|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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** | | | Week3 - Wednesday | **Time(s)** | |  | | | |
| **Duration** | | | 2 Hours | **Applicableto**  **Batches** | |  | | | |
| **AssignmentNumber:6.3**(Present assignment number)/**24**(Total number of assignments) | | | | | | | | | |
|  | | | | | | | | | |
|  | **Q.No.** | **Question** | | | | | | ***ExpectedTime***  ***to complete*** |  |
|  | 1 | Lab 6: AI-Based Code Completion – Classes, Loops, and Conditionals  **Lab Objectives:**   * To explore AI-powered auto-completion features for core Python constructs. * To analyze how AI suggests logic for class definitions, loops, and conditionals. * To evaluate the completeness and correctness of code generated by AI assistants.   **Lab Outcomes (LOs):**  After completing this lab, students will be able to:   * Use AI tools to generate and complete class definitions and methods. * Understand and assess AI-suggested loops for iterative tasks. * Generate conditional statements through prompt-driven suggestions. * Critically evaluate AI-assisted code for correctness and clarity.   **Task Description#1 (Classes)**   * Use AI to complete a Student class with attributes and a method. * Check output * Analyze the code generated by AI tool   **GITHUB CODE:**    **Instructions**:   * **Initialize class with attributes like name, roll no, marks** * **Method to display student details** * **Method to calculate grade based on marks (A:>=90, B: >=75, C: >=60, else Fail)**   Start Writing code and auto complete using any AI tool  **Expected Output#1**   * Class with constructor and display\_details() method   **GITHUB OUTPUT:**    EXPLANATION:  This code defines a simple student class and creates an instance to store a student's name, roll number, and marks. It assigns a grade based on the marks entered and then prints all the student details. The class is used only as a container for attributes, without any methods.  **Task Description#2 (Loops)**   * Prompt AI to complete a function that prints the first 10 multiples of a number using a loop. * Analyze the generated code * Ask AI to generate code using other controlled looping   Write code using **For** Loop, later complete code using **While** Loop  **GITHUB CODE:**    **Expected Output#2**   * Correct loop-based implementation   **GITHUB OUTPUT:**    **EXPLANATION:**  This code defines two functions to print the first 10 multiples of a user-input number: one uses a for loop and the other uses a while loop. Both functions prompt the user for a number and display its multiples from 1 to 10 in a formatted way. The for loop iterates over a fixed range, while the while loop uses a counter variable. This demonstrates two different looping techniques for the same task.  **Task Description#3 (Conditional Statements)**   * Ask AI to write nested if-elif-else conditionals to classify age groups. * Analyze the generated code * Ask AI to generate code using other conditional statements   **GITHUB CODE:**    **Expected Output#3**   * Age classification function with appropriate conditions and with explanation   **GITHUB OUTPUT:**    **EXPLANATION:**  This code defines a function [classify\_age()](vscode-file://vscode-app/c:/Users/SIDDHARTHA/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html" \o ") that asks the user to input their age and then classifies them into categories: Child (0-12), Teen (13-19), Adult (20-59), or Senior (60+). It uses nested if-elif-else statements to check the age range and print the appropriate category. If a negative age is entered, it prints an error message. The function is called at the end to execute the classification. This approach ensures clear age-based categorization and input validation.  **Task Description#4 (For and While loops)**   * Generate a sum\_to\_n() function to calculate sum of first n numbers * Analyze the generated code * Get suggestions from AI with other controlled looping   **GITHUB CODE:**    **Expected Output#4**   * Python code with explanation   **GITHUB OUTPUT:**    **EXPLANATION:**  **This code provides two functions to calculate the sum of the first**[**n**](vscode-file://vscode-app/c:/Users/SIDDHARTHA/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)**natural numbers, one using a for loop and the other using a while loop. Both functions initialize a**[**total**](vscode-file://vscode-app/c:/Users/SIDDHARTHA/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)**variable and add each number from 1 to**[**n**](vscode-file://vscode-app/c:/Users/SIDDHARTHA/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html)**to it. The user is prompted to enter a number, and the sum is displayed for each method. This demonstrates how both loop types can be used to solve the same problem.**  **Task Description#5 (Class)**   * Use AI to build a BankAccount class with deposit, withdraw, and balance methods. * Analyze the generated code * Add comments and explain code   **Instructions**   * **Initialize BankAccount class with attributes like name, balance** * **Method to deposit amount** * **Method to withdraw amount** * **Method to check balance**   **GITHUB CODE:**    **Expected Output#5**   * Python code with explanation   **GITHUB OUTPUT:**    EXPLANATION:  This code defines a [BankAccount](vscode-file://vscode-app/c:/Users/SIDDHARTHA/AppData/Local/Programs/Microsoft%20VS%20Code/resources/app/out/vs/code/electron-browser/workbench/workbench.html" \o ") class with methods to deposit, withdraw, and check balance. The user is prompted for their name and starting balance, then interacts with the account through a menu-driven loop. Deposits and withdrawals update the balance, with a check to prevent overdrawing. The loop continues until the user chooses to exit, providing a simple simulation of basic banking operations.  **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** | | --- | --- | | Class | 1.0 | | Loops | 1.0 | | Conditional Statements | 0.5 | | **Total** | **2.5 Marks** | | | | | | | Week3 - Wednesday |  |