RPG Game - Loops

1. Calculate Total Player Experience

- Loop through the array `experiencePoints = [100, 150, 75]` to sum the experience of all completed quests. Log the total.

2. Locate Quest by Title

- Search through `questTitles = ["Quest 1", "Quest 2", "Quest 3"]` for "Quest 2" and log its index.

3. List Enemies Above Health Threshold

- In `enemyHealths = [80, 120, 65]`, identify enemies with health over 100 and log their indices.

4. Enhance All Quest Rewards

- Given `questRewards = [50, 100, 75]`, increase each reward by 20. Log the updated rewards.

5. Count Occurrences of a Specific Reward

- Within `questRewards = [50, 100, 75, 100]` and given `specificReward = 100`, count how many times this reward appears.

6. Verify All Quests Offer Sufficient Challenge

- Given `questChallenges = [120, 150, 75]` and a challenge threshold of 100, write a loop to verify if all quests challenges in the array have challenges above this threshold. Log whether this is true or false.

7. Discover the Quest with the Maximum Reward

- From `questRewards = [50, 100, 75, 150]`, use a loop to identify the quest offering the highest reward and log the reward value along with the quest's index.

8. Combine Enemy Powers for a Total Assault Value

- For `enemyPowers = [50, 80, 65, 90]`, and `enemyHealths = [80, 120, 60, 100]`, calculate the total assault value by summing the product of each enemy's power and health. Log this total assault value.

Bonus - RPG Context with 2D Arrays

9. Calculate Total Experience from Quest Matrix

- Given a 2D array `questExperienceMatrix = [[100, 200], [150, 175], [120, 250]]` where each sub-array contains experience points from different quests, sum all experience points and log the total.

10. Find the Quest with Highest Experience in Each Category

- In a matrix `questExperienceMatrix = [[100, 200], [150, 175], [120, 250]]`, identify the quest with the highest experience points in each category (column) and log the highest points per category.

11. Count High-Difficulty Quests in Each Category

- Given a 2D array `questDifficultyMatrix = [[5, 7], [8, 6], [9, 7]]` representing the difficulty level of quests, count how many high-difficulty quests (difficulty ≥ 7) there are in each category and log the counts.

12. Sum of Rewards for High-Reward Quests by Category

- For a rewards matrix `questRewardsMatrix = [[50, 200], [300, 175], [120, 500]]`, calculate and log the sum of rewards for each category where the reward is greater than 250.

13. Average Health of Enemies by Type

- With an enemy health matrix
  `enemyHealthMatrix = [[80, 120], [60, 90], [200, 150]]`,
  where each row represents a different enemy type and columns
  represent health in different encounters, calculate the average health
  for each enemy type and log the averages.