CISC 3667: Game Design and Development

In-Class Activity 7: Unity Lab 1 - Getting Started, Sprites, and Movement

Assigned: Thursday, October 22

Submission: This activity continues into Unity Lab 2 on October 29. You will submit your completed project once, by October 29 at 11:59 PM.

Overview

In this lab, you will begin developing a simple 2D game in Unity called **Balloon Popper!** You will become familiar with the Unity Editor, import sprites, create and configure GameObjects, and write C# scripts to control movement.

We will begin the project together in class. You will continue developing the same game during our second Unity lab on October 29.

Part 1: Create and Organize Your Project

1. Create a new 2D Unity project named **BalloonPopper**.
2. Save your gameplay scene as **GameScene**.
3. Create the following folders in the Assets directory:
    - Sprites
    - Scripts
    - Prefabs
    - Scenes
4. Import the provided background, player, balloon, and pin images into the Sprites folder.
5. Examine the Unity Editor, including the Scene View, Game View, Hierarchy, Inspector, and Project Window.

Part 2: Set Up the Game Environment

1. Add the provided background image to your scene.

2. Configure its position and scale so that it fills the camera's view appropriately.
3. Adjust its Sprite Renderer sorting settings so that it appears behind the gameplay objects.
4. Add the player sprite to the scene.
5. Position the player near the bottom of the screen, above the ground.
6. Adjust the camera and Game View so that the background and player are visible.

Your scene should have a recognizable game environment with a visible player character.

Part 3: Implement Player Movement

1. Create a C# script named **PlayerMovement**.
2. Attach the script to your Player GameObject.
3. Implement keyboard input so that the player can move **left and right** using the arrow keys or A and D.
4. Create a movement speed variable that can be adjusted in the Inspector.
5. Use Time.deltaTime to keep movement consistent across different frame rates.
6. Prevent the player from moving beyond the left and right boundaries of the visible play area.

Your player should respond smoothly to keyboard input and remain within the gameplay area.

Part 4: Create a Moving Balloon

1. Add the balloon sprite to your scene.
2. Position it above the player.
3. Create a C# script named **BalloonMovement**.
4. Make the balloon move automatically across the screen without player input.
5. Reverse its movement direction when it reaches the left or right boundary.
6. Give the balloon its own adjustable movement speed.

The balloon should move continuously while the player can move independently below it.

Part 5: Prepare for the Next Lab

1. Create a Pin GameObject using the provided pin sprite.
2. Adjust its appearance and scale so that it is appropriately sized for the game.
3. Create a Prefab from the Pin GameObject and save it in the Prefabs folder.
4. Remove the test Pin instance from the scene after creating the Prefab.
5. Save your scene and project.

You do not need to implement shooting or collision detection yet. We will introduce these mechanics during Unity Lab 2.

Expected Progress by the End of Lab 1

By the end of this session, your project should contain:

- A configured 2D Unity scene with a background.
- A player that moves horizontally using keyboard input.
- A balloon that moves automatically and reverses direction at the screen boundaries.
- A Pin Prefab prepared for the next lab.
- Organized project folders and C# scripts.

Important: You do not need to submit anything today. Keep your Unity project and all associated files. You will continue working on this project on October 29 and submit the completed work after Unity Lab 2.