**[ Laboratory No. 1.1:** **Water Supply Management]**

**Objectives:**

1. To know the basics about Python Programming and its concepts
2. To create a program that takes input from the user and display necessary information as required

**Materials:**

1. PC or Laptop
2. Python Package Development Kit
3. Pycharm or any IDE

**Background**

**Installing Python**

Go to www.python.org and download the latest version of **Python** (version **3.7.3** as of this writing). It should be painless to install. If you have a Mac or Linux, you may already have Python on your computer, though it may be an older version so you should install a new and latest version. For windows OS, you do the same.

**IDE** is a simple integrated development environment **(IDE)** that comes with Python. It’s a program that allows you to type in your programs and run them. There are other IDEs for Python, but for now I would suggest sticking with IDE as it is simple to use. You can find IDE in the Python **3.7.3** folder on your computer. When you first start IDE, it starts up in the shell, which is an interactive window where you can type in Python code and see the output in the same window. I often use the shell in place of my calculator or to try out small pieces of code. But most of the time you will want to open up a new window and type the program in there.

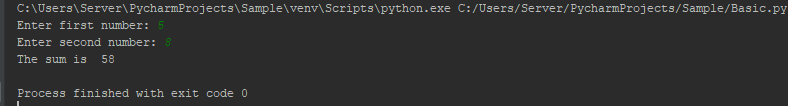
**Sample Program:**

num1 = **input**('Enter first number: ')

num2 = **input**('Enter second number: ')

**print**('The sum is ',(num1+num2))

Then, under the **Run** menu, choose Run Module (or press **F5**). IDE will ask you to save the file, and you should do so. Be sure to append **.py** to the filename as IDE will not automatically append it. This will tell IDE to use colors to make your program easier to read. Once you’ve saved the program, it will run in the shell window. The program will ask you for two numbers. Type in 5 and 8 respectively and press enter. The program’s output looks something like this:



Let’s examine how the program does what it does. The first and second lines ask the user to enter number values. The **input** function’s job is to ask the user to type something in and to capture what the user types. The part in quotes is the prompt that the user sees. It is called a string and it will appear to the program’s user exactly as it appears in the code itself.

The second line uses the **print** function to print out the conversion. The part in quotes is another string and will appear to your program’s user exactly as it appears in quotes here. The second argument to the **print** function is the calculation. Python will do the calculation and print out the numerical result.

This program may seem too short and simple to be of much use, but there are many websites that have little utilities that do similar calculations, and their code is not much more complicated than the code here.

**Instructions:**

1. Create class **WaterSupplyManagement*[Surname]***
2. **Problem Scenario**

A tank can contain 50 gallons of water and each gallon requires 5 liters. While each liter container contains 10 glasses to full. How much tanks, gallons, liters, and glasses is/are needed for an N water where N is an input supply of water.

1. **Input**

Input consist of integer N, the water supply in liter/glasses.

1. **Constraints**

**N ≥ 1**

1. **Output**

Output the Tanks, Gallons, Liters, and glasses required

1. **Source Codes**

|  |
| --- |
|  |

1. **Sample Input/Output (Atleast 3 attempts)**

**NOTE: Provide a screenshot and describe your observation for each action you performed based on the item below:**

* **Input any non-negative value for N**
* **When an N is negative value**
* **When an N is a string or contain string**

1. **Submit your file with filename convention: WaterSupplyManagement*[Surname]***

**Rules:**

1. Each laboratory activity has time limit of 1:30 minutes and is due on the day depending on the level of difficulty or constraints.
2. Each activity will only last every after 3 days and has deduction of 10 points every day from the day it was given