|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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** | | | | Venkataramana Veeramsetty | | | | | |
| **Instructor(s) Name** | | | | |  | | --- | | Dr. V. Venkataramana (Co-Ordinator) | | Dr. T. Sampath Kumar | | Dr. Pramoda Patro | | Dr. Brij Kishor Tiwari | | Dr.J.Ravichander | | Dr. Mohammand Ali Shaik | | Dr. Anirodh Kumar | | Mr. S.Naresh Kumar | | Dr. RAJESH VELPULA | | Mr. Kundhan Kumar | | Ms. Ch.Rajitha | | Mr. M Prakash | | Mr. B.Raju | | Intern 1 (Dharma teja) | | Intern 2 (Sai Prasad) | | Intern 3 (Sowmya) | | NS\_2 ( Mounika) | | | | | | |
| **Course Code** | | | 24CS002PC215 | **Course Title** | | AI Assisted Coding | | | |
| **Year/Sem** | | | II/I | **Regulation** | | R24 | | | |
| **Date and Day**  **of Assignment** | | | Week2 - Monday | **Time(s)** | |  | | | |
| **Duration** | | | 2 Hours | **Applicable to**  **Batches** | |  | | | |
| **Assignment Number:4.1**(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:**   * 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.   **Lab Outcomes (LOs):**  After completing this lab, students will be able to:   * 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 the three prompting strategies.   **Task #1 – Zero-Shot Prompting with Conditional Validation**  Objective  Use zero-shot prompting to instruct an AI tool to generate a function that validates an Indian mobile number.  Requirements   * The function must ensure the mobile number:   + Starts with 6, 7, 8, or 9   + Contains exactly 10 digits   PROMPT: Generate a python function that validates an Indian mobile number. It must Starts with 6, 7, 8, or 9,Contains exactly 10 digits. Ask the user to enter the number give me the output as true or false.  VSCODE:    CURSER:  Expected Output   * A valid Python function that performs all required validations without using any input-output examples in the prompt.   VSCODE:      CURSER:    **Task #2 – One-Shot Prompting with Edge Case Handling**  Objective  Use one-shot prompting to generate a Python function that calculates the factorial of a number.  Requirements   * Provide one sample input-output pair in the prompt to guide the AI. * The function should handle:   + 0! correctly   + Negative input by returning an appropriate message   PROMPT: Generate a Python function that calculates the factorial of a number. Negative input by returning an invalid input.  Example : Enter the integer:3  Factorial of given number is:6  VSCODE:    CURSER:    Expected Output   * A Python function with correct factorial logic and edge case handling, generated from a single example.   VSCODE:    CURSER:      **Task #3 – Few-Shot Prompting for Nested Dictionary Extraction**  Objective  Use few-shot prompting (2–3 examples) to instruct the AI to create a function that parses a nested dictionary representing student information.  Requirements   * The function should extract and return:   + Full Name   + Branch   + SGPA   PROMPT : Give me python function of nested Dictionary Extraction.  Example1:FullName: vadhan ganji, Branch:CSEAIML,SGPA:9.1  Example2:FullName: siddhu , Branch:EEE ,SGPA:7.1.  Example3:FullName: rishi , Branch:mechanical ,SGPA:8.5.  VSCODE:  CURSER:    Expected Output   * A reusable Python function that correctly navigates and extracts values from nested dictionaries based on the provided examples   VSCODE:    CURSER:    **Task #4 – Comparing Prompting Styles for File Analysis**  Objective  Experiment with zero-shot, one-shot, and few-shot prompting to generate functions for CSV file analysis.  Requirements   * Each generated function should:   + Read a .csv file   + Return the total number of rows   + Count the number of empty rows   + Count the number of words across the file   Expected Output   * Working Python functions for each prompting style, with a brief reflection comparing their accuracy, clarity, and efficiency.   **Task #5 – Few-Shot :Prompting for Text Processing and Word** **Frequency**  Objective  Use few-shot prompting (with at least 3 examples) to generate a Python function that processes text and analyzes word frequency.  Requirements  The function must:   * Accept a paragraph as input * Convert all text to lowercase * Remove punctuation * Return the most frequently used word   PROMPT: Generate a Python function that processes text and analyses word frequency. requirements The function must. Accept a paragraph as input .Convert all text to lowercase. Remove punctuation .Return the most frequently used word  Example1: VinayakaChavithi is a HOLY festival dedicated to Lord Ganesha. People CELEBRATE it with prayers, sweets, and colorful decorations. It brings JOY, UNITY, and SPIRITUAL energy to everyone’s home!  Example2: The INDIAN ARMY is one of the most powerful and respected forces in the world. It PROTECTS the nation with bravery, dedication, and HONOR. Every soldier stands for COURAGE, SACRIFICE, and DUTY to the motherland.  Example3: **Warangal** is a HISTORIC city in Telangana, known for its rich CULTURAL heritage. It was once the capital of the mighty KAKATIYA dynasty. Famous for its FORT, temples, and SCULPTURES, Warangal attracts many visitors!  VSCODE:    CURSER:      Expected Output   * A functional Python script that performs text cleaning, tokenization, and returns the most common word using only the examples provided in the prompt   VSCODE:  CURSER:  **Note:** Report should be submitted a word document for all tasks in a single document with prompts, comments & code explanation, and output and if required, screenshots  **Evaluation Criteria:**   | **Criteria** | **Max Marks** | | --- | --- | | Zero Shot (Task #1) | 0.5 | | One Shot (Task#2) | 0.5 | | Few Shot (Task#3, Task#4 & Task #5) | 1.5 | | **Total** | **2.5 Marks** | | | | | | | Week2 - Monday |  |