20.7-Exercise Player Maze Explorer Game

Program

```
using UnityEngine;
using UnityEngine.UI;

public class QuizGame : MonoBehaviour
{
    public Text questionText;
    public Button[] answerButtons;

    private Question[] questions;
    private int currentQuestionIndex = 0;
    private int score = 0;

    void Start()
    {
        // Initialize questions
        questions = new Question[]
        {
            new Question("What is the capital of France?", new string[]{"Paris", "London", "Berlin"}, 0),
            new Question("What is 2 + 2?", new string[]{"3", "4", "5"}, 1),
            // Add more questions here...
        };

        ShowQuestion();
    }
```

```
    void ShowQuestion()
    {
        if (currentQuestionIndex < questions.Length)
        {
            // Display question text
            questionText.text = questions[currentQuestionIndex].questionText;

            // Display answer options
            for (int i = 0; i < answerButtons.Length; i++)
            {
                                                answerButtons[i].GetComponentInChildren<Text>().text    =
questions[currentQuestionIndex].answers[i];
            }
        }
        else
        {
            // Quiz finished
            Debug.Log("Quiz Finished! Score: " + score);
        }
    }

    public void AnswerSelected(int answerIndex)
    {
        if (answerIndex == questions[currentQuestionIndex].correctAnswerIndex)
        {
            // Correct answer
```

```
            score++;
            Debug.Log("Correct!");
        }
        else
        {
            // Incorrect answer
            Debug.Log("Incorrect!");
        }

        // Move to next question
        currentQuestionIndex++;
        ShowQuestion();
    }
}

public class Question
{
    public string questionText;
    public string[] answers;
    public int correctAnswerIndex;

    public Question(string questionText, string[] answers, int correctAnswerIndex)
    {
        this.questionText = questionText;
        this.answers = answers;
        this.correctAnswerIndex = correctAnswerIndex;
    }
```

}

Exercise

Creating a simple Unity Quiz Game. In the game questions are stored in an array of Question objects, each containing the question text, answer options, and the index of the correct answer.

The game has a UI consisting of a text element to display the current question and buttons to display answer options.

The game keeps track of the player's score, incrementing it when the correct answer is selected.

The game progresses through the questions sequentially. After the player selects an answer, it moves to the next question until all questions have been answered.

Basic feedback is provided to the player indicating whether their answer was correct or incorrect.

Once all questions have been answered, the game displays a message indicating that the quiz is finished and shows the player's final score.

Hint

Setting up the Scene:

Create a Canvas GameObject in Unity.

Add UI Text elements to display the question and answer options.

Add UI Buttons for the user to select their answers.

Question Management

Create a C# script named "Question" to represent a single question.

Add fields for the question text, an array of answer options, and the index of the correct answer.

Create an array or list of Question objects in another script to hold all the questions for the quiz.

UI Management

Create a C# script named "QuizUIManager" to handle updating the UI elements.

Use public references to the Text elements for the question and answer options.

Create a method to update the UI with the current question and answer options.

Add event handlers for the UI Buttons to call a method when clicked.

Scoring

Create a variable in the QuizUIManager script to track the player's score.

Increment the score when the correct answer is selected.

Update the UI to display the current score.

Game Flow

Create a C# script named "QuizGameManager" to manage the flow of the quiz.

Use a variable to keep track of the current question index.

Start the game by displaying the first question on the UI.

When a player selects an answer, check if it's correct, update the score, and move to the next question.

Optionally, you can implement a timer to move to the next question after a certain amount of time.

Additional Tips

Use public variables or properties in your scripts to easily reference UI elements and other game objects in the Unity editor.

Make use of Unity's event system to handle button clicks efficiently.

Organize your scripts logically and separate concerns to keep your code clean and maintainable.

Test your game frequently to ensure everything works as expected.

Explanation

This Quiz Game typically involves several key components, including setting up the user interface, managing questions and answers, implementing scoring, and controlling the game flow. In setting up the scene, you would create a Canvas Game Object to hold the UI elements, including Text elements to display the question and answer options, as well as Buttons for user interaction. Question management involves creating a script to represent a single question, containing fields for the question text, an array of answer options, and the index of the correct answer. These questions can be stored in an array or list within another script to hold all the questions for the quiz.

UI management is essential for updating the UI elements based on the current question and answer options. This involves creating a script to handle UI updates dynamically, such as updating the question text and displaying answer options. Additionally, event handlers are necessary for the UI Buttons to trigger actions when clicked, such as selecting an answer.

Scoring is a crucial aspect of a quiz game, as it provides feedback to the player and motivates them to continue playing. Implementing scoring involves keeping track of the player's score, typically through a variable in a script dedicated to managing game logic. The score is incremented when the correct answer is selected, and the UI is updated accordingly to display the current score.

Game flow management controls the progression of the quiz, starting with the first question and moving to the next question after the player selects an answer. This is typically handled by a script responsible for managing the flow of the game, keeping track of the current question index, and updating the UI with the next question. Optionally, a timer can be implemented to move to the next question automatically after a certain amount of time, adding an additional challenge to the game.