Adding Visual Effects (VFX) to Your Game

Objective:

Students will learn how to add visual effects (VFX) to their Unity game to simulate events such as explosions and pickups. By the end of the lesson, students will implement VFX correctly within their game.

Step 1: Understanding Visual Effects (VFX)

What are VFX?

- VFX are special effects that simulate small objects like smoke, fire, and dust.
- They make the game more visually appealing and interactive.

What will we do?

- Add an explosion effect when an enemy catches the player.
- Add a pickup effect when the player collects a token.

Expected Outcome:

- Students will see their game visually enhanced with effects.

Step 2: Requirements

Before starting, ensure the following:

- A Unity project is set up.
- Basic game elements (player, enemy, token) are present.
- A Game Design Document (GDD) has been created to plan effects.

Expected Outcome:

- Students confirm their game setup is ready for adding VFX.

Step 3: Reviewing the Game Design Document (GDD)

What is the GDD?

- A document that outlines the theme, mechanics, and effects used in the game.

What to do?

- Open the GDD and identify where the VFX will be used.
- Ensure the effects match the game's theme.

Expected Outcome:

- Students understand how VFX fit into their game design.

Step 4: Reviewing the Asset Pack

What are Assets?

- Pre-made elements (like VFX) that can be used in the game.

Steps:

1. Open the Unity project.
2. Locate the **Prefabs > VFX** folder in the Asset Pack.
3. Drag sample particle systems into the scene to preview.

Expected Outcome:

- Students become familiar with available VFX assets.

Step 5: Adding an Explosion Effect

Purpose:

- Show an explosion when the enemy catches the player.

Steps:

1. Open the **PlayerController** script.
2. Add this code inside the `OnCollisionEnter` function:

```
Instantiate(explosionFX, transform.position,
Quaternion.identity);
```

3. Ensure a public GameObject variable `explosionFX` is assigned.

4. Set the **particle system to play on awake**.

Expected Outcome:

- When the player collides with an enemy, an explosion effect appears.

Step 6: Adding a Pickup Effect

Purpose:

- Show a special effect when the player collects a token.

Steps:

1. Open the **PlayerController** script.
2. Add this code inside the `OnTriggerEnter` function:

```
var currentPickupFX = Instantiate(pickupFX,
other.transform.position, Quaternion.identity);
```

3. Remove the effect after it plays:

```
Destroy(currentPickupFX, 3);
```

4. Set the **Particle Simulation space to Local**.

Expected Outcome:

- When the player picks up a token, an effect plays and disappears after 3 seconds.

Step 7: Additional Challenges

Try these optional enhancements:

1. **Easy:** Add a celebration effect when the player wins.
2. **Medium:** Add trailing smoke/dust behind the player as they move.

Expected Outcome:

- Students can further enhance their game's visuals if they choose.

Step 8: Next Steps

- You have successfully added VFX to your game!
- In the next lesson, you will learn how to add a user interface (UI) to your game.

Deliverable:

- Export your Unity project as a **WebGL** build.
- Share the WebGL link to showcase your VFX enhancements.

Congratulations! You have successfully added VFX to your game. 🎮