B.P4

SDLC – Software Development Life Cycle
1. Take at least 2 of the SDLC:
2. Definition
3. Pros and Cons
4. Which one would you recommend for our scenario (assignment) and why?

Requirements:
- Welcome Message
- Display all the info
- Get individual names up to 20
- Ask if joining as Team or Individual
- Scoring system for Team and Individual
- Get the ranking for Team

Designing:
1. Algorithm
2. Pseudo code
3. Flowchart
4. Inputs/Outputs
5. Testing

Algorithm:
- Application will Display a message such as Welcome Deivi's Leader Board
- Tell the user what inputs needed such choosing Team name, member names for each team
- Individual names, Scoring point, Events details
- The application should prompt user to input up to 20 names.

Pseudo Code:
- Output: "Welcome to Deivie's Application"
- Output: "Display all the necessary information so that user can make an informative choice "
- Input: "Get 20 individual names"
- Inputs/Outputs:
- Print 'welcome message'
- Print 'All the info'
- String input '20 names'

B.D3

Create a plan:
- Timescale of the completing the assignment
- Dates and Times would be invested
- Keep a diary

BC.M2

- Justify your Designing above.

BC.D2

- Evaluate your Designing

BC.D3

- Explain what self-study you did to complete this assignment.
Such as:
  - Did you watch videos online?
  - Did you read any book?
  - Did you have a conversation with your colleagues?
  - Where did you get support outside class if you needed?

B.P4

- Data Validation
  - Look at your Inputs and Outputs and write why type of data will be used for inputs particularly.
    E.g: Team_name will be string not a numerical data.

B.P4

- Why do you think Python is a good choice for this scenario? Please explain.
- What will be the disadvantages of using Python for this scenario? Please explain.

B.P5

You need to work in a small group of minimum with two people to review the design of your application. How many meetings were hold?

Date Of Meeting	People Attending	Purpose of Meeting	Completed

B.P5

- Minutes of Meeting #1
  - Date of Meeting
  - Those Present
  - Agenda Items
  - Action Points
  - Date and Time of Next Meeting

Repeat this task according to number of the meetings.

BC.D3

- Explain what was your contribution for the meetings above?
- Who chaired meeting?
- Who took notes and were they correct?

B.P5

Identify Improvements
- What improvements did you decide to implement in your design after the meeting(s)?
- Were they helpful improvements or essentials ones?
- Did you give feedback to your colleagues?
- Did you receive feedback from your colleagues? If so, what was the feedback?

BC.M2

Justify Improvements
- Can you justify why the improvements were essentials? If not, explain why there were not.

BC.D2

- Evaluate Improvement?
- List all the pros and cons of the improvements.
- Conclude how the improvements made the application a better one.

BC.D3

- How do you evaluate your personal contribution?
- What responsibility did you take?
- What creativity did you offer?

C.P6

- The development Environment
  - Take screenshot of your output and display how you run the application from beginning to the end.
  - Annotate each screenshot by explain what the code is doing
- Copy and Paste your actually code
- List all the functions that you created
  - E.g: print, import math…
- List all the function that are built-in
  - E.g. def Event1()

BC.D3

- Tell us what new skills you acquire to complete this assignment?
- What did you teach yourself for this assignment that you did not know before?

C.P6

- Functional Testing
- Create a list of errors you come across while you were coding?
- How did you solve them?
- Take screenshots the error you come across
- Did you use your testing table to test your code?
- If so, show us screenshot of testing

C.M3

- Optimising Software
  - How the testing helped to optimise your code (make it accurate, strong, less error prone?
  - You need to justify how these will result in an effective solution

BC.D2

- Evaluate Optimisation

- How did the testing table help you to find out the errors?
- How did the testing table help you to debug?
- Can you recommend for new programmer to test their applications? If so, give details how would they could do that?

BC.D3

- What did you learn in previous task?
- How did help you to develop yourself?
- Did you look into any different testing techniques?
- Did you find out?
- What is a software tester?
- What jobs do they do?
- What is the salary like?
- Would you like to be a software tester? Explain

C.M3

- User Feedback
  - Get some (two or three) IT users (your parents, siblings, friends who do no coding) to run your application.
  - Get their opinion and record them here.

- User 1: What action point did you take after the feedback from user 1?
- User 2: What action points did you take after the feedback from user 2?
- Take screenshots before you amended the code after action points carried out?
- Take screenshots after you amended the code?

BC.D3

- Record your interactions with users above?
- What did you learn from them?
- What did you explain them?
- What were their reactions while running your code?
- What questions did they ask and how you answered them?

BC.D2

- Evaluate Design improvements after the feedback

  - After the amendment of the code (after user feedback)
  - What does it work better now?
  - What pros and cons the amendment provided?

C.P7

- Review of Solution
  - How suitable is your code for different audience?
  - Is it user friendly?
  - Is it easy to understand what inputs need to be entered at every stage of your code?

C.P7

- Quality of the application:
  - How reliable is your application? Explain
  - How efficient is your application? Explain
  - How is the performance of your application like? Explain
  - How is it easy to maintain the application (new version)? Explain
  - How is it easy to make the application portable? Explain

  - What are the drawback of your application? Explain
  - What else could be done to develop this application better? Any suggestion, ideas.
  - What other programming language could have been used to develop this application? Explain
  - What error handling methods did you implement?
  - Any suggestion to deal with errors? For example, wrong inputs.

BC. D3

- Please explain followings:

  - Write a report after you complete the application to describe your personal development?
  - Has this application helped you to be a better developer?
  - Has this application helped you to learn more about Python language?
  - Has this application helped you to work as part of team?
  - Has this application helped you to work with non-experienced user?
  - Any other personal quality you gained?