

AI ASSISTANT CODING

ASSIGNMENT-3.2

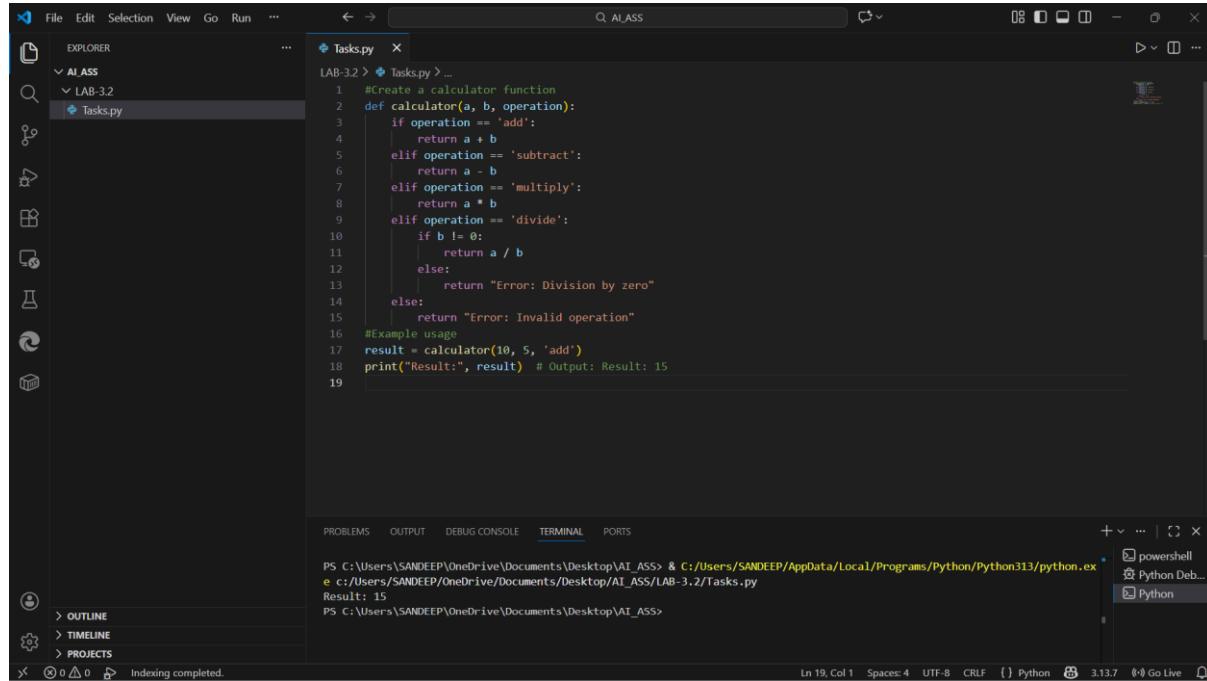
Task Description-1

- Progressive Prompting for Calculator Design: Ask the AI to design a simple calculator program by initially providing only the function name. Gradually enhance the prompt by adding comments and usage examples.

Expected Output-1

- Comparison showing improvement in AI-generated calculator logic and structure.

CODE:



```
File Edit Selection View Go Run ... < > AI_ASS ... D ...  
EXPLORER ... Tasks.py x  
AI_ASS  
LAB-3.2  
Tasks.py  
  
LAB-3.2 > Tasks.py > ...  
1 #Create a calculator function  
2 def calculator(a, b, operation):  
3     if operation == 'add':  
4         return a + b  
5     elif operation == 'subtract':  
6         return a - b  
7     elif operation == 'multiply':  
8         return a * b  
9     elif operation == 'divide':  
10        if b != 0:  
11            return a / b  
12        else:  
13            return "Error: Division by zero"  
14    else:  
15        return "Error: Invalid operation"  
16 #Example usage  
17 result = calculator(10, 5, 'add')  
18 print("Result:", result) # Output: Result: 15  
19  
  
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS  
PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS> & C:/Users/SANDEEP/AppData/Local/Programs/Python/Python313/python.exe e c:/Users/SANDEEP/OneDrive/Documents/Desktop/AI_ASS/LAB-3.2/Tasks.py  
Result: 15  
PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS>  
Indexing completed.  
0 ▲ 0 ⌂ Indexing completed.  
Indexing completed.  
+ ... | x  
powershell  
Python Deb...  
Python  
Ln 19, Col 1 Spaces: 4 UTF-8 CRLF () Python 3.13.7 ⓘ Go Live
```

A screenshot of the Visual Studio Code (VS Code) interface. The left sidebar shows a project structure with 'AI ASS' expanded, containing 'LAB-3.2' and 'Tasks.py'. The main editor window displays a Python script named 'Tasks.py' with the following code:

```
20
21 """LEVEL_2"""
22
23
24 #Create a calculator function that supports addition, subtraction, multiplication, and division.
25 # Add basic error handling.
26 def calculator(a, b, operation):
27     if operation == "add":
28         return a + b
29     elif operation == "subtract":
30         return a - b
31     elif operation == "multiply":
32         return a * b
33     elif operation == "divide":
34         if b != 0:
35             return a / b
36         else:
37             return "Error: Division by zero"
38     else:
39         return "Error: Invalid operation"
```

The terminal at the bottom shows the command 'python Tasks.py' and its output: 'Result: 15'. The status bar indicates the file is 3.13.7 and the code is in Python.

A screenshot of the Visual Studio Code (VS Code) interface. The left sidebar shows a project structure with 'AI ASS' expanded, containing 'LAB-3.2' and 'Tasks.py'. The main editor window displays a Python script named 'Tasks.py' with the following code:

```
45
46 #Create a calculator function with docstring and example usage.
47 def calculator(a, b, operation):
48     """
49     A simple calculator function that performs basic arithmetic operations.
50     Parameters:
51     a (float): The first number.
52     b (float): The second number.
53     operation (str): The operation to perform. Can be 'add', 'subtract', 'multiply', or 'divide'.
54     Returns:
55     float: The result of the arithmetic operation.
56     str: Error message if division by zero or invalid operation is attempted.
57     Example usage:
58     >>> calculator(10, 5, 'add')
59     15
60     >>> calculator(10, 5, 'subtract')
61     5
62     >>> calculator(10, 5, 'multiply')
63     50
64     >>> calculator(10, 5, 'divide')
65     2.0
66     >>> calculator(10, 0, 'divide')
67     Error: Division by zero.
68     >>> calculator(10, 5, 'modulus')
69     Error: Invalid operation
70     """
71     if operation == "add":
72         return a + b
73     elif operation == "subtract":
74         return a - b
75     elif operation == "multiply":
76         return a * b
77     elif operation == "divide":
78         if b != 0:
79             return a / b
80         else:
81             return "Error: Division by zero"
82     else:
83         return "Error: Invalid operation"
```

The terminal at the bottom shows the command 'python Tasks.py' and its output: 'Result: 15'. The status bar indicates the file is 3.13.7 and the code is in Python.

Task Description-2

- Refining Prompts for Sorting Logic: Start with a vague prompt for sorting student marks, then refine it to clearly specify sorting order and constraints.

Expected Output-2

- AI-generated sorting function evolves from ambiguous logic to an accurate and efficient implementation.

```

File Edit Selection View Go Run ...
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
25.18.0-win32-x64\bundled\libs\debugpy\launcher` '53631' ... "c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS\LAB-3.2\Tasks.py"
● PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS> c;; cd "c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS"; & "c:\Users\SANDEEP\AppData\Local\Programs\Python\Python313\python.exe" "c:\Users\SANDEEP\.vscode\extensions\ms-python.python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher` '53653' ... "c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS\LAB-3.2\Tasks.py"
● PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS> c;; cd "c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS"; & "c:\Users\SANDEEP\AppData\Local\Programs\Python\Python313\python.exe" "c:\Users\SANDEEP\.vscode\extensions\ms-python.python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher` '50831' ... "c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS\LAB-3.2\Tasks.py"
Sorted student marks: [79, 85, 88, 92, 95]
○ PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS>

```

```

File Edit Selection View Go Run ...
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
25.18.0-win32-x64\bundled\libs\debugpy\launcher` '50831' ... "c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS\LAB-3.2\Tasks.py"
● PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS> c;; cd "c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS"; & "c:\Users\SANDEEP\AppData\Local\Programs\Python\Python313\python.exe" "c:\Users\SANDEEP\.vscode\extensions\ms-python.python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher` '62315' ... "c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS\LAB-3.2\Tasks.py"
● PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS> c;; cd "c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS"; & "c:\Users\SANDEEP\AppData\Local\Programs\Python\Python313\python.exe" "c:\Users\SANDEEP\.vscode\extensions\ms-python.python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher` '60781' ... "c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS\LAB-3.2\Tasks.py"
Sorted student marks in descending order: [92, 90, 88, 85, 78]
○ PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS>

```

Task Description-3

- Few-Shot Prompting for Prime Number Validation: Provide multiple input-output examples for a function that checks whether a number is prime. Observe how few-shot prompting improves correctness.

Expected Output-3

Improved prime-checking function with better edge-case handling.

```

1  #Check if a number is prime.
2  # Input: 2 -> Output: Prime
3  # Input: 4 -> Output: Not Prime
4  # Input: 7 -> Output: Prime
5  # Input: 1 -> Output: Not Prime
6
7  def is_prime(num):
8      """
9          Checks if a number is prime.
10         Parameters:
11             num (int): The number to check.
12         Returns:
13             str: 'Prime' if the number is prime, 'Not Prime' otherwise.
14         Example usage:
15         >>> is_prime(2)
16         'Prime'
17         >>> is_prime(4)
18         'Not Prime'
19         >>> is_prime(7)
20         'Prime'
21         >>> is_prime(1)
22         'Not Prime'
23     """
24     if num <= 1:
25         return "Not Prime"
26     for i in range(2, int(num**0.5) + 1):
27         if num % i == 0:
28             return "Not Prime"
29     return "Prime"
30
31 print(is_prime(2)) # Output: Prime
32 print(is_prime(4)) # Output: Not Prime
33 print(is_prime(7)) # Output: Prime
34 print(is_prime(1)) # Output: Not Prime

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS> c: cd 'c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS'; & 'c:\Users\SANDEEP\AppData\Local\Programs\Python\Python311\python.exe' Tasks.py

Prime
Not Prime
Prime
Not Prime

PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS>

Task Description-4

- Prompt-Guided UI Design for Student Grading System: Create a user interface for a student grading system that calculates total marks, percentage, and grade based on user input.

Expected Output-4

- Well-structured UI code with accurate calculations and clear output display.

```

1  #Create a Python UI program that accepts student marks,
2  #calculates total, percentage, and assigns grade.
3  def student_grading():
4      marks = []
5      for i in range(5):
6          mark = float(input("Enter marks for subject (i+1): "))
7          marks.append(mark)
8      total = sum(marks)
9      percentage = (total / 500) * 100
10     if percentage >= 90:
11         grade = 'A'
12     elif percentage >= 80:
13         grade = 'B'
14     elif percentage >= 70:
15         grade = 'C'
16     elif percentage >= 60:
17         grade = 'D'
18     else:
19         grade = 'F'
20     print("Total Marks: {total}")
21     print("Percentage: {percentage:.2f}%")
22     print("Grade: {grade}")
23 student_grading()

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS> c: cd 'c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS'; & 'c:\Users\SANDEEP\AppData\Local\Programs\Python\Python311\python.exe' Tasks.py

Enter marks for subject 1: 90
Enter marks for subject 2: 80
Enter marks for subject 3: 70
Enter marks for subject 4: 60
Enter marks for subject 5: 50
Total Marks: 350.0
Percentage: 70.00%
Grade: C

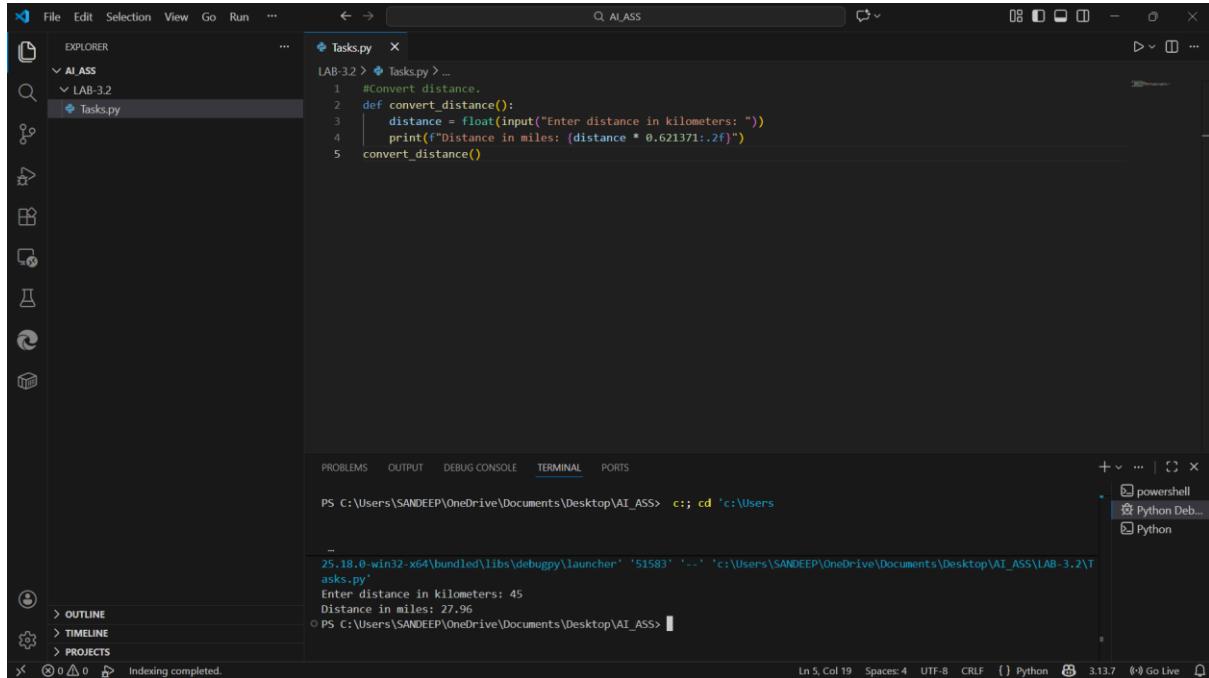
PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS>

Task Description-5

- Analyzing Prompt Specificity in Unit Conversion Functions: Improving a Unit Conversion Function (Kilometers to Miles and Miles to Kilometers) Using Clear Instructions.

Expected Output-5

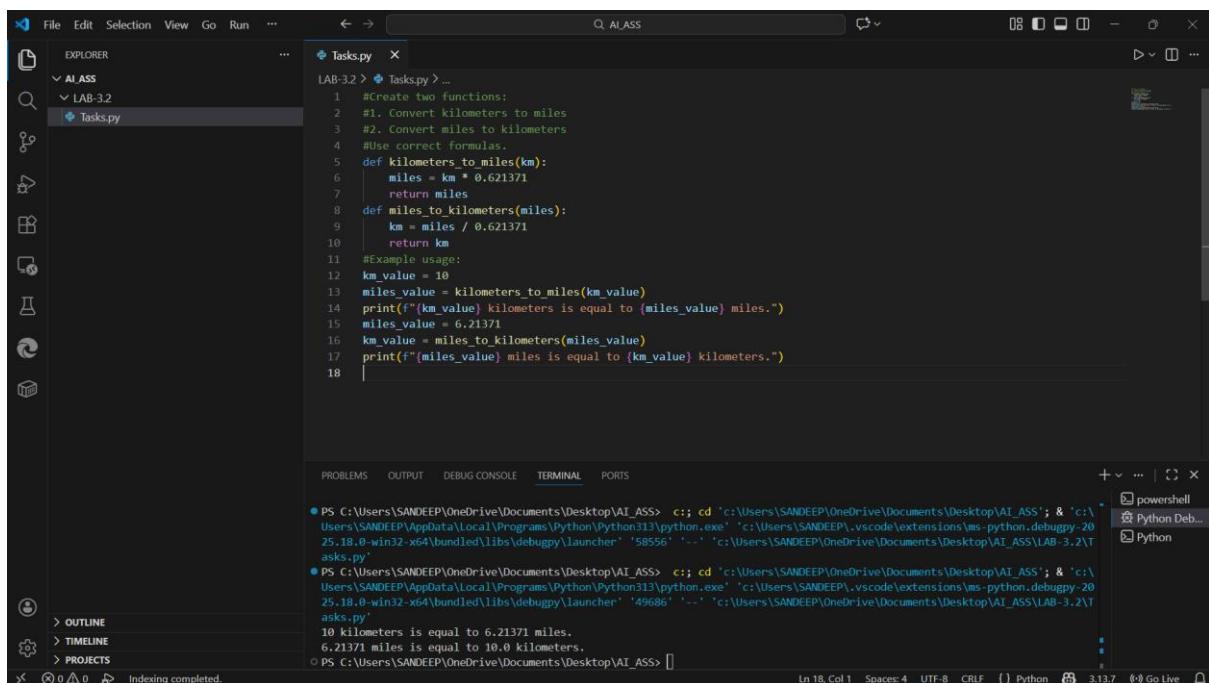
- Analysis of code quality and accuracy differences across multiple prompt variations.



```
File Edit Selection View Go Run ... < > Q AI_ASS
EXPLORER
  AI_ASS
    LAB-3.2
      Tasks.py
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS> c:; cd 'c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS\LAB-3.2\asks.py'
25.18.0-win32-x64\bundled\libs\debugpy\launcher' '51583' '--' 'c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS\LAB-3.2\asks.py'
Enter distance in kilometers: 45
Distance in miles: 27.96
PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS>

```

Kilometers to Miles and Miles to Kilometers:



```
File Edit Selection View Go Run ... < > Q AI_ASS
EXPLORER
  AI_ASS
    LAB-3.2
      Tasks.py
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS> c:; cd 'c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS'; & 'c:\Users\SANDEEP\AppData\Local\Programs\Python\Python311\python.exe' 'c:\Users\SANDEEP\vscode\extensions\ms-python.debugpy-2023.1.10000\python\debugpy\launcher' '58556' '--' 'c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS\LAB-3.2\asks.py'
PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS> c:; cd 'c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS'; & 'c:\Users\SANDEEP\AppData\Local\Programs\Python\Python311\python.exe' 'c:\Users\SANDEEP\vscode\extensions\ms-python.debugpy-2023.1.10000\python\debugpy\launcher' '49686' '--' 'c:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS\LAB-3.2\asks.py'
10 kilometers is equal to 6.21371 miles.
6.21371 miles is equal to 10.0 kilometers.
PS C:\Users\SANDEEP\OneDrive\Documents\Desktop\AI_ASS>

```