Challenge 1

Create the Race Elements

Exam Objectives Covered

- Select the appropriate properties, scripts, and components of GameObjects for required tasks
- Obtain a defined result by using Unity API methods, given Unity's API documentation

Challenge Overview: Implementing Race Mechanics and Flags

Welcome to the next challenge in your SkiFree-inspired game development journey! Now that the game's basic elements are in place, it's time to add features for race mechanics. The goal of the game is for players to complete the ski course as quickly as possible while avoiding obstacles and correctly navigating the course. In this session, you will create a race timer and various types of flags to enhance the racing experience.

Step 1: Create the Race Timer

Implement a race timer that tracks the player's progress through the course.

Instructions:

1. Create Timer Logic:
   - Create a new script, RaceTimer, to manage the timer logic.
   - Implement functionality to start the timer when the player passes through the start flags.
   - Implement functionality to stop the timer when the player passes through the end flags.
   - Add functionality to increase the total time if the player passes flags incorrectly.
   - Ensure the total time is stored and displayed in the console once the race is over.

Step 2: Create Start Flags

Create start flags that trigger the beginning of the race.

Instructions:

1. Create Start Flags:
   - Use prefab assets to create a pair of start flags.
   - Add a trigger collider to these flags to detect when the player passes through.
   - Implement logic in the RaceTimer script to start the race timer when the player passes through the start flags.

Step 3: Create Blue Flags with Pass Conditions

Create blue flags that the player must pass on the left side.

Instructions:

1. Create Blue Flag Prefab:
   - Use prefab assets to create a blue flag.
   - Add a trigger collider to detect when the player passes by.
2. Implement Pass Logic:
   - Write a script, BlueFlag, that determines if the player passes on the correct side.
   - If passed on the left, change the flag color to green.
   - If passed on the right, change the flag color to black and add a one-second penalty to the race timer.
3. Place Blue Flags:
   - Place at least two blue flags in your level in strategic locations.

Step 4: Create Pink Flags with Pass Conditions

Create pink flags that the player must pass on the right side.

Instructions:

1. Create Pink Flag Prefab:
   - Use prefab assets to create a pink flag.
   - Add a trigger collider to detect when the player passes by.
2. Implement Pass Logic:

- Write a script, PinkFlag, that determines if the player passes on the correct side.
- If passed on the right, change the flag color to green.
- If passed on the left, change the flag color to black and add a one-second penalty to the race timer.

3. Place Pink Flags:
    - Place at least two pink flags in your level in strategic locations.

Step 5: Create Finish Flags

Create finish flags that trigger the end of the race and stop the timer.

Instructions:

1. Create Finish Flags:
    - Use prefab assets to create a pair of finish flags.
    - Add a trigger collider to these flags to detect when the player passes through.
    - Implement logic in the RaceTimer script to stop the race timer when the player passes through the finish flags.

Once you've completed these steps, your game will have functional race mechanics that challenge the player to navigate the course correctly and quickly.

Good luck with implementing race mechanics! This exercise will help you master event-driven programming, timer logic, and interactive game elements in Unity. Happy developing!