|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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** | | | | Dr. Rishabh Mittal | | | | | |
| **Instructor(s) Name** | | | | |  | | --- | | Mr. S Naresh Kumar | | Ms. B. Swathi | | Dr. Sasanko Shekhar Gantayat | | Mr. Md Sallauddin | | Dr. Mathivanan | | Mr. Y Srikanth | | Ms. N Shilpa | | Dr. Rishabh Mittal (Coordinator) | | Dr. R. Prashant Kumar | | Mr. Ankushavali MD | | Mr. B Viswanath | | Ms. Sujitha Reddy | | Ms. A. Anitha | | Ms. M.Madhuri | | Ms. Katherashala Swetha | | Ms. Velpula sumalatha | | Mr. Bingi Raju | | | | | | |
| **CourseCode** | | | 23CS002PC304 | **Course Title** | | AI Assisted Coding | | | |
| **Year/Sem** | | | III/II | **Regulation** | | R23 | | | |
| **Date and Day**  **of Assignment** | | | **Week5 – Monday** | **Time(s)** | | 23CSBTB01 To 23CSBTB52 | | | |
| **Duration** | | | 2 Hours | **Applicable to**  **Batches** | | All batches | | | |
| **Assignment Number: 9.1**(Present assignment number)/**24**(Total number of assignments) | | | | | | | | | |
|  | | | | | | | | | |
|  | **Q.No.** | **Question** | | | | | | ***Expected Time***  ***to complete*** |  |
|  | 1 | **Lab Experiment: Documentation Generation -Automatic documentation and code comments**  **Lab Objectives**   1. To understand automatic documentation generation. 2. To generate code comments and docstrings using AI tools. 3. To learn the importance of documentation in software development.   **Lab Outcomes**   1. Students will be able to generate documentation automatically for code. 2. Students will be able to add clear comments and docstrings to programs. 3. Students will be able to improve code readability and maintainability using documentation.   **Problem 1:**  Consider the following Python function:  def find\_max(numbers):  return max(numbers)  **Task:**   * Write documentation for the function in all three formats:   (a) Docstring (b) Inline comments (c) Google-style documentation   * Critically compare the three approaches. Discuss the advantages, disadvantages, and suitable use cases of each style. * Recommend which documentation style is most effective for a **mathematical utilities library** and justify your answer. | | | | | | Week5 -Monday |  |
|  |  | **Problem 2:** Consider the following Python function:  def login(user, password, credentials):  return credentials.get(user) == password  **Task:**   1. Write documentation in all three formats. 2. Critically compare the approaches. 3. Recommend which style would be most helpful for new developers onboarding a project, and justify your choice. | | | | | |  |  |
|  |  | **Problem 3: Calculator (Automatic Documentation Generation)**  Task: Design a Python module named calculator.py and demonstrate automatic documentation generation.  Instructions:   1. Create a Python module calculator.py that includes the following functions, each written with appropriate docstrings:    * add(a, b) – returns the sum of two numbers    * subtract(a, b) – returns the difference of two numbers    * multiply(a, b) – returns the product of two numbers    * divide(a, b) – returns the quotient of two numbers 2. Display the module documentation in the terminal using Python’s documentation tools. 3. Generate and export the module documentation in HTML format using the pydoc utility, and open the generated HTML file in a web browser to verify the output.   **Problem 4: Conversion Utilities Module**  **Task:**   1. Write a module named conversion.py with functions:    * decimal\_to\_binary(n)    * binary\_to\_decimal(b)    * decimal\_to\_hexadecimal(n) 2. Use Copilot for auto-generating docstrings. 3. Generate documentation in the terminal. 4. Export the documentation in HTML format and open it in a browser.   **Problem 5 – Course Management Module**  **Task:**   1. Create a module course.py with functions:    * add\_course(course\_id, name, credits)    * remove\_course(course\_id)    * get\_course(course\_id) 2. Add docstrings with Copilot. 3. Generate documentation in the terminal. 4. Export the documentation in HTML format and open it in a browser. | | | | | |  |  |