

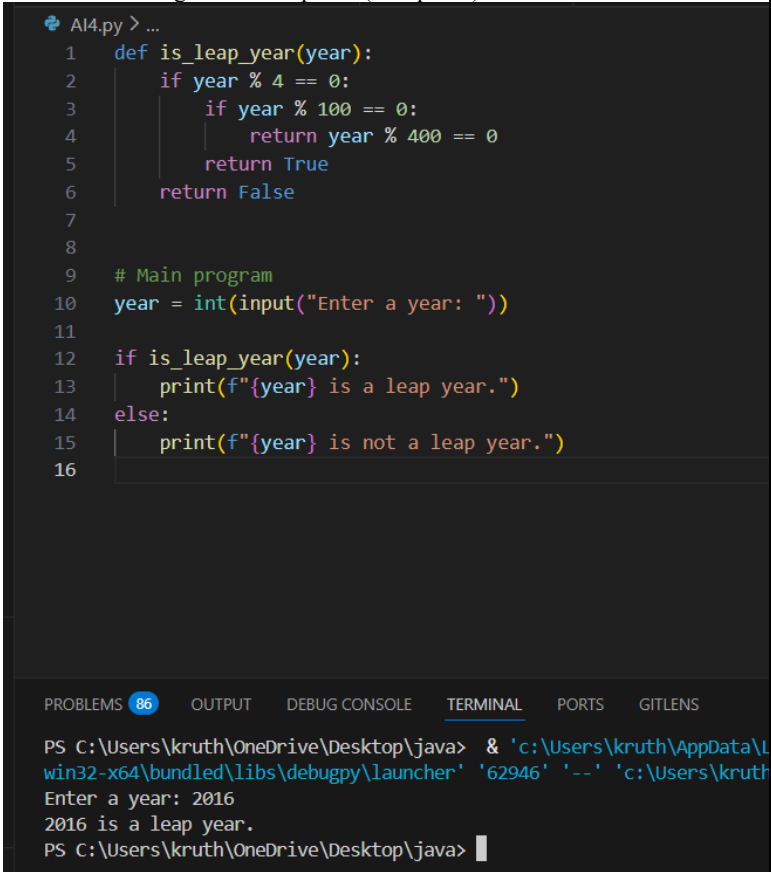
NAME:CH.Kruthankiran

H.NO:2303A51404

BATCH:26

SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE		DEPARTMENT OF COMPUTER SCIENCE ENGINEERING	
Program Name: B. Tech		Assignment Type: Lab	Academic Year:2025-2026
Course Coordinator Name		Dr. Rishabh Mittal	
Instructor(s) Name		Mr. S Naresh Kumar	
		Ms. B. Swathi	
		Dr. Sasanko Shekhar Gantayat	
		Mr. Md Sallauddin	
		Dr. Mathivanan	
		Mr. Y Srikanth	
		Ms. N Shilpa	
		Dr. Rishabh Mittal (Coordinator)	
		Dr. R. Prashant Kumar	
		Mr. Ankushavali MD	
		Mr. B Viswanath	
		Ms. Sujitha Reddy	
		Ms. A. Anitha	
		Ms. M.Madhuri	
		Ms. Katherashala Swetha	
		Ms. Velpula sumalatha	
		Mr. Bingi Raju	
Course Code	23CS002PC304	Course Title	AI Assisted Coding
Year/Sem	III/I	Regulation	R23
Date and Day of Assignment	Week 2 - Wednesday	Time(s)	23CSBTB01 To 23CSBTB52
Duration	2 Hours	Applicable to Batches	All batches
Assignment Number: 3.3(Present assignment number)/24(Total number of assignments)			

Q.No.	Question	Expected Time to complete
1	Lab 4: Advanced Prompt Engineering – Zero-shot, One-shot, and Few-shot Techniques Lab Objectives <ul style="list-style-type: none">To explore and apply different levels of prompt examples in AI-assisted code generationTo understand how zero-shot, one-shot, and few-shot prompting affect AI	Week2 - Wednesday

	<p>output quality</p> <ul style="list-style-type: none"> To evaluate the impact of context richness and example quantity on AI performance To build awareness of prompt strategy effectiveness for different problem types <p>Lab Outcomes (LOs) After completing this lab, students will be able to:</p> <ul style="list-style-type: none"> Use zero-shot prompting to instruct AI with minimal context Use one-shot prompting with a single example to guide AI code generation Apply few-shot prompting using multiple examples to improve AI responses Compare AI outputs across different prompting strategies 	
	<p>Task 1: Zero-Shot Prompting – Leap Year Check Scenario Zero-shot prompting involves giving instructions without providing examples. Task Description Use zero-shot prompting to instruct an AI tool to generate a Python function that:</p> <ul style="list-style-type: none"> Accepts a year as input Checks whether the given year is a leap year Returns an appropriate result <p>Note: No input-output examples should be provided in the prompt. Expected Output</p> <ul style="list-style-type: none"> AI-generated leap year checking function Correct logical conditions Sample input and output Screenshot of AI-generated response (if required)  <pre> AI4.py > ... 1 def is_leap_year(year): 2 if year % 4 == 0: 3 if year % 100 == 0: 4 return year % 400 == 0 5 return True 6 return False 7 8 9 # Main program 10 year = int(input("Enter a year: ")) 11 12 if is_leap_year(year): 13 print(f"{year} is a leap year.") 14 else: 15 print(f"{year} is not a leap year.") 16 PROBLEMS 86 OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS PS C:\Users\kruth\OneDrive\Desktop\java> & 'c:\Users\kruth\AppData\Local\Microsoft\Windows\apps\winget-cli.exe' --source 'winget' --id 'Python.Python.3.11' --name 'Python 3.11' --install Enter a year: 2016 2016 is a leap year. PS C:\Users\kruth\OneDrive\Desktop\java> </pre>	

Task 2: One-Shot Prompting – Centimeters to Inches Conversion

Scenario

One-shot prompting guides AI using a single example.

Task Description

Use one-shot prompting by providing one input-output example to generate a Python function that:

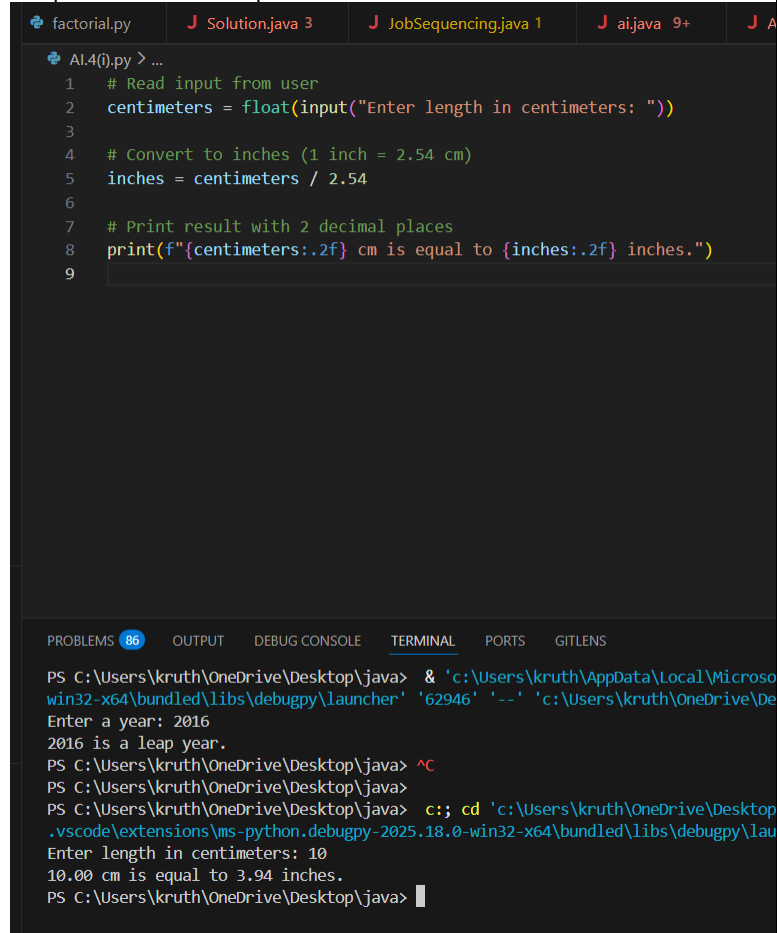
- Converts centimeters to inches
- Uses the correct mathematical formula

Example provided in prompt:

Input: 10 cm → Output: 3.94 inches

Expected Output

- Python function with correct conversion logic
- Accurate calculation
- Sample test cases and outputs



```
factorial.py  Solution.java 3  JobSequencing.java 1  ai.java 9+  J A

AI.4(i).py > ...
1 # Read input from user
2 centimeters = float(input("Enter length in centimeters: "))
3
4 # Convert to inches (1 inch = 2.54 cm)
5 inches = centimeters / 2.54
6
7 # Print result with 2 decimal places
8 print(f"{centimeters:.2f} cm is equal to {inches:.2f} inches.")
9

PROBLEMS 86 OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS

PS C:\Users\kruth\OneDrive\Desktop\java> & 'c:\Users\kruth\AppData\Local\Microsoft\WindowsApps\Microsoft.Windows.Common-UI\10.0.17134.0\bin\win32-x64\bundled\libs\debugpy\launcher' '62946' '--' 'c:\Users\kruth\OneDrive\Desktop\java'
Enter a year: 2016
2016 is a leap year.
PS C:\Users\kruth\OneDrive\Desktop\java> ^C
PS C:\Users\kruth\OneDrive\Desktop\java>
PS C:\Users\kruth\OneDrive\Desktop\java> c:: cd 'c:\Users\kruth\OneDrive\Desktop\java'
.vsc\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher
Enter length in centimeters: 10
10.00 cm is equal to 3.94 inches.
PS C:\Users\kruth\OneDrive\Desktop\java> |
```

Task 3: Few-Shot Prompting – Name Formatting

Scenario

Few-shot prompting improves accuracy by providing multiple examples.

Task Description

Use few-shot prompting with 2–3 examples to generate a Python function that:

- Accepts a full name as input
- Formats it as “Last, First”

Example formats:

- "John Smith" → "Smith, John"

	<ul style="list-style-type: none">• "Anita Rao" → "Rao, Anita" <p>Expected Output</p> <ul style="list-style-type: none">• Well-structured Python function• Output strictly following example patterns• Correct handling of names• Sample inputs and outputs <div><pre>AI4(ii).py > ... 6 7 # Check if at least first and last name are present 8 if len(name_parts) >= 2: 9 first_name = name_parts[0] 10 last_name = name_parts[-1] # last word as last name 11 12 # Print in "Last, First" format 13 print(f'{last_name}, {first_name}') 14 else: 15 print("Please enter a valid full name with at least a first and last name.") 16</pre><p>PROBLEMS 86 OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS</p><pre>PS C:\Users\kruth\OneDrive\Desktop\java> ^C PS C:\Users\kruth\OneDrive\Desktop\java> PS C:\Users\kruth\OneDrive\Desktop\java> c:: cd 'c:\Users\kruth\OneDrive\Desktop\java'; & .\vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '54 Enter length in centimeters: 10 10.00 cm is equal to 3.94 inches. PS C:\Users\kruth\OneDrive\Desktop\java> ^C PS C:\Users\kruth\OneDrive\Desktop\java> PS C:\Users\kruth\OneDrive\Desktop\java> c:: cd 'c:\Users\kruth\OneDrive\Desktop\java'; & .\vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' '51 Enter your full name (First Last): Kruthan Kiran Kiran, Kruthan PS C:\Users\kruth\OneDrive\Desktop\java> </pre><p>hpaid 71 15 Indexing completed. Java: Ready</p></div> <ul style="list-style-type: none">•	
	<p>Task 4: Comparative Analysis – Zero-Shot vs Few-Shot Scenario</p> <p>Different prompt strategies may produce different code quality.</p> <p>Task Description</p> <ul style="list-style-type: none">• Use zero-shot prompting to generate a function that counts vowels in a string• Use few-shot prompting for the same problem• Compare both outputs based on:<ul style="list-style-type: none">○ Accuracy○ Readability○ Logical clarity <p>Expected Output</p> <ul style="list-style-type: none">• Two vowel-counting functions• Comparison table or short reflection paragraph• Conclusion on prompt effectiveness	

	 <pre> AI4(iii).py > ... 1 # Function to count vowels in a string 2 def count_vowels(text): 3 count = 0 4 vowels = "aeiouAEIOU" 5 6 for ch in text: 7 if ch in vowels: 8 count += 1 9 10 return count 11 12 13 # Main program 14 user_input = input("Enter a string: ") 15 16 vowel_count = count_vowels(user_input) 17 18 print("Number of vowels in the given string:", vowel_count) 19 </pre> <p>PROBLEMS 86 OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS</p> <pre> PS C:\Users\kruth\OneDrive\Desktop\java> ^C PS C:\Users\kruth\OneDrive\Desktop\java> PS C:\Users\kruth\OneDrive\Desktop\java> c::; cd 'C:\Users\kruth\OneDrive\Desktop\java'; .vscod\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' Enter your full name (First Last): Kruthan Kiran Kiran, Kruthan PS C:\Users\kruth\OneDrive\Desktop\java> ^C PS C:\Users\kruth\OneDrive\Desktop\java> PS C:\Users\kruth\OneDrive\Desktop\java> c::; cd 'C:\Users\kruth\OneDrive\Desktop\java'; .vscod\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher' Enter a string: once upon a time Number of vowels in the given string: 7 PS C:\Users\kruth\OneDrive\Desktop\java> </pre> <p>hpaid 71 15 Indexing completed. Java: Ready</p>	
	<p>•</p> <p>Task 5: Few-Shot Prompting – File Handling</p> <p>Scenario File processing requires clear logical understanding.</p> <p>Task Description Use few-shot prompting to generate a Python function that:</p> <ul style="list-style-type: none"> • Reads a .txt file • Counts the number of lines in the file • Returns the line count <p>Expected Output</p> <ul style="list-style-type: none"> • Working Python file-processing function • Correct line count • Sample .txt input and output • AI-assisted logic explanation 	

```
ing.java 1  J ai.java 9+  J Ai2.java 9+  J AI3.java 9+  Hpc ass3.py 2  AI4.p

AI4(iv).py > ...
1 def count_lines_in_file(file_path):
2     line_count = 0
3     try:
4         with open(file_path, "r") as file:
5             for _ in file:
6                 line_count += 1
7             return line_count
8     except FileNotFoundError:
9         print("File not found.")
10        return 0
11    except IOError as e:
12        print("An error occurred while reading the file:", e)
13        return 0
14
15
16 # Main execution (similar to Java main method)
17 if __name__ == "__main__":
18     file_path = r"C:\Users\kruth\Downloads" # Replace with your file path
19     line_count = count_lines_in_file(file_path)
20     print("Number of lines in the file:", line_count)
21

PROBLEMS 86  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  GITLENS

PS C:\Users\kruth\OneDrive\Desktop\java> ^C
PS C:\Users\kruth\OneDrive\Desktop\java>
PS C:\Users\kruth\OneDrive\Desktop\java> c;; cd 'c:\Users\kruth\OneDrive\Desktop\java';
.vscod\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher'
An error occurred while reading the file: [Errno 13] Permission denied: 'C:\Users\kruth\OneDrive\Desktop\java'
Number of lines in the file: 0
PS C:\Users\kruth\OneDrive\Desktop\java> ^C
PS C:\Users\kruth\OneDrive\Desktop\java>
PS C:\Users\kruth\OneDrive\Desktop\java> c;; cd 'c:\Users\kruth\OneDrive\Desktop\java';
.vscod\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\debugpy\launcher'
An error occurred while reading the file: [Errno 13] Permission denied: 'C:\Users\kruth\OneDrive\Desktop\java'
Number of lines in the file: 0
PS C:\Users\kruth\OneDrive\Desktop\java> 
```

•
Note: Report should be submitted as a word document for all tasks in a single document with prompts, comments & code explanation, and output and if required, screenshots.