

School of Computer Science and Artificial Intelligence

---

**Lab Assignment # 7.2**

---

Program : B. Tech (CSE)  
Specialization  
Course Title : AI Assisted Coding  
Course Code : 23CS002PC304  
Semester 11  
Academic Session : 2025-2026  
Name of Student : P. ESHWAR  
Enrollment No. : 2403A51L26  
Batch No. 51  
Date : 30/01/26

---

**Submission Starts here**

Screenshots:

**Task 1 — Runtime Error Due to Invalid Input Type**

The issue is fixed by converting the input to a numeric type using float0 and handling invalid input with try-except

**(Buggy Code):**

```
num      =  
input("Enter a  
number: ")  
  
result = num +  
10  
  
print(result)
```

The screenshot shows a code editor interface with a dark theme. At the top, there's a status bar with tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (which is currently selected), and PORTS. Below the status bar is a code editor window containing a Python script named 'ai\_coding.py'. The code is as follows:

```
❶ ai_coding.py > ...
1 #Task 1 - Runtime Error Due to Invalid Input Type
2 #* A Python program accepts user input and performs arithmetic operations. However, the program throws a runtime error because the input is treated
3 #Example (Buggy Code):
4 #result = num + 10
5 #print(result)
6 #* Task:
7 #Use AI tools to identify the cause of the runtime error and modify the program so it executes correctly.
8 #Expected Output -1:
9 #* AI converts the input to the appropriate numeric type and eliminates the runtime error.
10 # Corrected Code
11 num = input("Enter a number: ")
12 num = float(num) # Convert input to a numeric type (float)
13 result = num + 10
14 print(result)
15 # Expected Output -2:
16 #* When the user inputs a number, the program correctly adds 10 and displays the result without any errors.
17 # Example Input: 5
18 # Example Output: 15.0
19 # Example Input: 5.5
20 # Example Output: 15.5
21
22
```

Below the code editor is a terminal window showing the execution of the script. The terminal output is:

```
PS C:\Users\Eshwar\OneDrive\Desktop> & "C:\Program Files\Python314\python.exe" c:/Users/Eshwar/OneDrive/Desktop/python/ai_coding.py
Enter a number: 5
15.0
PS C:\Users\Eshwar\OneDrive\Desktop>
```

## Task 2 - Incorrect Function Return Value

A function is designed to calculate the square of a number, but it does not return the computed result properly.

### Example (Buggy Code):

```
def square(n):
    result = n * n
```

### Task:

Use AI assistance to analyze the function and ensure the correct value is returned.

```
ai_coding.py > ...
1 #Incorrect Function Return Value
2 #A function is designed to calculate the square of a number, but it does not return the computed result properly.
3 #Example (Buggy Code):
4 # def square(n):
5 #     result = n * n
6 #Task:
7 # Use AI assistance to analyze the function and ensure the correct value is returned.
8 def square(n):
9     result = n * n
10    return result # Ensure the computed result is returned properly
11 # Example usage:
12 print(square(4)) # Output should be 16
13 print(square(5)) # Output should be 25
14 print(square(6)) # Output should be 36
15 # Corrected Code:
16 def square(n):
17     result = n * n
18     return result
19 # Ensure the computed result is returned properly
20 # Example usage:
21 print(square(4)) # Output should be 16
22 print(square(5)) # Output should be 25
23 print(square(6)) # Output should be 36
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Eshwar\OneDrive\Desktop\python> & "C:\Program Files\Python314\python.exe" c:/Users/Eshwar/OneDrive/Desktop/python/ai_coding.py
16
25
36
16
25
36
PS C:\Users\Eshwar\OneDrive\Desktop\python>
```

## Task 3 – IndexError in List Traversal

A Python program iterates over a list using incorrect index limits, causing an IndexError.

### Example (Buggy Code):

```
numbers = [10, 20, 30]
for i in range(0, len(numbers)+1):
    print(numbers[i])
```

### Task:

Use AI to identify the incorrect loop boundary and correct the iteration logic.

```
ai_coding.py > ...
1 #IndexError in List Traversal
2 #A Python program iterates over a list using incorrect index limits, causing an IndexError.
3 #Example (Buggy Code):
4 # numbers = [10, 20, 30]
5 # for i in range(0, len(numbers)+1):
6 #     print(numbers[i])
7 # Task:
8 # Use AI to identify the incorrect loop boundary and correct the iteration logic.
9 numbers = [10, 20, 30]
10 for i in range(0, len(numbers)):
11     print(numbers[i])
12 # Corrected Code:
13 # The loop now correctly iterates from 0 to len(numbers)-1, preventing IndexError.
14
15
16
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Eshwar\OneDrive\Desktop\python> & "C:\Program Files\Python314\python.exe" c:/Users/Eshwar/OneDrive/Desktop/python/ai_coding.py
10
20
30
PS C:\Users\Eshwar\OneDrive\Desktop\python>
```

## Task 4 – Uninitialized Variable Usage

A program uses a variable in a calculation before assigning it any value.

**Example (Buggy Code):**

if True:

pass

print(total)

**Task:**

Use AI tools to detect the uninitialized variable and correct the program.

The screenshot shows a code editor with a dark theme. In the top-left corner, there's a file icon followed by "ai\_coding.py > ...". Below it is a multi-line text area containing Python code. The code is a script named "ai\_coding.py" with comments explaining the task and a buggy section where a variable is used before being assigned. A corrected version is shown below, with line 11 highlighted in yellow. Lines 12 and 13 are also highlighted in yellow. Line 14 is a blank line.

```
ai_coding.py > ...
1 #Task 4 - Uninitialized Variable Usage
2 #A program uses a variable in a calculation before assigning it any value.
3 #Example (Buggy Code):
4     #if True:
5     #    pass
6     #    print(total)
7 #Task:
8 #Use AI tools to detect the uninitialized variable and correct the program.
9 #Corrected Code:
10 if True:
11     total = 0 # Initialize the variable before using it
12 print(total)
13
14
```

Below the code editor, there's a terminal window with tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL (which is selected), and PORTS. The terminal shows the command "python ai\_coding.py" being run, with the output "0" displayed. The command prompt is "PS C:\Users\Eshwar\OneDrive\Desktop\python>".

## Task 5 – Logical Error in Student Grading System

A grading program assigns incorrect grades due to improper conditional logic.

### Example (Buggy Code):

```
marks = 85
if marks >= 90:
    grade = "A"
elif marks >= 80:
    grade = "C"
else:
    grade = "B"
print(grade)
```

### Task:

Use AI to analyze the grading conditions and correct the logical flow.

The screenshot shows a code editor interface with a dark theme. On the left is the code editor pane containing Python code for a grading system. On the right is the terminal pane showing the execution of the code.

**Code Editor Content:**

```
ai_coding.py > ...
1 #Task 5 - Logical Error in Student Grading System
2 #A grading program assigns incorrect grades due to improper conditional logic.
3 #Example (Buggy code):
4 #marks = 85
5 #if marks >= 90:
6 | # grade = "A"
7 elif marks >= 80:
8 | # grade = "C"
9 else:
10 | # grade = "B"
11 print(grade)
12 #Task:
13 #Use AI to analyze the grading conditions and correct the logical flow.
14 marks = 85
15 if marks >= 90:
16 | grade = "A"
17 elif marks >= 80:
18 | grade = "B"
19 else:
20 | grade = "C"
21 print(grade)
22
23
```

**Terminal Content:**

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS powershell
PS C:\Users\Eshwar\OneDrive\Desktop\python> & "C:\Program Files\Python314\python.exe" c:/Users/Eshwar/OneDrive/Desktop/python/ai_coding.py
B
PS C:\Users\Eshwar\OneDrive\Desktop\python>
```