AI Assisted Coding — Factorial Assignment

Question 1 — Factorial without using user defined function

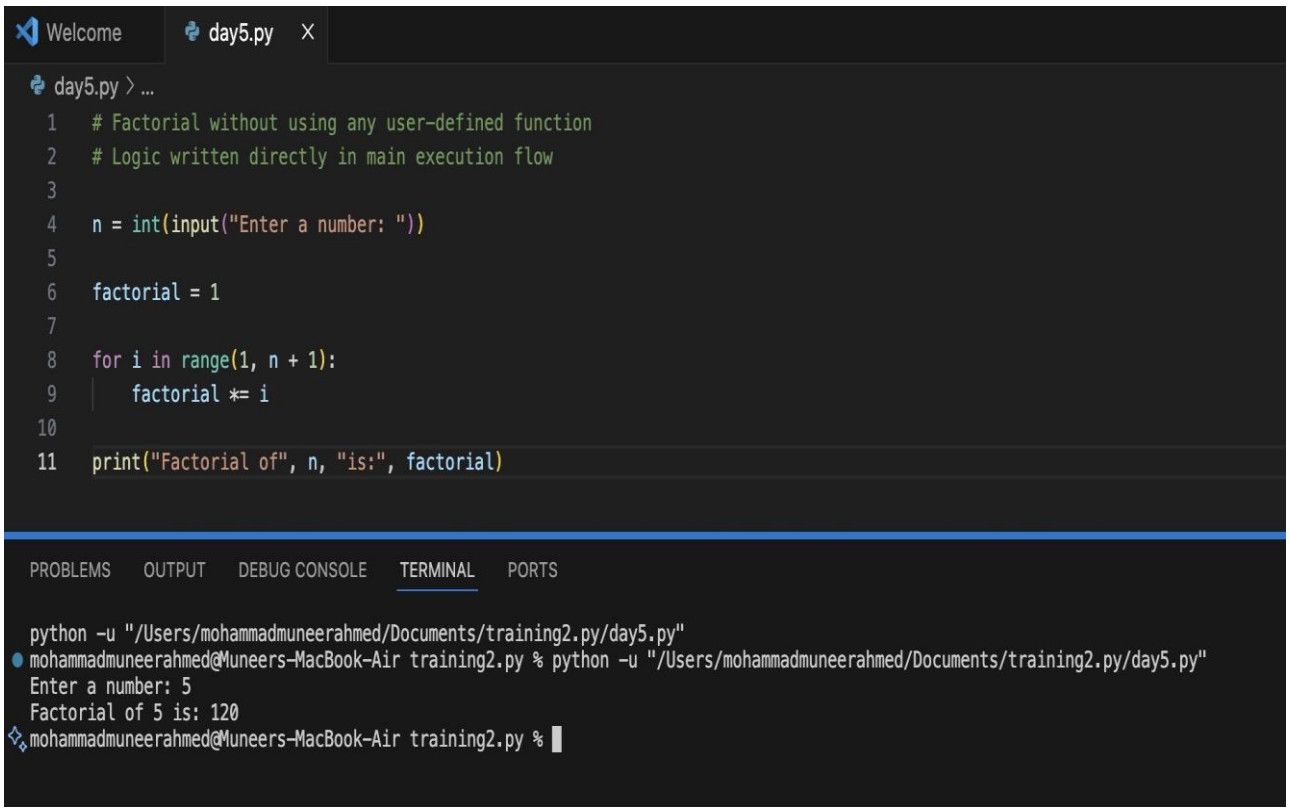
Algorithm

1 Input number 2 Initialize factorial 3 Loop from 1 to n 4 Multiply each step 5 Print result

Pseudo Code

```
START READ n
fact = 1 FOR i =
1 TO n  fact =
fact * i PRINT
fact
END
```

Execution Screenshot



Question 2 — Optimized factorial without functions

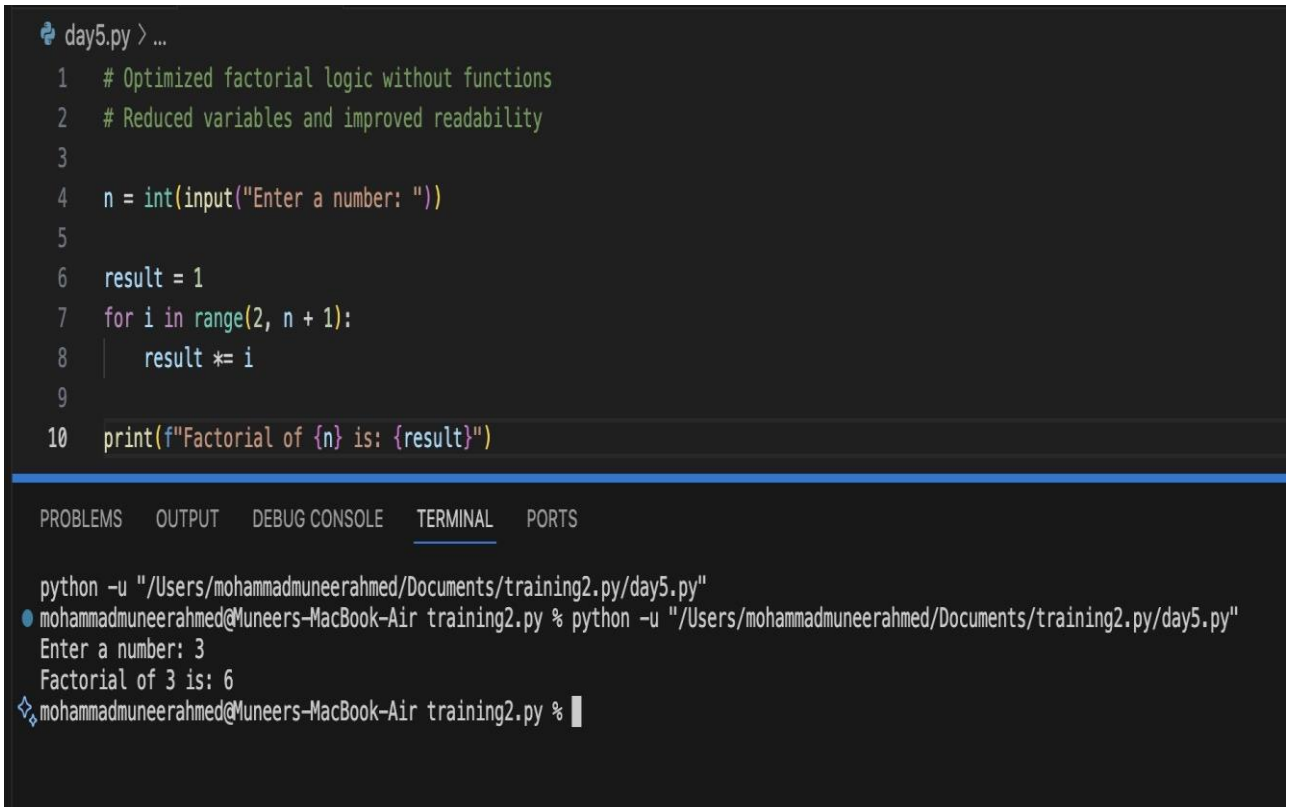
Algorithm

1 Input n 2 Initialize result 3 Loop from 2 to n 4 Multiply 5 Print result

Pseudo Code

```
START READ n result
= 1 FOR i = 2 TO n
result = result * i
PRINT result
END
```

Execution Screenshot



Question 3 — Factorial using function

Algorithm

1 Define factorial function 2 Initialize result 3 Loop multiply 4 Return value 5 Call and print

Pseudo Code

```
FUNCTION factorial(n)
result = 1  FOR i = 1
TO n   result =
result * i  RETURN
result
```

Execution Screenshot

Question 4 — Iterative factorial function

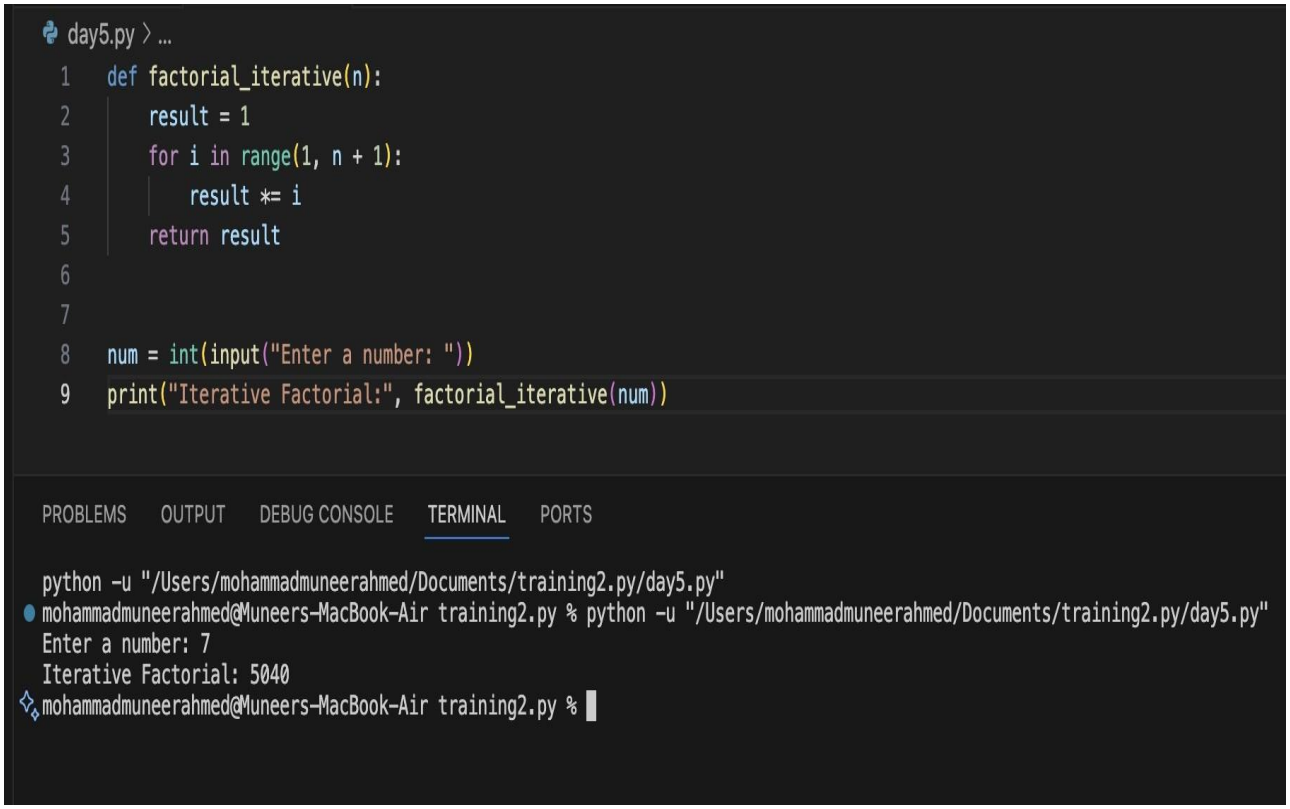
Algorithm

1 Define iterative function 2 Loop multiply 3 Return result

Pseudo Code

```
FUNCTION iterative(n)
result = 1  FOR i = 1
TO n   result =
result * i  RETURN
result
```

Execution Screenshot



Question 5 — Procedural vs Modular comparison

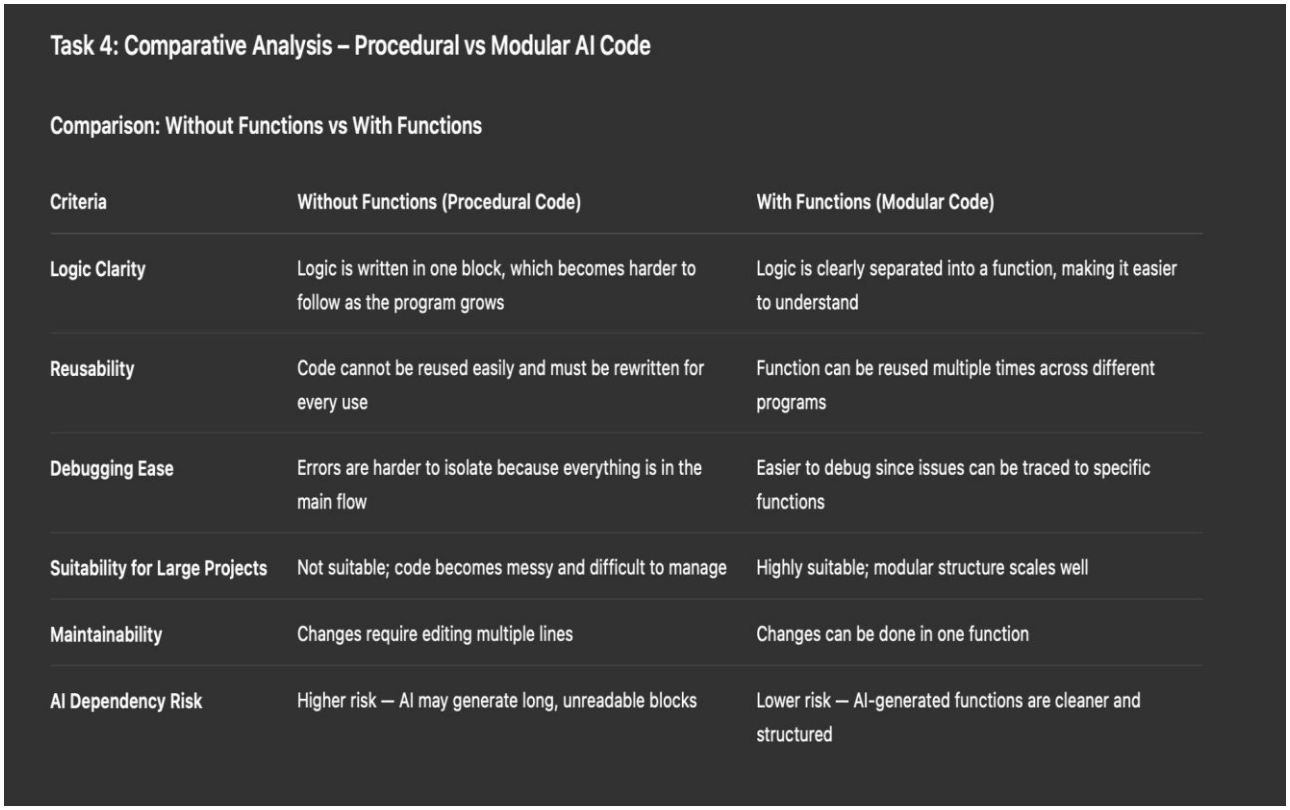
Algorithm

1 Compare clarity 2 Compare reusability 3 Compare debugging 4 Compare scalability

Pseudo Code

```
START
COMPARE procedural vs modular
WRITE differences
END
```

Execution Screenshot



Task 4: Comparative Analysis – Procedural vs Modular AI Code

Comparison: Without Functions vs With Functions

Criteria	Without Functions (Procedural Code)	With Functions (Modular Code)
Logic Clarity	Logic is written in one block, which becomes harder to follow as the program grows	Logic is clearly separated into a function, making it easier to understand
Reusability	Code cannot be reused easily and must be rewritten for every use	Function can be reused multiple times across different programs
Debugging Ease	Errors are harder to isolate because everything is in the main flow	Easier to debug since issues can be traced to specific functions
Suitability for Large Projects	Not suitable; code becomes messy and difficult to manage	Highly suitable; modular structure scales well
Maintainability	Changes require editing multiple lines	Changes can be done in one function
AI Dependency Risk	Higher risk — AI may generate long, unreadable blocks	Lower risk — AI-generated functions are cleaner and structured