

|  |  |  |
| --- | --- | --- |
|  | **Task Description#1**   * Try 3 different prompts to generate a factorial function.   **Expected Output#1**   * Comparison of AI-generated code styles     Prompt 1: write python function to calculate factorial        OUTPUT:      Prompt 2:  Write python function that take input from user and return factorial of the number user entered      OUTPUT:    Prompt 3: Create a Python program to compute the factorial of a number using recursion with proper function docstring. |  |

|  |  |  |
| --- | --- | --- |
|  | OUTPUT:    COMPARISION:  **First prompt** defines a reusable factorial(n) function with input validation and computes factorial using a loop.  **Second prompt**  combines user input handling and factorial computation within the get\_user\_factorial() function.  **Third prompt** is identical to the second one—also takes user input, validates it, and computes the factorial, showing no structural difference.      **Task Description#2**   * Provide a clear example input-output prompt to generate a sorting function.   **Expected Output#2**   * Functional sorting code from AIPROMPT:   Write a Python function that takes a list of integers as input and returns the list sorted in descending order |  |

|  |  |  |
| --- | --- | --- |
|  | **OUTPUT:**    **Task Description#3**   * Start with the vague prompt “Generate python code to calculate power bill” and improve it step-by-step   **Expected Output#3**   * Enhanced AI output with clearer prompts   Prompt 1: Generate python code to calculate power bill |  |

|  |  |  |
| --- | --- | --- |
|  | **OUTPUT:**    **Prompt 2: generate python code to calculate power bill using functions** |  |

|  |  |  |
| --- | --- | --- |
|  | **OUTPUT:**    **COMPARISION:**  **First code uses tiered rates with fixed charges; second uses a single rate.First takes only units; second takes units and rate as inputs.First has conditional logic for billing slabs; second uses simple multiplication.First aims for detailed billing; second is straightforward and flexible.**      **Task Description#4**   * Write structured comments to help AI generate two linked functions (e.g., login\_user() and register\_user()).   **Expected Output#4**   * Consistent functions with shared logic   PROMPT: generate two linked functions (Celsius ,Fahrenheit) |  |

|  |  |  |
| --- | --- | --- |
|  | OUTPUT:        **Task Description#5**   * Analyzing Prompt Specificity: Improving Temperature Conversion Function with Clear Instructions   **Expected Output#5**   * Code quality difference analysis for various prompts     PROMPT 1:Write a Python program to convert Celsius to Fahrenheit |  |

|  |  |  |
| --- | --- | --- |
|  | **OUTPUT:**    Prompt 2: Generate e a python code for temperature conversion |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **OUTPUT:**      **COMPARISION:**  **The first code converts only Celsius to Fahrenheit, while the second supports both Celsius to Fahrenheit and Fahrenheit to Celsius conversions.The first code has a single conversion function; the second has two separate functions for each conversion.The second code includes a menu for user choice, adding flexibility.**  **Both handle invalid inputs with try-except blocks and print formatted results.**      **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** |  | | Factorial Function (Task#1) | 0.5 | | Sorting Function (Task#2) | 0.5 | | Vogue Vs. Specific Prompting (Task #3) | 0.5 | | Linked Functions (Task #4) | 0.5 | | Temperature Conversion Function (Task #5) | 0.5 | | **Total** | **2.5 Marks** | |  |