Lesson 5.1: Flying Dragon - Part 1

Objectives

In this lesson, students will:
- ❖ Practice building a program by experimenting and iterating.
- ❖ Learn programmatic animation techniques.
- ❖ Practice using programmatic constructs such as conditionals, loops, comments, random numbers, testing and code-reuse.

Agenda

1. Getting Started	15 mins
2. Student Activity: Adding Treasure	25 mins
3. Wrap Up and Reflections	10 mins

Preparation

- ❏ Projector for code explanations
- ❏ Print student activity worksheet (one per student pair)

Resources & Links

- ❏ Flying Dragon starter project: https://scratch.mit.edu/projects/282849422/

- ❏ Flying Dragon solution project: https://scratch.mit.edu/projects/282848597

1. Overview

Students will code a flying dragon game during the student activity.   Students are given a starter project to which they will iteratively add additional features to the game over the course of 2 lessons. In this lesson students will code most of the basic games.
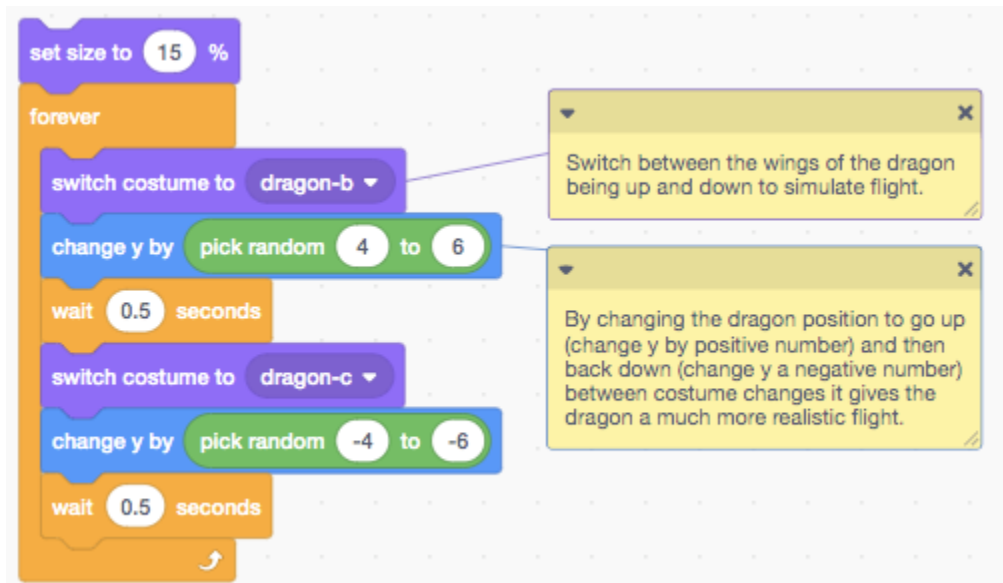
2. Getting Started

Explain how the lesson will take place before you get started.

Demonstrate and explain the flying dragon game (the project page explains how to play and score points during the game) by using the solution project:
https://scratch.mit.edu/projects/282848597

Once students understand what they will be creating, **open** the starter project and **explain** the scripts for the clouds and dragon sprites line by line. You can prompt students to try to explain sections of the code as to what it does.

Script to animate the dragon Sprite:


```
set size to 15 %
forever
    switch costume to dragon-b ▼          Switch between the wings of the dragon
    change y by (pick random 4 to 6)      being up and down to simulate flight.
    wait 0.5 seconds
    switch costume to dragon-c ▼          By changing the dragon position to go up
    change y by (pick random -4 to -6)    (change y by positive number) and then
    wait 0.5 seconds                      back down (change y a negative number)
                                          between costume changes it gives the
                                          dragon a much more realistic flight.
```

3. Student Activity: Adding Treasures

It is recommended that students work in pairs for this activity.   If students are not able to finish this activity, there will be some time during the next lesson to give them a chance to finish.

Distribute the student activity worksheet, one per student pair. **Explain** the activity.
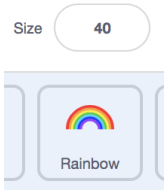
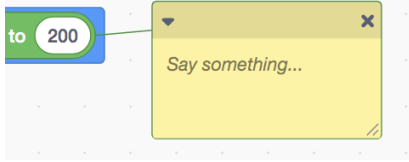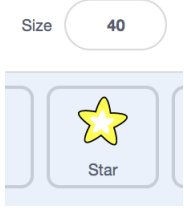### 3. Wrap Up and Reflections

📐 Reflection Points:
● What did you learn today? ● Why do we want the sprites (other than the dragon) moving towards the left of the stage? ● Which code for the dragon sprite helps it look like it is flying? ● What was challenging about today's activity?

Student Activity:Adding Treasures

Take turns coding with your partner. The navigator can help read the instructions.

If you are stuck, you can: Discuss with your partner
                               Ask another team
                               Ask the teacher a question

What to do:	Using/Details:
Remix and **save** the Flying Dragon starter project	[282849422](#)
Add a **rainbow** sprite and set its size to 40%. The rainbow glides across the screen very much like the red walls. So to save time, start by copying the code from the red wall to the rainbow sprite.	Size 40 Rainbow
There are some differences between the red wall and the rainbow sprite. Let us address each one at a time: ● The rainbow only has one costume, so delete the code to change the costume ● We only want the rainbow to appear every so often. At the beginning we want to hide the rainbow and then, inside the forever-loop wait a few seconds before the rainbow shows itself.	
Test your code for at least 20 seconds. Notice how the rainbow always shows up at the same location on the right? We want the rainbow to show up at random locations on the right. You can do this using the **set y** block. **Use a random number between the top and bottom of the screen size.**	set y to 0 pick random to

Add a comment to explain the code you just added. **Test your code again.** That's better, the rainbow appears at different locations. If it does not, keep exploring.	
Now add the **star** sprite for the star treasure. You can copy the code from the rainbow, but change 2 things in the script: 1) Have the star move faster across the stage 2) Change the wait time so that the rainbow and star don't show up at the same time.	Size 40
Test your code to make sure the rainbow and star are working.	