|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE** | | | | **DEPARTMENT OF COMPUTER SCIENCE ENGINEERING** | | |
| **Program Name:** M.Tech. and MCA | | | **Assignment Type: Lab** | | | **AcademicYear:**2025-2026 |
| **Course Coordinator Name** | | | Venkataramana Veeramsetty | | | |
| **Course Code** | |  | **Course Title** | | AI Assisted Problem Solving Using Python | |
| **Year/Sem** | | I/I | **Regulation** | | R24 | |
| **Date and Day**  **of Assignment** | | Week3 - Monday | **Time(s)** | |  | |
| **Duration** | | 2 Hours | **Applicable to**  **Batches** | | M.Tech. and MCA | |
| **AssignmentNumber:8.3**(Present assignment number)/**24**(Total number of assignments) | | | | | | |
| **ASSIGNMENT – 8**  **Name: Rimsha Mujeeb Roll Number:2503B05138 ( M.Tech - CSE)** | | | | | | |
|  | **Q.No.** | **Question** | | | | |  |  |
|  | ***Q1.***  ***Q2.***  ***Q3.***  ***Q4.***  ***Q5.*** | **Task Description#1**  **Use AI to generate test cases for is\_valid\_email(email) and then implement the validator function.**  **Requirements:**   * **Must contain @ and . characters.** * **Must not start or end with special characters.** * **Should not allow multiple @.**   **Expected Output#1**   * **Email validation logic passing all test cases**         **Task Description#2 (Loops)**   * Ask AI to generate test cases for assign\_grade(score) function. Handle boundary and invalid inputs.   **Requirements**   * AI should generate test cases for assign\_grade(score) where: 90-100: A, 80-89: B, 70-79: C, 60-69: D, <60: F * Include boundary values and invalid inputs (e.g., -5, 105, "eighty").   **Expected Output#2**  Grade assignment function passing test suite  ***PROMPT:***  # Python: Generate test cases and implementation for assign\_grade(score).  # Grades: A (90-100), B (80-89), C (70-79), D (60-69), F (<60).  # Include boundary cases (60, 89, 100) and invalid inputs (-5, 105, "text").  # Then, implement the function and add code to take user input for the score and print the result.      **Task Description#3**   * Generate test cases using AI for is\_sentence\_palindrome(sentence). Ignore case, punctuation, and spaces   **Requirement**   * Ask AI to create test cases for is\_sentence\_palindrome(sentence)   (ignores case, spaces, and punctuation).   * Example:   "A man a plan a canal Panama" → True  **Expected Output#3**   * Function returns True/False for cleaned sentences * Implement the function to pass AI-generated tests.   ***PROMPT:***  # Python: Generate test cases and implementation for is\_sentence\_palindrome(sentence).  # Ignore case, punctuation, and spaces.  # Include test cases like "A man a plan a canal Panama" → True.  # Then, implement the function and add code to take user input for the sentence and print the result.      **Task Description#4**   * Let AI fix it Prompt AI to generate test cases for a ShoppingCart class (add\_item, remove\_item, total\_cost).   **Methods:**  Add\_item(name,orice)  Remove\_item(name)  Total\_cost()  **Expected Output#4**   * Full class with tested functionalities   ***PROMPT:***  Generate Python code for a *ShoppingCart system* with an inventory. Let customers view available items, add or remove items from their cart, and display their total bill. Include methods: add\_item(name, price), remove\_item(name), and total\_cost(). Add AI-generated test cases to verify each method and include an interactive menu (view inventory, add, remove, display cart, exit).            OUTPUT:          **Task Description#5**   * Use AI to write test cases for convert\_date\_format(date\_str) to switch from "YYYY-MM-DD" to "DD-MM-YYYY".   **Example: "2023-10-15" → "15-10-2023"**  **Expected Output#5**   * Function converts input format correctly for all test cases   ***PROMPT:***  Write a Python function convert\_date\_format(date\_str) that converts a date from "YYYY-MM-DD" format to "DD-MM-YYYY". Example: "2023-10-15" → "15-10-2023".    ***OUTPUT:*** | | | | |  |  |