|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE** | | | | | **DEPARTMENT OF COMPUTER SCIENCE ENGINEERING** | | | | |
| **ProgramName:**B. Tech | | | | **Assignment Type: Lab** | | | **AcademicYear:**2025-2026 | | |
| **CourseCoordinatorName** | | | | 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) | | | | | | |
| **CourseCode** | | | 24CS002PC215 | **CourseTitle** | | AI Assisted Coding | | | |
| **Year/Sem** | | | II/I | **Regulation** | | R24 | | | |
| **Date and Day**  **of Assignment** | | | Week4 - Wednesday | **Time(s)** | |  | | | |
| **Duration** | | | 2 Hours | **Applicableto**  **Batches** | |  | | | |
| **AssignmentNumber:9.3**(Present assignment number)/**24**(Total number of assignments) | | | | | | | | | |
|  | | | | | | | | | |
|  | | | | | | | | | |
|  | **Q.No.** | **Question** | | | | | | ***ExpectedTime***  ***to complete*** |  |
|  | 1 | Lab 8: Documentation Generation: Automatic documentation and code comments  **Lab Objectives:**   * To understand the importance of documentation and code comments in software development. * To explore how AI-assisted coding tools can generate meaningful documentation and inline comments. * To practice generating function-level and module-level docstrings automatically. * To evaluate the quality, accuracy, and limitations of AI-generated documentation. * To develop a small automated tool for documentation generation in Python..     **Lab Outcomes (LOs):**  After completing this lab, students will be able to:   * Apply AI-assisted coding tools to generate docstrings and inline comments for Python code. * Critically analyze AI-generated documentation for correctness, completeness, and readability. * Create structured documentation (function-level, module-level) following standard formats. * Design and implement a mini documentation generator tool to automate code commenting and docstring creation.   **Task Description#1 Basic Docstring Generation**   * Write python function to return sum of even and odd numbers in the given list. * Incorporate manual **docstring** in code with Google Style * Use an AI-assisted tool (e.g., Copilot, Cursor AI) to generate a docstring describing the function. * Compare the AI-generated docstring with your manually written one.   **Prompt:**  **Provide a python code to print and return the sum of even and odd numbers.**    **Prompt:** **"Develop a Python function that computes and displays the cumulative sum of even and odd integers from a given input collection."**    **Expected Outcome#1:** Students understand how AI can produce function-level documentation.  **OBSERVATION :**  **The AI-generated docstring is well-structured and follows standard documentation practices. It clearly explains the function’s purpose, inputs, and outputs, making it easy to understand for other developers. In contrast, a manually written docstring might be shorter or less formal, depending on the programmer’s style. While manual docstrings can be more personalized, the AI version tends to be more consistent and professional, which is helpful for collaborative or large-scale projects.**  **Task Description#2 Automatic Inline Comments**   * Write python program for **sru\_student** class with attributes like name, roll no., hostel\_status and **fee\_update** method and **display\_details** method. * Write comments manually for each line/code block * Ask an AI tool to add inline comments explaining each line/step. * Compare the AI-generated comments with your manually written one.   **Prompt:**  **- "Create a Python program that defines a class SRUStudent with instance variables for student name, roll number, and hostel status. The class should include a method to update fee status and another to print student details."**    **Now, add inline comments:**    **Expected Output#2:** Students critically analyze AI-generated code comments.  **Observation:**  **Manual comments are often personalized, brief, and vary in clarity depending on the developer’s experience. They may skip obvious steps or reflect informal tone and inconsistent formatting.**   * **AI-generated comments are typically more structured, formal, and consistent. They follow documentation standards, explain each step clearly, and are well-suited for collaborative or professional environments.** * **While manual comments offer flexibility and human insight, AI-generated comments provide clarity and uniformity—making them ideal for clean, maintainable code.**   **Task Description#3**   * Write a Python script with 3–4 functions (e.g., calculator: add, subtract, multiply, divide). * Incorporate manual **docstring** in code with NumPy Style * Use AI assistance to generate a module-level docstring + individual function docstrings. * Compare the AI-generated docstring with your manually written one.   **Prompt:**  **"Write a Python program that performs basic arithmetic operations based on user input."**    **Write a code using numpy style "Write a Python program that performs basic arithmetic operations based on user input."**    **Expected Output#3:** Students learn structured documentation for multi-function scripts  **Observation:**  AI-written docstrings are useful for quickly creating a rough version, especially when working with lots of code or just starting out. But when you need clear, detailed, and reliable explanations, writing them yourself is better. The smartest way to work is to let AI give you a first draft, then improve it by hand to make sure it's accurate and complete.  **Push documentation whole workspace as .md file in GitHub Repository**  **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** | | | | | | Week4 - Wednesday |  |