Lab 03 – Game Structure (Part 2)

Objectives

1. Explore GDevelop more by solving additional problems.

Practices

The following problems continue from Lab02 in which we work on the Duck Shooter game using GDevelop.

Problem 1: [Medium]

Create a new instance of **TargetColored** object and place it somewhere on the right of the scene. Unlike the old **TargetColored** instance (done in Lab02, Problem 5), we want the new instance to move up and down. What should we do to achieve this ?

Sample Output

Check this video.

Hints:

1. Behavior Video / Behavior Documentation
2. Tween Behavior
3. At the beginning of the scene (condition)

Problem 2: [Medium]

Try shooting both of the **TargetColored** instance. You will notice that all **Shot** instance only attach to the first **TargetColored** instance that is placed in the Scene.

This happens because GDevelop does not know which **TargetColored** instance to associate the **Shot** instance to. To solve this problem, we need to link every **Shot** instance to the correct **TargetColored** instance the moment the **Shot** is created.

Add the necessary events related to linked objects, such that upon previewing, the output is like the **Sample Output** below.

Sample Output

Check this video.

Hints

1. Add Sticker Extension into the project.
2. Object Behaviors.
3. Stick object to another project.

Problem 3: [Easy]

Create a **Sprite** object and name it as **"StickMetal".** Set its first animation to **"Assets/Objects/stick_metal_outline.png"**.

Update the events in the Event Editor such that for each instance of **TargetColored** object, a new instance **StickMetal** object is created, is placed below the **TargetColored** object, and moves together with its associated **TargetColored** object. Upon previewing, the output should look like the Sample Output below.

Sample Output

Check this video.

Hints

1. Linked Objects.
2. Edit points.

Problem 4: [Medium]

In this problem, we will add ducks into the game as shooting targets, just like the **TargetColored** object. We begin by creating a **Sprite** object and name it as **"Duck Yellow"** and followed by setting its first animation to **"Assets/Objects/duck_outline_yellow.png"**.

After the above step, we simply apply what we did to **TargetColored** object to the **DuckYellow** object (follow from Lab02`s Problem 5 to Lab03`s Problem 3), which includes placing a **DuckYellow** object into the scene, editting its object and instace variables, and adding a Tween behavior to move the **DuckYellow** object.

We would be adding more than one **DuckYellow** objects into the scene, so we want the **Shots** object to be linked with the correct **DuckYellow** object when it is shot. We also want to attach a **StickMetal** object below each **DuckYellow** object so that the **StickMetal** object will follow its corresponding **DuckYellow** object wherever it goes. Therefore, we need to apply what we did from Problem 1 to Problem 3 just now to the **DuckYellow** object.

All the above involves many copy-and-paste tasks, so we need to be very careful and need to test one step at a time.

[**Optional:** You may introduce new movement pattern to **DuckYellow** so that it move differently from the **TargetColored** object. Check out the Sample Output for an example]

Sample Output

Check this video.

Hints

1. Review what you have done for **TargetColored** object.

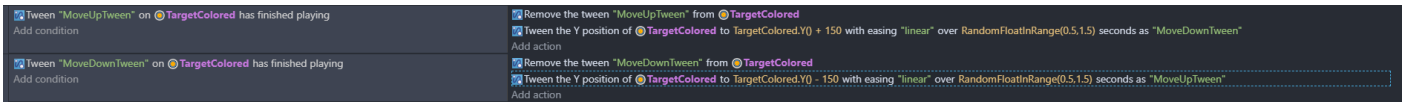
Problem 5: [Easy]

Except for different animation, the **TargetColored** object and **DuckYellow** object have the same variables and events. This results in duplicating events written in the Event Editor which would affect the maintainability and readability of the events.

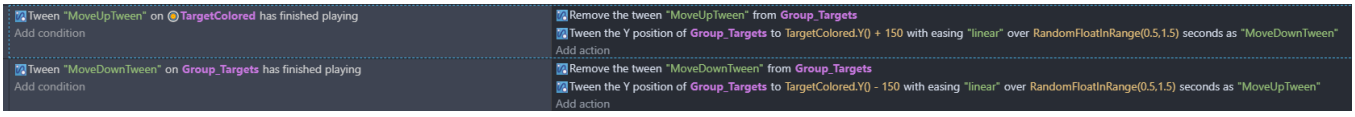
Thankfully, we can solve this problem by using **Object Group.** The object group allows us to group objects so that the objects in that group can be referred by a single name in Events Editor.

Open the **Object Group Editor** by clicking on this icon  at the top right panel of GDevelop. This would open up the Object Groups panel on the right side. Then, add a group named ”Group_Targets” and add **TargetColored** and **DuckYellow** object into the group.

Now, go to the Events Editor and look for events that applies to **TargetColored** and **DuckYellow** objects. For example:



Then, replace these with one event that applies to Group_Targets object group instead. For example :



Sample Output

Similar to Problem 4`s Sample Output.

Hints

1. Organizing Objects into Groups

Acknowledgement

- The assets for the problems are obtained from Kenney's Shooting Gallery Pack.
- Original Lab Lesson prepared by Ban Kar Weng (William).
- Modified by ZulHisham Rozami.