# Weekly Task Report 2

| Name | Abu Sufian Robin | | |
|---|---|---|---|
| Week 2 Task Report | | | |
| Focus Area | Task Goals Complete | Completed Task Details | Corresponding Code Files |
| Functions & Exception Handling | ☑ Learn modular programming.<br>☑ Error Handling.<br>☑ Understand scope, arguments and return values. | A Python function is a block of organized, reusable code that performs a single, related action. Functions provide better modularity for your application and a high degree of code reusing. In this code I have used Python's function concept. Where I have written two functions for array and assembly and have calculated them by calling those functions with input parameters. | *python_function.py* |
| | | I have used global scope, local scope in all these Python programs. In this problem I have used global scope, local scope concept and used different variables which are accessible outside and inside the function. Also in the Python code I have passed different methods and arguments. And in this text file I have basically shown the input output operations of a sample use case. | *Wishing_Card_CLI_Project (main.py, wish.py, use_Case.txt )* |
| | | In this problem (Python_CLI_Project) too, I have completed the work related to arguments, variables, scope, return values. Where I have basically completed the work by creating different methods in this file | *Python_CLI_Project (main.py, python_to_do_list.py )*<br>*Task list save in : tasks.json.*<br>*Use Case show : use_case.txt* |

| | | | |
|---|---|---|---|
| | | (python_to_do_list.py ) for which I have to make different arguments, return values. Apart from this, I have called all those methods from this file and completed the work using the command prompt. | |
| | | In this code (Python_Operations ) too, I have used the concepts of functions, arguments, variables, and return values of Python. Apart from this, various mathematical operations have been done using the try - except block concept. BMI calculation, CGPA calculator, operations of various objects such as area calculation of rectangle, circle, triangle have been shown. And for exception inputs, a try - except block has been used. Which is launched through the command prompt. | *Python_Operations (* main.py, operations.py, use_Case.txt *)* |

# Project Details

## Scope

Scope refers to the region where a variable is accessible. In Python, there are generally two main types of scope:

- **Global Scope:** Variables defined outside of any function are in the global scope. They can be accessed from anywhere in the program, including inside functions.

- **Local Scope:** Variables defined inside a function are in the local scope of that function. They are only accessible within that specific function and cease to exist once the function finishes execution.

## Arguments

Arguments are the values passed into a function when it is called. They allow you to provide input data to the function, enabling it to perform its task with specific values.

- **Parameters:** These are the names defined in the function signature that act as placeholders for the arguments the function expects to receive.
- **Arguments:** These are the actual values passed to the function when it is invoked.

## Return Values

The return statement is used to send a value (or multiple values) back from a function to the part of the code that called it.

- A function can return any Python object, such as numbers, strings, lists, or even other functions.
- If a function does not explicitly use a return statement, it implicitly returns None.
- Any code after a return statement within a function will not be executed.

## Try - Except Concept

In Python, try and except blocks are used for handling exceptions, which are errors that occur during the execution of a program. This mechanism allows for robust code that can gracefully manage unexpected situations instead of crashing.

- **try block:** This block contains the code that is potentially "risky" and might raise an exception.
- **except block(s):** These blocks specify how to handle particular types of exceptions. If an exception listed in an except clause occurs within the try block, the corresponding except block's code is executed. You can have multiple except blocks to handle different

types of exceptions, or a single generic except Exception as e: to catch any unhandled exception.

So I basically created a **Command Prompt Interface** (CLI) project using try-except blocks, functions, methods, variables for these problems (***Wishing_Card_CLI_Project, Python_CLI_Project , Python_Operations*** ). And I have shown the use cases of all those projects in text files. I have handled the complexity of the programs using try-except blocks and exception cases.