PROJECT REPORT

Week 1 task:--

Develop a text-based adventure game using Python. The game should be interactive, where users make choices to navigate through different scenarios. The game will include features such as multiple paths, inventory management, and decision-based outcomes. Implement basic Python concepts like loops, conditionals, and functions to create a rich user experience.

Structure:

-User Interface: Design a console-based user interface for input and output, allowing users to interact with the calculator using simple commands or GUI using libraries like Tkinter (optional).

-Functionality: Implement features to perform basic arithmetic operations, error handling for invalid inputs, and support for decimal or integer inputs.

-Reports: Generate reports or logs of performed operations (optional) and save them into a text file or a database.

Technologies:

-Python: For developing the application's core logic and features.

-Tkinter (Optional): For building a GUI-based calculator with buttons, input fields, and result displays.

-SQLite (Optional): For saving the history of operations performed by the user.

Optional:

-SQLite: For persistent data storage to save the user's history of calculations.

-Pandas (Optional): For storing the history in a table format and generating reports for operations performed.

-Logging: For generating logs of calculations performed and storing them for later reference.

Deliverables:

-Complete Python project files (including source code and necessary configuration files).

-A Python executable or script for running the application.

-Documentation of the application’s functionality, usage, and setup.

Checklist:

-Design the system for arithmetic operations with error handling for invalid inputs (e.g., dividing by zero).

-Implement functionality for performing basic operations (addition, subtraction, multiplication, division).

-Optionally implement GUI using Tkinter for a better user experience.

-Optionally implement a history feature using SQLite or another method to store past operations.

-Test the application for various edge cases (e.g., non-numeric input, division by zero).

Part 1: Calculator Console Application with Logging

Code :

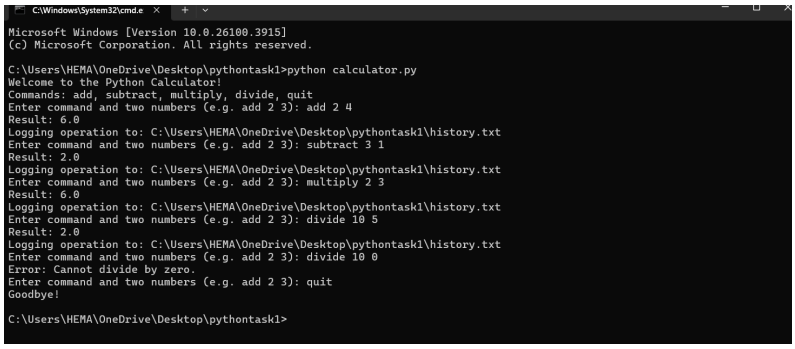
```
import os


def add(x, y):
    return x + y


def subtract(x, y):
    return x - y


def multiply(x, y):
    return x * y


def divide(x, y):
    if y == 0:
        raise ValueError("Cannot divide by zero.")
```

```
    return x / y


def log_operation(operation, num1, num2, result):
    filepath = os.path.abspath("history.txt")
    print(f"Logging operation to: {filepath}")
    with open("history.txt", "a") as f:
        f.write(f"{operation} {num1} {num2} = {result}\n")


def main():
    print("Welcome to the Python Calculator!")
    print("Commands: add, subtract, multiply, divide, quit")

    while True:
        user_input = input("Enter command and two numbers (e.g. add 2 3): ")
        if user_input.lower() == 'quit':
            print("Goodbye!")
            break

        parts = user_input.split()
        if len(parts) != 3:
            print("Invalid input format. Use: command number1 number2")
            continue

        command, num1_str, num2_str = parts
```

```
try:
    num1 = float(num1_str)
    num2 = float(num2_str)
except ValueError:
    print("Please enter valid numbers.")
    continue

try:
    if command == "add":
        result = add(num1, num2)
    elif command == "subtract":
        result = subtract(num1, num2)
    elif command == "multiply":
        result = multiply(num1, num2)
    elif command == "divide":
        result = divide(num1, num2)
    else:
        print("Unknown command.")
        continue

    print(f"Result: {result}")
    log_operation(command, num1, num2, result)

except ValueError as e:
    print(f"Error: {e}")
```

```
if __name__ == "__main__":
    main()
```

Execution & output:--



Part 2: Optional GUI with Tkinter :

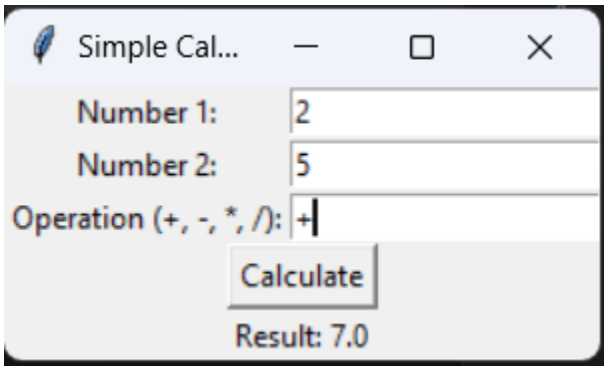
Code : -

```
import tkinter as tk
from tkinter import messagebox

def calculate():
    try:
        num1 = float(entry_num1.get())
        num2 = float(entry_num2.get())
        op = operation.get()

        if op == '+':
            result = num1 + num2
        elif op == '-':
```

```
            result = num1 - num2
        elif op == '*':
            result = num1 * num2
        elif op == '/':
            if num2 == 0:
                raise ValueError("Cannot divide by zero")
            result = num1 / num2
        else:
            messagebox.showerror("Error", "Invalid operation")
            return

        label_result.config(text=f"Result: {result}")
    except ValueError as e:
        messagebox.showerror("Error", f"Invalid input: {e}")

root = tk.Tk()
root.title("Simple Calculator")

tk.Label(root, text="Number 1:").grid(row=0, column=0)
entry_num1 = tk.Entry(root)
entry_num1.grid(row=0, column=1)

tk.Label(root, text="Number 2:").grid(row=1, column=0)
entry_num2 = tk.Entry(root)
entry_num2.grid(row=1, column=1)
```

```
tk.Label(root, text="Operation (+, -, *, /):").grid(row=2, column=0)

operation = tk.StringVar()

entry_op = tk.Entry(root, textvariable=operation)

entry_op.grid(row=2, column=1)


btn_calc = tk.Button(root, text="Calculate", command=calculate)

btn_calc.grid(row=3, column=0, columnspan=2)


label_result = tk.Label(root, text="Result:")

label_result.grid(row=4, column=0, columnspan=2)


root.mainloop()
```

Execution and output :-



Part 3: Text-Based Adventure Game

Code : -

```
def start_game():

    print("Welcome to the Adventure Game!")
```

```
inventory = []


def first_room():
    print("\nYou are in a dark room. There is a door to your left and right.")
    choice = input("Which door do you take? (left/right) ").lower()
    if choice == 'left':
        left_room()
    elif choice == 'right':
        right_room()
    else:
        print("Invalid choice.")
        first_room()


def left_room():
    print("\nYou found a key!")
    inventory.append('key')
    print(f"Inventory: {inventory}")
    first_room()


def right_room():
    print("\nYou meet a dragon!")
    if 'key' in inventory:
        print("You use the key to open a treasure chest and win!")
    else:
        print("You have no weapon and the dragon defeats you. Game Over.")
```

```
        quit()


    first_room()


if __name__ == "__main__":
    start_game()
```

Execution & output :-

```
You are in a dark room. There is a door to your left and right.
Which door do you take? (left/right) left

You found a key!
Inventory: ['key']

You are in a dark room. There is a door to your left and right.
Which door do you take? (left/right) right

You meet a dragon!
You use the key to open a treasure chest and win!
PS C:\Users\HEMA\OneDrive\Desktop\pythontask1>
```