## **Problem Statement**

**Intern Infotech Virtual Learning Internship Program** 



## **Decoding Gaming Behaviour**

In this internship, you will be working with a dataset related to a game. The dataset includes two tables: `Player Details` and `Level Details`. Below is a brief description of the dataset and the tasks you need to perform:

### **Dataset Description:**

#### Player Details Table:

'P\_ID': Player ID

• `PName`: Player Name

• `L1\_status`: Level 1 Status

• `L2 status`: Level 2 Status

• `L1\_code`: Systemgenerated Level 1 Code

• `L2 code`: Systemgenerated Level 2 Code

#### **Level Details Table:**

`P ID`: Player ID

Dev ID: Device ID

• `start time`: Start Time

• `stages crossed`: Stages Crossed

• `level`: Game Level

• `difficulty`: Difficulty Level

• `kill\_count`: Kill Count

headshots count: Headshots Count

'score': Player Score

'lives\_earned': Extra Lives Earned

#### What you have to do?

Below are 15 questions for which you have to find the answers by writing SQL queries. Each question carries 2 marks.

1. Extract `P\_ID`, `Dev\_ID`, `PName`, and `Difficulty\_level` of all players at Level 0.

# **Problem Statement**

# Intern Infotech

#### **Intern Infotech Virtual Learning Internship Program**

- 2. Fins the total number of stages crossed at each difficulty level for Level 2 with players.
- 3. Find `Level1\_code` wise average `Kill\_Count` where `lives\_earned` is 2, and at least 3 stages are crossed.
  - using `zm\_series` devices. Arrange the result in decreasing order of the total number of stages crossed.
- 4. Extract `P\_ID` and the total number of unique dates for those players who have played games on multiple days.
- 5. Find `P\_ID` and levelwise sum of `kill\_counts` where `kill\_count` is greater than the average kill count for Medium difficulty.
- 6. Find `Level` and its corresponding `Level\_code` wise sum of lives earned, excluding Level 0. Arrange in ascending order of level.
- 7. Find the top 3 scores based on each `Dev\_ID` and rank them in increasing order using `Row\_Number`. Display the difficulty as well.
- 8. Find the 'first login' datetime for each device ID.
- 9. Find the top 5 scores based on each difficulty level and rank them in increasing order using `Rank`. Display `Dev\_ID` as well.
- 10. Find the device ID that is first logged in (based on `start\_datetime`) for each player (`P\_ID`). Output should contain player ID, device ID, and first login datetime.
- 11. For each player and date, determine how many `kill\_counts` were played by the player so far.
  - a) Using window functions
  - b) Without window functions
- 12. Find the cumulative sum of stages crossed over `start\_datetime` for each `P\_ID`, excluding the most recent `start\_datetime`.
- 13. Extract the top 3 highest sums of scores for each 'Dev ID' and the corresponding 'P ID'.
- 14. Find players who scored more than 50% of the average score, scored by the sum of scores for each `P ID`.
- 15. Create a stored procedure to find the top `n` `headshots\_count` based on each `Dev\_ID` and rank them in increasing order using `Row\_Number`. Display the difficulty as well.