real number and is written with decimal values.
    - Command V = volume_cube(a) is used to confirm the declared variable from the start of the program, and it will be moved to V to make sure that the value of V is computed.
- Last few lines of the code are about printing out the value of the cube. The reason why the output has 2 significant digits to the right only is because of the string formatting. {0.2f} tells Python to put as much whole numbers but only 2 significant decimal values.
- Code and Result:

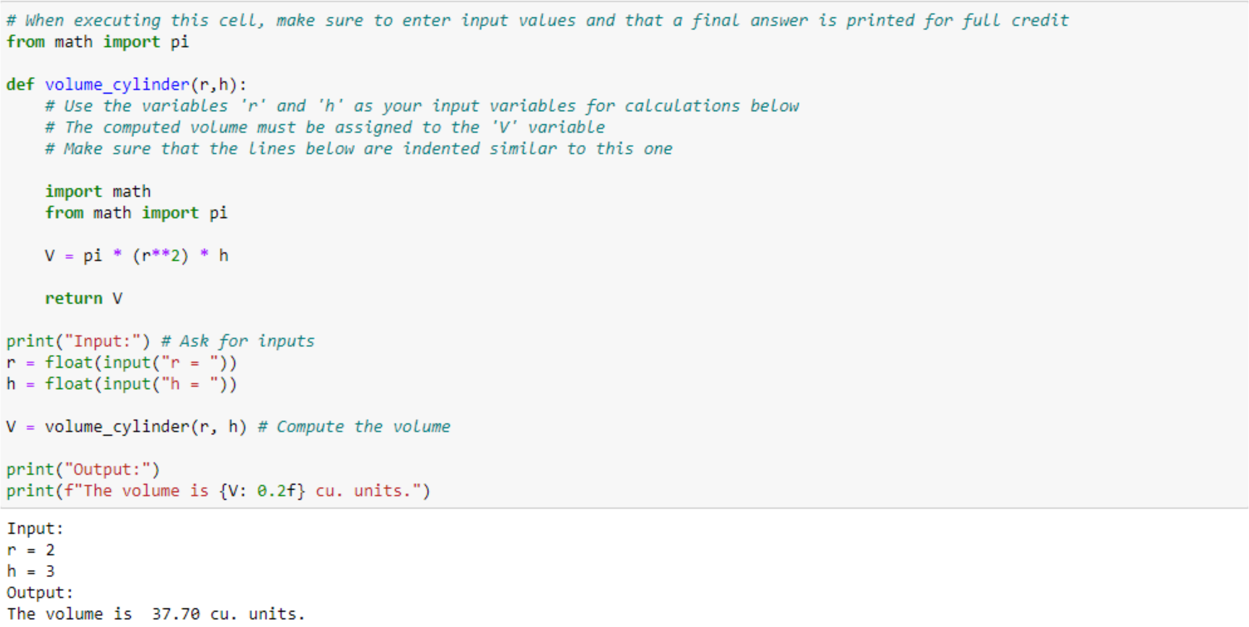


```
# When executing this cell, make sure to enter input values and that a final answer is printed for full credit

def volume_cube(a):
    # Use the variable 'a' as your input variable for calculations below
    # The computed volume must be assigned to the 'V' variable
    # Make sure that the lines below are indented similar to this one

    V = a**3

    return V

print("Input:") # Ask for inputs
a = float(input("a = "))

V = volume_cube(a) # Compute the volume

print("Output:")
print(f"The volume is {V: 0.2f} cu. units.")
```

```
Input:
a = 3
Output:
The volume is  27.00 cu. units.
```

4. Familiarization Exercise 4: Cylinder Equation

- This problem is the same as exercise 3 but in this exercise, the solution requires the value of pi. Formula should be precise for this program to work, and we need to import the built-in function in Python called ‘math’. After importing the function ‘math’, I also imported the function ‘pi’, so I can easily declare the value of ‘pi’ by just inputting the word ‘pi’. Same with the previous Exercise, in which the user is required to input the radius and the height of the cylinder, which is written as variable r, and h.
    - The command return V will return the said variable as the computed volume is assigned.
    - Command a = float(input(“r = “)), it will require the user to input a number, and that input is set as a float. A float represents a real number and is written with decimal values.
    - Command V = volume_cylinder(r, h) is used to confirm the declared variable from the start of the program, and it will be moved to V to make sure that the value of V is computed.
- Last few lines of the code are about printing out the value of the cylinder is the same as exercise 3. The reason why the output has 2 significant digits to the right only is because of the string formatting. {0.2f} tells Python to put as much whole numbers but only 2 significant decimal values.
- Code and Result:



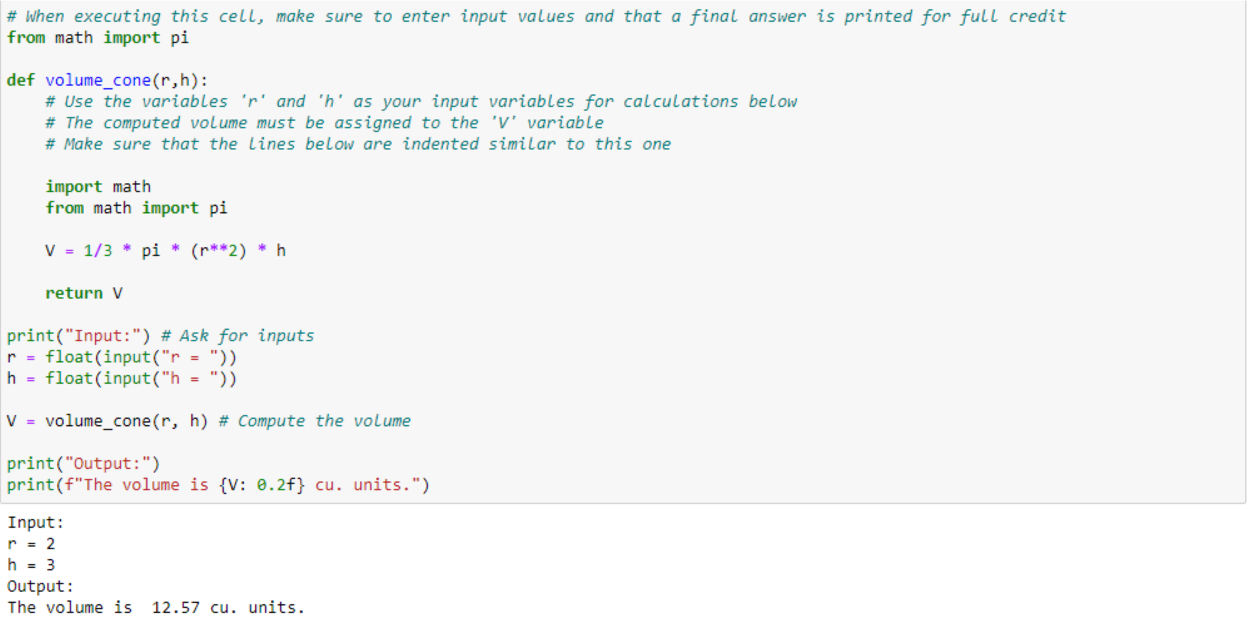
```
# When executing this cell, make sure to enter input values and that a final answer is printed for full credit
from math import pi

def volume_cylinder(r,h):
    # Use the variables 'r' and 'h' as your input variables for calculations below
    # The computed volume must be assigned to the 'V' variable
    # Make sure that the lines below are indented similar to this one

    import math
    from math import pi

    V = pi * (r**2) * h

    return V

print("Input:") # Ask for inputs
r = float(input("r = "))
h = float(input("h = "))

V = volume_cylinder(r, h) # Compute the volume

print("Output:")
print(f"The volume is {V: 0.2f} cu. units.")
```

```
Input:
r = 2
h = 3
Output:
The volume is  37.70 cu. units.
```

Familiarization Exercise 5: Cone Equation

- The problem in this exercise is almost the same in exercise 4 in terms of the formula used is similar for solving the value of the cone. The only difference in the formula is that the equation should be multiplied by one-third. We need to import the built-in function ‘math’ because we will be needing the value of pi again. Same functions defining the volume cone, execute the code with the formula and returning the value of V after inputting the formula. In terms of output, we will be using the string formatting so there will be only 2 significant figures to the right.

- Code and Results:

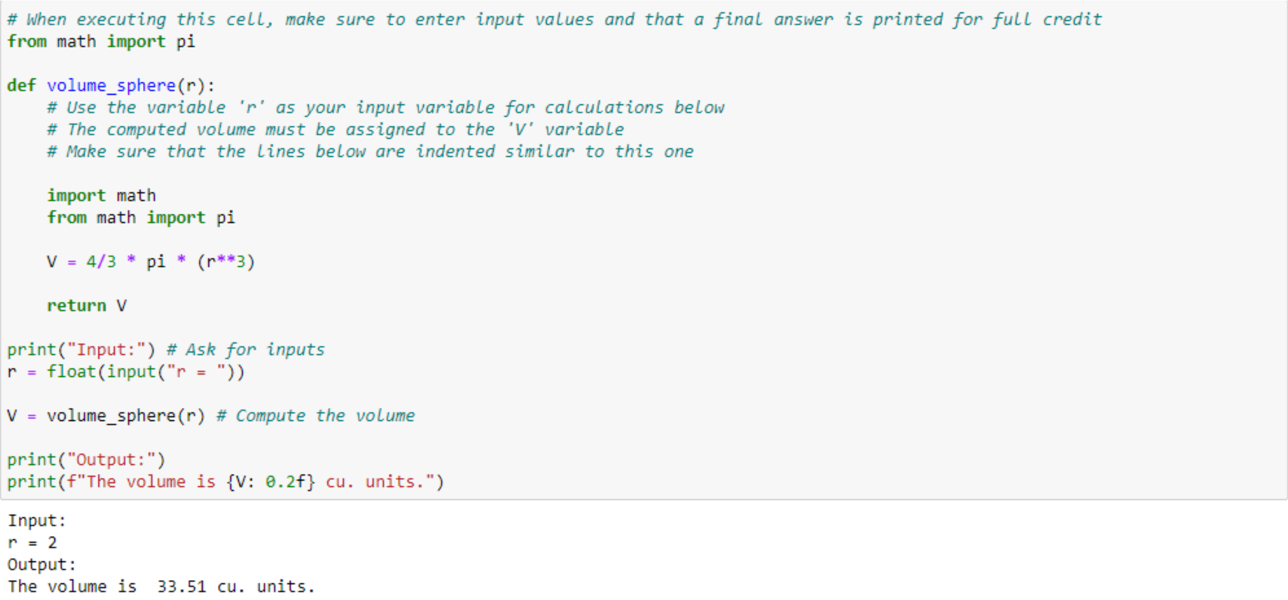


```
# When executing this cell, make sure to enter input values and that a final answer is printed for full credit
from math import pi

def volume_cone(r,h):
    # Use the variables 'r' and 'h' as your input variables for calculations below
    # The computed volume must be assigned to the 'V' variable
    # Make sure that the lines below are indented similar to this one

    import math
    from math import pi

    V = 1/3 * pi * (r**2) * h

    return V

print("Input:") # Ask for inputs
r = float(input("r = "))
h = float(input("h = "))

V = volume_cone(r, h) # Compute the volume

print("Output:")
print(f"The volume is {V: 0.2f} cu. units.")
```

```
Input:
r = 2
h = 3
Output:
The volume is  12.57 cu. units.
```

Familiarization Exercise 6: Sphere Equation

- The final exercise is determining the volume of a sphere, which is quite a combination of problems 4 and 5. Same process, but we must combine the things we learned from the previous problems. Using the built-in function ‘math’, to get the constant value of pi and multiplying it 4/3 to get the volume of the sphere.
- Code and Result:



```
# When executing this cell, make sure to enter input values and that a final answer is printed for full credit
from math import pi

def volume_sphere(r):
    # Use the variable 'r' as your input variable for calculations below
    # The computed volume must be assigned to the 'V' variable
    # Make sure that the lines below are indented similar to this one

    import math
    from math import pi

    V = 4/3 * pi * (r**3)

    return V

print("Input:") # Ask for inputs
r = float(input("r = "))

V = volume_sphere(r) # Compute the volume

print("Output:")
print(f"The volume is {V: 0.2f} cu. units.")
```

```
Input:
r = 2
Output:
The volume is  33.51 cu. units.
```

Codes/Data/ Program	Data is well utilized in the program. Program codes are easy to read. Program output has no error. Questions are answered completely and correctly	Data is somewhat utilized in the program. Program code are easy to read. Program output has an output but logically incorrect. Some questions are answered completely and correctly	Data is not utilized in the program. It has a missing significant code/syntax in the program.	No program presented	30%

CONCLUSION:

This module's goal is for us to build a foundation on how to write a code. In this case, we are using one of the best and one of the easiest programming languages which is Python. It is quite easy to understand, and I know we will tackle more challenges and complicated problems in the future, but this foundation will make us, future software developers to be independent and competent in our work.

In this module I learned and somehow mastered the use of input and output, arithmetic equations and being familiarized with the syntax or the rules on writing and executing the code in Python. Another objective is to learn about how to use arithmetic equations to solve mathematical questions that completely works in python. For people who are confused in this module, I would suggest them to re-watch, read and to know by heart the things in Module 1, because every introductory lesson will be the foundation for every coder. Practicing and experimenting would also be my recommendation so you can fully understand how input-output and how to use arithmetic equations in Python.

The rest of the rubric criteria are as follows:

Grammar, logical presentation, and format ***(SO-PI: G1)***	The report was grammatically correct, logically presented and used the required format.	The report had minimal grammatical errors and somewhat presented logically. The required format was used.	The report had a lot of grammatical errors and not logically presented; the required format was barely used.	The report had a lot of grammatical errors, was not logically presented and the required format was not used.	20%

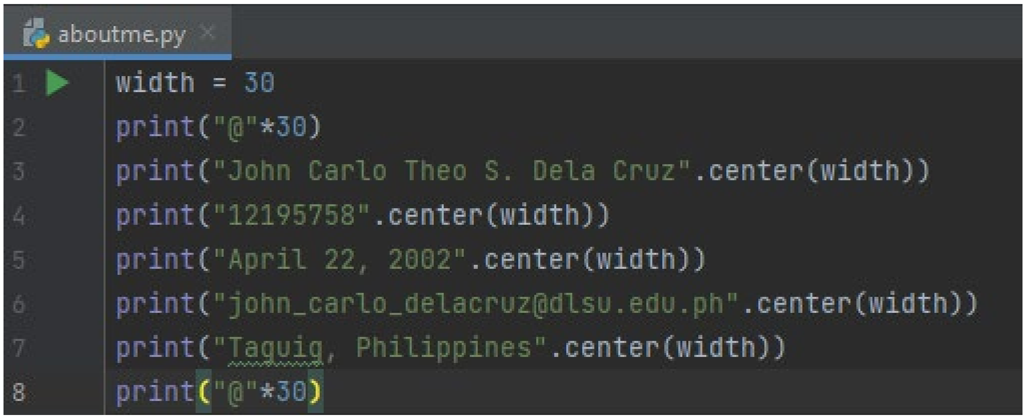
REFERENCES (*Enumerate references in APA format*)

stackOverflow (n.d.). Retrieved from. https://stackoverflow.com/

w3school (n.d.). Retrieved from. https://www.w3schools.com/python/python_intro.asp

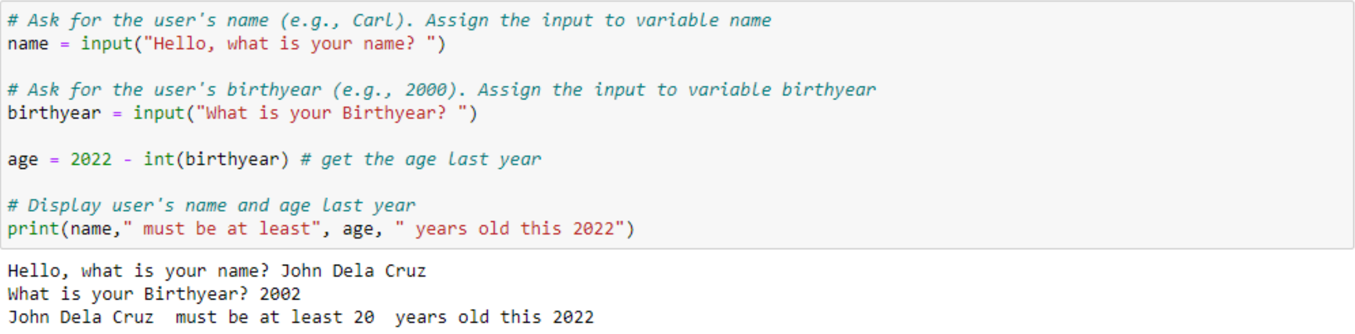
APPENDIX (*Attach all the source codes here per problem category*)

1. Familiarization Exercise 1:



```
width = 30
print("@"*30)
print("John Carlo Theo S. Dela Cruz".center(width))
print("12195758".center(width))
print("April 22, 2002".center(width))
print("john_carlo_delacruz@dlsu.edu.ph".center(width))
print("Taguig, Philippines".center(width))
print("@"*30)
```

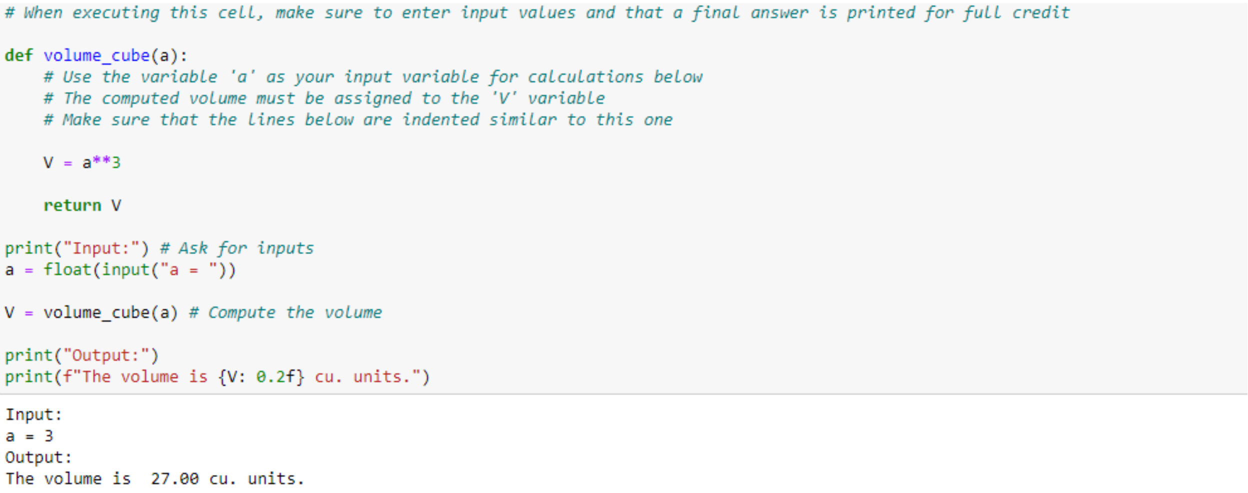
2. Familiarization Exercise 2:



```
# Ask for the user's name (e.g., Carl). Assign the input to variable name
name = input("Hello, what is your name? ")

# Ask for the user's birthyear (e.g., 2000). Assign the input to variable birthyear
birthyear = input("What is your Birthyear? ")

age = 2022 - int(birthyear) # get the age last year

# Display user's name and age last year
print(name," must be at least", age, " years old this 2022")
```

```
Hello, what is your name? John Dela Cruz
What is your Birthyear? 2002
John Dela Cruz  must be at least 20  years old this 2022
```

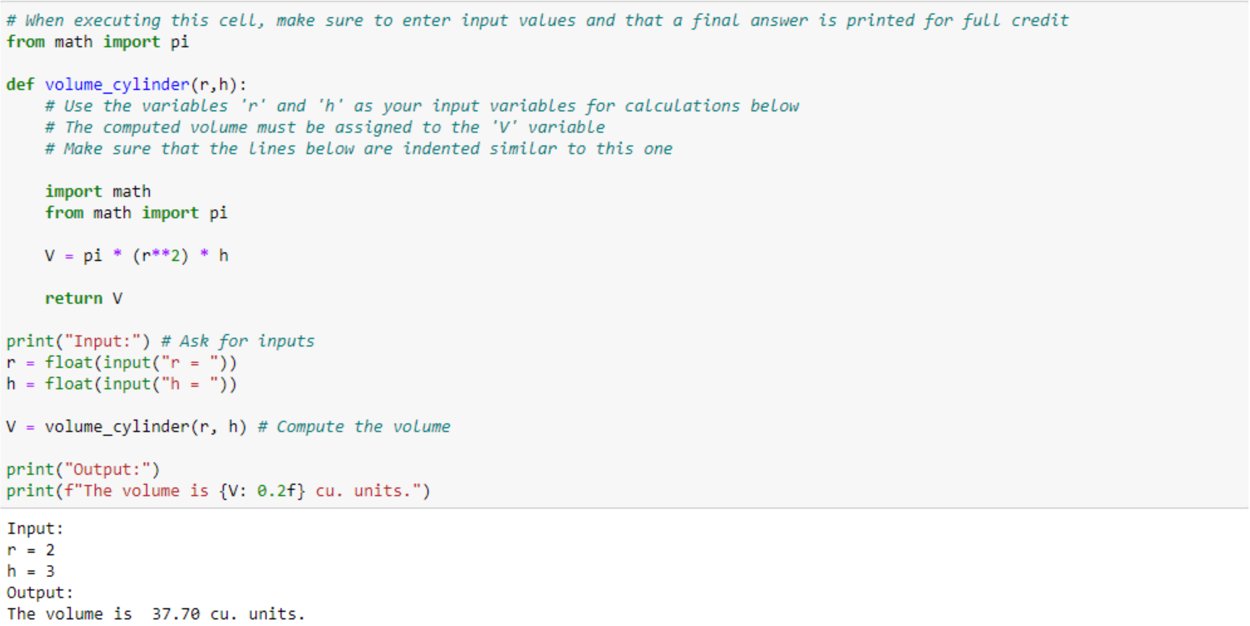
3. Familiarization Exercise 3:



```
# When executing this cell, make sure to enter input values and that a final answer is printed for full credit

def volume_cube(a):
    # Use the variable 'a' as your input variable for calculations below
    # The computed volume must be assigned to the 'V' variable
    # Make sure that the lines below are indented similar to this one

    V = a**3

    return V

print("Input:") # Ask for inputs
a = float(input("a = "))

V = volume_cube(a) # Compute the volume

print("Output:")
print(f"The volume is {V: 0.2f} cu. units.")
```

```
Input:
a = 3
Output:
The volume is  27.00 cu. units.
```

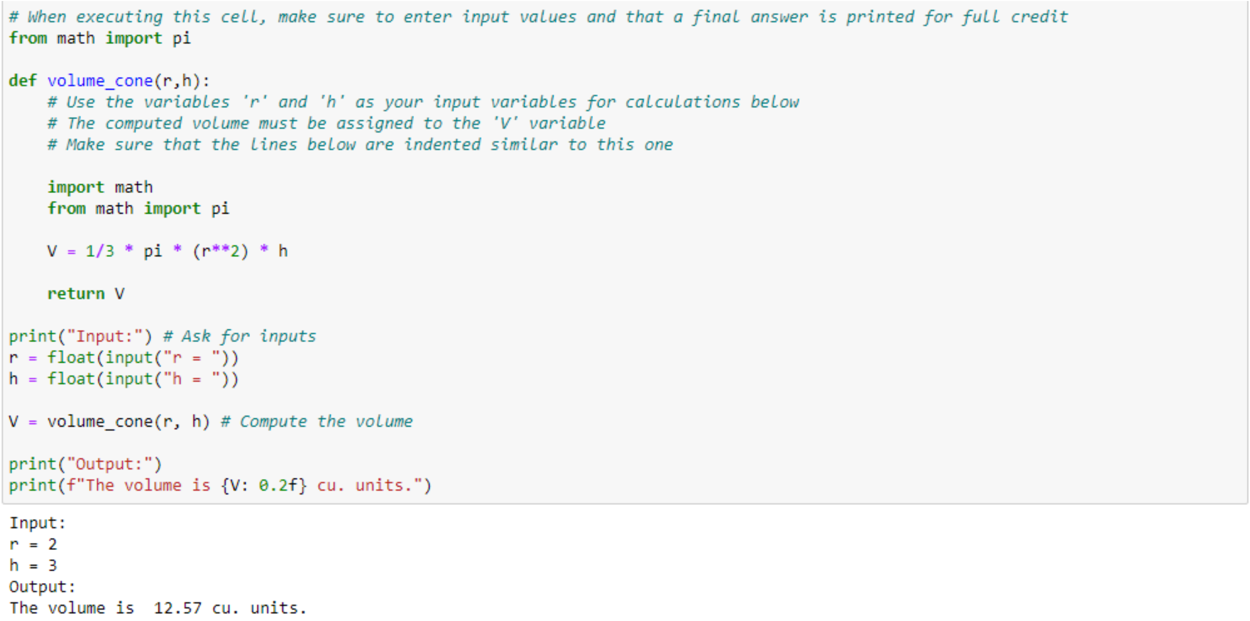
4. Familiarization Exercise 4:



```
# When executing this cell, make sure to enter input values and that a final answer is printed for full credit
from math import pi

def volume_cylinder(r,h):
    # Use the variables 'r' and 'h' as your input variables for calculations below
    # The computed volume must be assigned to the 'V' variable
    # Make sure that the lines below are indented similar to this one

    import math
    from math import pi

    V = pi * (r**2) * h

    return V

print("Input:") # Ask for inputs
r = float(input("r = "))
h = float(input("h = "))

V = volume_cylinder(r, h) # Compute the volume

print("Output:")
print(f"The volume is {V: 0.2f} cu. units.")
```

```
Input:
r = 2
h = 3
Output:
The volume is  37.70 cu. units.
```

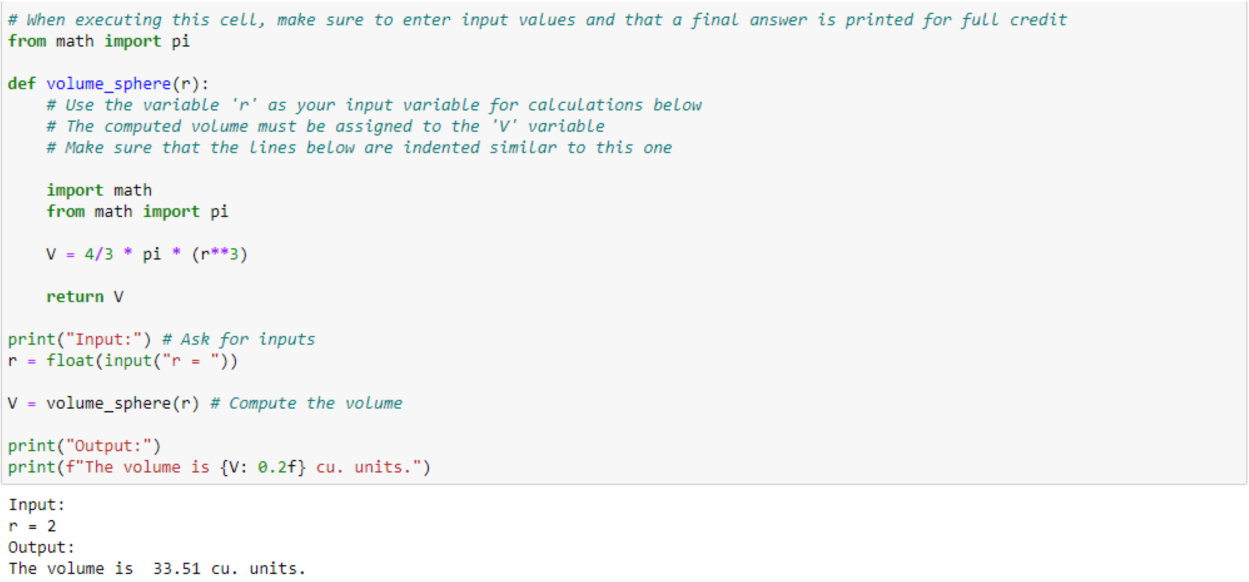
5. Familiarization Exercise 5:



```
# When executing this cell, make sure to enter input values and that a final answer is printed for full credit
from math import pi

def volume_cone(r,h):
    # Use the variables 'r' and 'h' as your input variables for calculations below
    # The computed volume must be assigned to the 'V' variable
    # Make sure that the lines below are indented similar to this one

    import math
    from math import pi

    V = 1/3 * pi * (r**2) * h

    return V

print("Input:") # Ask for inputs
r = float(input("r = "))
h = float(input("h = "))

V = volume_cone(r, h) # Compute the volume

print("Output:")
print(f"The volume is {V: 0.2f} cu. units.")
```

```
Input:
r = 2
h = 3
Output:
The volume is  12.57 cu. units.
```

6. Familiarization Exercise 6:



```
# When executing this cell, make sure to enter input values and that a final answer is printed for full credit
from math import pi

def volume_sphere(r):
    # Use the variable 'r' as your input variable for calculations below
    # The computed volume must be assigned to the 'V' variable
    # Make sure that the lines below are indented similar to this one

    import math
    from math import pi

    V = 4/3 * pi * (r**3)

    return V

print("Input:") # Ask for inputs
r = float(input("r = "))

V = volume_sphere(r) # Compute the volume

print("Output:")
print(f"The volume is {V: 0.2f} cu. units.")
```

```
Input:
r = 2
Output:
The volume is  33.51 cu. units.
```