

Assignment Number: 4.3(Present assignment number)/24(Total number of assignments)

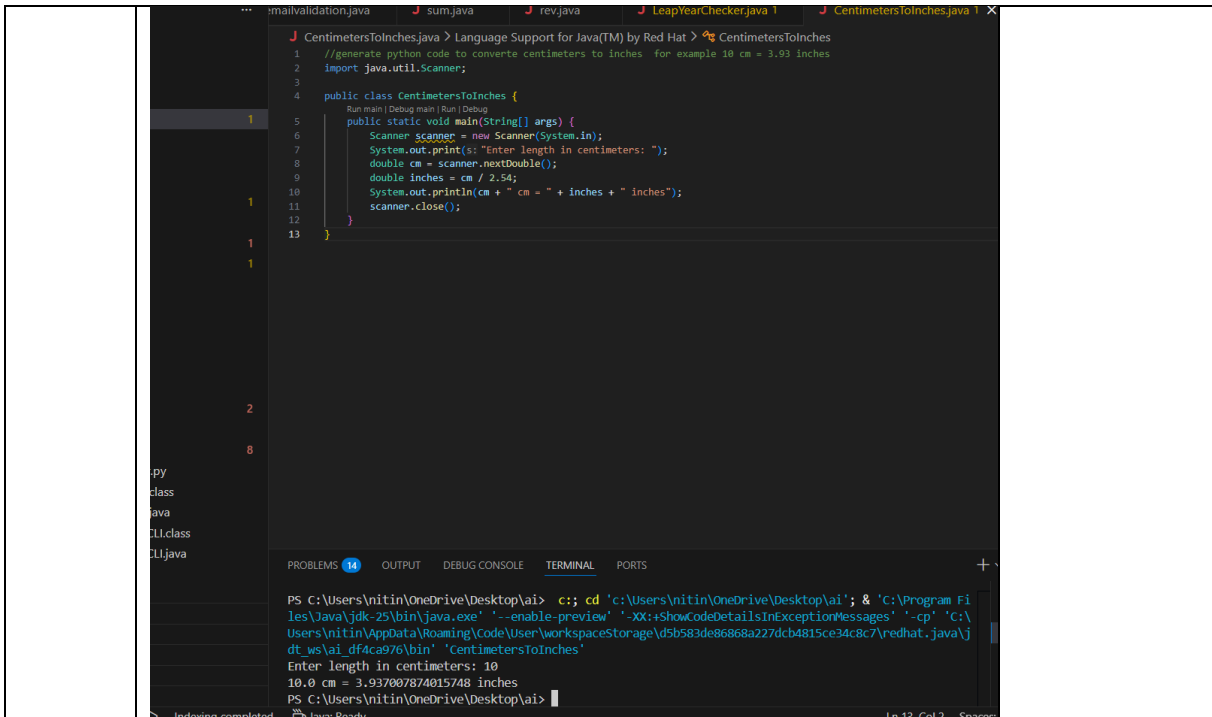
B.NITHIN

2303a51803

B-26

Q.No.	Question	Expected Time to complete
1	<p>Lab 4: Advanced Prompt Engineering – Zero-shot, One-shot, and Few-shot Techniques</p> <p>Lab Objectives</p> <ul style="list-style-type: none">• To explore and apply different levels of prompt examples in AI-assisted code generation• To understand how zero-shot, one-shot, and few-shot prompting affect AI output quality• 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	Week2 - Wednesday
	<p>Task 1: Zero-Shot Prompting – Leap Year Check</p> <p>Scenario Zero-shot prompting involves giving instructions without providing examples.</p> <p>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.</p> <p>Expected Output</p>	

	<div><div>LeapYearChecker.java > Language Support for Java(TM) by Red Hat > LeapYearChecker</div><div><pre>1 //generate the java code for check the year is leap year or not and take the input from user 2 import java.util.Scanner; 3 4 public class LeapYearChecker { 5 public static void main(String[] args) { 6 Scanner scanner = new Scanner(System.in); 7 System.out.print(s: "Enter a year: "); 8 int year = scanner.nextInt(); 9 10 if (isLeapYear(year)) { 11 System.out.println(year + " is a leap year."); 12 } else { 13 System.out.println(year + " is not a leap year."); 14 } 15 16 scanner.close(); 17 } 18 19 public static boolean isLeapYear(int year) { 20 return (year % 4 == 0 && year % 100 != 0) (year % 400 == 0); 21 } 22 }</pre></div><div><div>PROBLEMS 13 OUTPUT DEBUG CONSOLE TERMINAL PORTS</div><div>PS C:\Users\nitin\OneDrive\Desktop\ai> & 'C:\Program Files\Java\jdk-25\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\nitin\AppData\Roaming\Code\User\workspaceStorage\d5b583de86868a227dcb4815ce34c8c7\redhat.java\jdt_ws\ai_df4ca976\bin' 'LeapYearChecker'</div><div>Enter a year: 2024</div><div>2024 is a leap year.</div><div>PS C:\Users\nitin\OneDrive\Desktop\ai> </div></div></div>	
	<div><div>Task 2: One-Shot Prompting – Centimeters to Inches Conversion</div><div>Scenario</div><div>One-shot prompting guides AI using a single example.</div><div>Task Description</div><div>Use one-shot prompting by providing one input-output example to generate a Python function that:</div><div><ul style="list-style-type: none">• Converts centimeters to inches• Uses the correct mathematical formula</div><div>Example provided in prompt:</div><div>Input: 10 cm → Output: 3.94 inches</div><div>Expected Output</div></div>	



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"
- "Anita Rao" → "Rao, Anita"

Expected Output

```
J NameFormatter.java > Language Support for Java(TM) by Red Hat > NameFormatter
1 //generate a java code that accept the full name as input as a single string and then format it to "Last Name, First N
2 import java.util.Scanner;
3
4 public class NameFormatter {
5     public static void main(String[] args) {
6         Scanner scanner = new Scanner(System.in);
7         System.out.print(s: "Enter your full name: ");
8         String fullName = scanner.nextLine();
9
10        String formattedName = formatName(fullName);
11        System.out.println("Formatted name: " + formattedName);
12
13        scanner.close();
14    }
15
16    public static String formatName(String fullName) {
17        String[] parts = fullName.trim().split(regex: "\\s+");
18        if (parts.length < 2) {
19            return fullName;
20        }
21
22        String lastName = parts[parts.length - 1];
23        StringBuilder firstNameAndMiddleNames = new StringBuilder();
24
25        for (int i = 0; i < parts.length - 1; i++) {
26            firstNameAndMiddleNames.append(parts[i]).append(str: " ");
27        }
28
29        return lastName + ", " + firstNameAndMiddleNames.toString().trim();
30    }
31 }
```

PROBLEMS 15 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\nitin\OneDrive\Desktop\ai> c:; cd 'c:\Users\nitin\OneDrive\Desktop\ai'; & 'C:\Program Fi
les\Java\jdk-25\bin\java.exe' '-enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\
Users\nitin\AppData\Roaming\Code\User\workspaceStorage\d5b583de86868a227dcb4815ce34c8c7\redhat.java\j
dt_ws\ai_df4ca976\bin' 'NameFormatter'
Enter your full name: bodasu nithin
Formatted name: nithin, bodasu
PS C:\Users\nitin\OneDrive\Desktop\ai>
```

Task 4: Comparative Analysis – Zero-Shot vs Few-Shot Scenario

Different prompt strategies may produce different code quality.

Task Description

- 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:
 - Accuracy
 - Readability
 - Logical clarity

Expected Output

1.vowel count using zero shot

```

public class VowelCounter {
    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print(s: "Enter a string: ");
        String input = scanner.nextLine();

        int count = countVowels(input);
        System.out.println("Number of vowels in the string: " + count);

        scanner.close();
    }

    public static int countVowels(String str) {
        int count = 0;
        String vowels = "aeiouAEIOU";

        for (int i = 0; i < str.length(); i++) {
            if (vowels.indexOf(str.charAt(i)) != -1) {
                count++;
            }
        }

        return count;
    }
}

```

PROBLEMS 16 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

C:\Users\nitin\OneDrive\Desktop\ai> c:: cd 'c:\Users\nitin\OneDrive\Desktop\ai'; & 'C:\Program F
Java\jdk-25\bin\java.exe' '-enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:
s\nitin\AppData\Roaming\Code\User\workspaceStorage\d5b583de86868a227dcb4815ce34c8c7\redhat.java\
s\ai_df4ca976\bin' 'VowelCounter'
Enter a string: abhiiramesh
Number of vowels in the string: 5
C:\Users\nitin\OneDrive\Desktop\ai>

```

2.vowelcount using few shot

```

//generate te java code to count the number of vowels in a given string and take input from user for ex ample input "he
import java.util.Scanner;

public class VowelCounter {
    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print(s: "Enter a string: ");
        String input = scanner.nextLine();

        int count = countVowels(input);
        System.out.println("Number of vowels in the string: " + count);

        scanner.close();
    }

    public static int countVowels(String str) {
        int count = 0;
        String vowels = "aeiouAEIOU";

        for (int i = 0; i < str.length(); i++) {
            if (vowels.indexOf(str.charAt(i)) != -1) {
                count++;
            }
        }

        return count;
    }
}

```

```

PS C:\Users\nitin\OneDrive\Desktop\ai> c:: cd 'c:\Users\nitin\OneDrive\Desktop\ai'; & 'C:\Program F
Java\jdk-25\bin\java.exe' '-enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:
Users\nitin\AppData\Roaming\Code\User\workspaceStorage\d5b583de86868a227dcb4815ce34c8c7\redhat.java\
dt_ws\ai_df4ca976\bin' 'VowelCounter'
Enter a string: eeejjiiieelllloooiii
Number of vowels in the string: 13
PS C:\Users\nitin\OneDrive\Desktop\ai>

```

Task 5: Few-Shot Prompting – File Handling

Scenario

File processing requires clear logical understanding.

Task Description

Use few-shot prompting to generate a Python function that:

- Reads a .txt file
- Counts the number of lines in the file
- Returns the line count

Expected Output

The screenshot shows an IDE with a Java file named `LineCounter.java`. The code defines a `main` method that takes an array of strings `args` and a static method `countLinesInFile` that takes a file path and returns the number of lines. The `main` method calls `countLinesInFile` with the path `"C:\\Users\\nitin\\OneDrive\\Desktop\\sample\\sample.txt"` and prints the result. The `countLinesInFile` method uses a `BufferedReader` to read the file line by line and increments a counter. The terminal output shows the command `cd 'c:\\Users\\nitin\\OneDrive\\Desktop\\ai'; & 'C:\\Program Files\\Java\\jdk-25\\bin\\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\\Users\\nitin\\AppData\\Roaming\\Code\\User\\workspaceStorage\\d5b583de86868a227dcb4815ce34c8c7\\redhat\\.java\\jdt_ws\\ai_d5b583de86868a227dcb4815ce34c8c7\\bin' 'LineCounter'` and the output `Number of lines in the file: 5`.

```

1 // Generate a code function to count lines in a text file.Example:File contains 3 lines Output: 3
2 import java.io.BufferedReader;
3 import java.io.FileReader;
4 import java.io.IOException;
5 public class LineCounter {
6     public static void main(String[] args) {
7         String filePath = "C:\\Users\\nitin\\OneDrive\\Desktop\\sample\\sample.txt"; // Replace with your file path
8         try {
9             int lineCount = countLinesInFile(filePath);
10            System.out.println("Number of lines in the file: " + lineCount);
11        } catch (IOException e) {
12            System.err.println("An error occurred while reading the file: " + e.getMessage());
13        }
14    }
15
16    public static int countLinesInFile(String filePath) throws IOException {
17        int lineCount = 0;
18        try (BufferedReader reader = new BufferedReader(new FileReader(filePath))) {
19            while (reader.readLine() != null) {
20                lineCount++;
21            }
22        }
23        return lineCount;
24    }
25 }
26
27
28

```

PROBLEMS 16 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\nitin\OneDrive\Desktop\ai>
PS C:\Users\nitin\OneDrive\Desktop\ai> cd 'c:\Users\nitin\OneDrive\Desktop\ai'; & 'C:\Program Files\Java\jdk-25\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\nitin\AppData\Roaming\Code\User\workspaceStorage\d5b583de86868a227dcb4815ce34c8c7\redhat\.java\jdt_ws\ai_d5b583de86868a227dcb4815ce34c8c7\bin' 'LineCounter'
Number of lines in the file: 5
PS C:\Users\nitin\OneDrive\Desktop\ai>

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.