****

**Green University of Bangladesh**

**Department of Computer Science and Engineering(CSE)**

**Faculty of Sciences and Engineering**

**Semester: (Spring, Year:2024), B.Sc. in CSE (Day)**

**LAB REPORT NO #01**

**Course Title: Artificial Intelligence Lab**

**Course Code: CSE 316 Section: 213D3**

**Lab Experiment Name: Introduction to Basic Operations on Python.**

**Student Details**

|  |  |  |
| --- | --- | --- |
| **Name** | | **ID** |
| **1.** | Shoaib Ahmed | 212902019 |

**Lab Date : 24-02-24**

**Submission Date : 02-03-24**

**Course Teacher’s Name : Md. Zahidul Hasan**

**[For Teachers use only: Don’t Write Anything inside this box]**

|  |
| --- |
| **Lab Report Status**  **Marks: ………………………………… Signature:.....................**  **Comments:.............................................. Date:..............................** |

**1. TITLE OF THE LAB EXPERIMENT**

Introduction to Basic Operations on Python.

**2. OBJECTIVES/AIM**

* Introduce function concepts in Python for basic operations.
* Demonstrate creating reusable functions for finding the largest number and

calculating the sum of numbers.

* Practice using parameters, return values, and function organization with a main

function.

**3. PROCEDURE / ANALYSIS / DESIGN**

**Largest Number:**

**1. Input:** Define a ‘main’ function to handle user interaction.

a. Prompt the user to enter the first number (‘num1’).

b. Prompt the user to enter the second number (‘num2’).

**2. Function Call:** Call the largest\_number function, passing num1 and num2 as

arguments.

**3. Comparison:** Compare 'num1’ and ‘num2’ using an ‘if-else’ statement.

**4. Output:**

a. If ‘num1’ is greater, return ‘num1’.

b. Otherwise, return ‘num2’.

**5. Display Result:** Back in ‘main’, receive the returned value from the function call

and display it as the largest number

**The sum of Numbers:**

**1. Input:** Define a ‘main’ function to handle user interaction.

a. Initialize an empty list of ‘numbers’ to store user input.

**2. User Input Loop:** Use a ‘while True’ loop to continuously prompt for numbers.

a. Prompt the user to enter a number.

b. If the user enters an empty string (""), break out of the loop (user

finished entering numbers).

c. Otherwise, convert the user input to a number and append it to the

‘numbers’ list.

**3. Function Call:** Call the ‘sum\_of\_numbers’ function, passing the entire ‘numbers’ list as a single argument using ‘\*numbers’.

**4. Sum Calculation:** Inside the ‘sum\_of\_numbers’ function:

a. Initialize a variable ‘total\_sum’ to 0.

b. Iterate through each number in the ‘numbers’ list using a ‘for’ loop.

c. For each number:

d. Add the number to the ‘total\_sum’ (‘total\_sum += number’).

**5. Output:** Return the calculated ‘total\_sum’.

**6. Display Result:** Back in ‘main’, receive the returned value from the function call

and display it as the sum of the numbers.

**4. IMPLEMENTATION**

Code-1:

|  |
| --- |
| def largest\_number(num1, num2):  if num1 > num2:  return num1  else:  return num2  def main():  num1 = input("Enter the first number: ")  num2 = input("Enter the second number: ")  result = largest\_number(num1, num2)  print(f"The largest number is {result}")  if \_\_name\_\_ == "\_\_main\_\_":  main() |

Code-2:

|  |
| --- |
| def sum\_of\_numbers():  n = int(input("Enter the number of inputs: "))  total\_sum = 0  for i in range(n):  num = float(input(f"Enter number {i+1}: "))  total\_sum += num  return total\_sum  result = sum\_of\_numbers()  print(f"The sum of the entered numbers is: {result}") |

**5. TEST RESULT / OUTPUT**

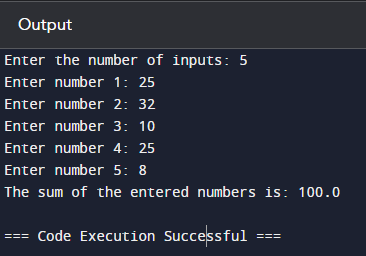
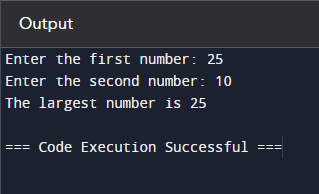


Fig -1.1: Output of largest numbers. Fig-1.2: Sum of given numbers

**6. ANALYSIS AND DISCUSSION**

In this Lab report We use two distinct functions: one to determine the largest number between two inputs and another to calculate the sum of multiple inputs. The largest\_number function compares two numbers and returns the larger one. The main function gathers two numerical inputs from the user, calls largest\_number, and prints the result. The second function, sum\_of\_numbers, prompts the user for the number of inputs, then iteratively collects these inputs, computes their sum, and prints the total. Both functions effectively demonstrate basic input handling, conditional statements, and loops in Python, offering practical examples of user interaction and fundamental operations on numeric data.