|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **AI ASSISTED CODING**  NAME: J.KEERTHI PRIYA  ROLL NO:2403A510G4  ASSIGNMENT : 8.3 |  |  |
|  | 1 | **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   **PROMPT:**  write a python function using to generate gmail Adress code  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 @.  **CODE:**    **OUTPUT:**  **A screen shot of a computer program  AI-generated content may be incorrect.**  **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  **PROMT:**  write a python code  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").  **CODE:**    **A screen shot of a computer  AI-generated content may be incorrect.**  **OUTPUT:**  **A screenshot of a computer  AI-generated content may be incorrect.**  **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:**  Write a python code 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.  **CODE:**    **OUTPUT:**  **A screenshot of a computer  AI-generated content may be incorrect.**  **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:**  Write a python program to generate test cases for a ShoppingCart class (add\_item, remove\_item, total\_cost).  Methods:  Add\_item(name,orice)  Remove\_item(name)  Total\_cost() . give the code dynamically  **CODE:**    A screen shot of a computer program  AI-generated content may be incorrect.  **OUTPUT:**  **A screenshot of a computer  AI-generated content may be incorrect.**  **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 program to generate convert\_date\_format(date\_str) to switch from "YYYY-MM-DD" to "DD-MM-YYYY".  Example: "2023-10-15" → "15-10-2023". give code dynamically  **CODE:**    **OUTPUT:A screenshot of a computer  AI-generated content may be incorrect.**  **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** | | --- | --- | | Task #1 | 0.5 | | Task #2 | 0.5 | | Task #3 | 0.5 | | Task #4 | 0.5 | | Task #5 | 0.5 | | **Total** | **2.5 Marks** | |  |  |