CS 4730: Computer Game Design

SFX and Level Team: Scenes and Parsing

Overview

For this assignment, you will write one class (Scene) and integrate into your code base. A Scene is a single set of Sprites rooted at the Scene object (DisplayObjectContainer) that represents a single room or area of your game. Your primary task is to write a method that loads the scene from a file and constructs the Display Tree from it. You'll need to keep in mind complications your game's specific design might provide down the road.

PART 1: Scene Class

The Scene header file provided extends DisplayObjectContainer and contains one method (loadScene). This method reads a text file describing a scene of the game in detail and constructs the display tree that is described. The root of this display tree is this Scene object.

Your first task is to write this class and this loadScene() method. In order to do this, you'll need a file format for your scenes. You may define whatever format you'd like, but it MUST be able to do the following:

- Describe all sprites in the game, including all fields in DisplayObject, DisplayObjectContainer, AnimatedSprite, and Sprite. This means the file can describe an object and its x,y location, its scale, rotation, image file path, etc.
- You also need to be able to describe AnimatedSprites, including the names of all animations, the names of the image files for each, etc. Keep in mind that the engine team will be support for Spritesheets later, so you'll want to think about how to work that into this code eventually.
- Also keep in mind that your game will have very specific types of Sprites eventually. Maybe you'll have ice blocks or enemies. You won't be able to implement these things yet, but your format needs to be flexible enough that you can add them when necessary. Several future assignments will say "update your scene parser to handle these new features".

Part 2: Integrate with the rest of the code

Integrate this new Scene class into the rest of your code. Your game should contain a field that is the current scene. The scene should be settable and will be added as a child of the game when set.

Part 3: A Basic Test

Add a scenes folder under the resources folder of your project. Create two text files with two different demo scenes. Write a small demo program that loads one of the scenes from the file. When the "P" button is pressed, the other scene is loaded. If "P" is pressed again, the first scene reloads, etc.

Turn In

As always, submit your code on Collab as a single zip file.