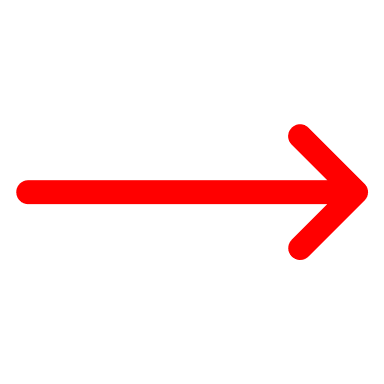
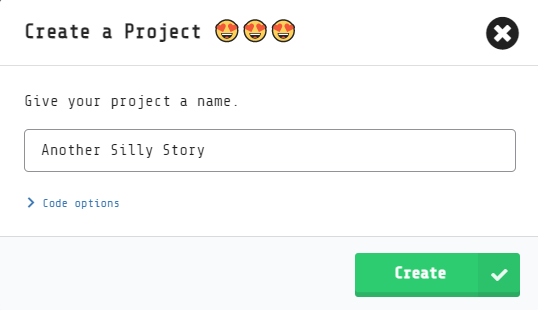
# Lab 1.3: Another silly story time!

## Overview

Recall from your time in Blocks that you created a silly story project where you prompted the player for different values and then created a story out of that input. In this lab, you will create another silly story and turn it into a project in MakeCode Arcade using either JavaScript or Python.

## Creating a language-specific project in MakeCode Arcade

Before we begin this activity, let’s create a project that keeps you in the appropriate programming language.

1. Start creating a project in the usual way. Give your project an appropriate name. Then, select **Code options**.   
   
2. The default is to allow all languages. Instead, select the language that you will use for your project:

Creating a project in MakeCode Arcade. The "code options" has been expanded to show the options available:

Blocks, JavaScript, and Python.
Python only.
JavaScript only.

1. After selecting your language, select **Create**.

For simple projects, moving back and forth among the different languages is fine. When your projects get more complex, though, switching to different languages in your project can make a mess. So, get in the habit of creating your projects exclusively for your programming language.

You still can switch back and forth between Blocks and the other languages! It’s a great way to remember how to do something that you’ve learned in Blocks or to help you visualize the syntax in your programming language better. When you want to do that, though, create a new project. (Perhaps use the word *playground* or *sandbox* in your new project’s name.) Build a small program in Blocks, switch between the different languages to learn what you need, and then return to your main project to apply what you learned.

## Part 1: Basic Story

1. Write a brief story. It does not need to be very long at first; three or four sentences is a good start. You can expand the story later.
2. To create a silly version of your story, remove some of the words, replacing them with blanks. Decide what prompt you will use for each blank. You can ask for a part of speech, or you might ask for something a little more specific.
3. Create a MakeCode Arcade project. You can name it something like **Another Silly Story**. You could name your project after the topic for your story. You also could name it something like **Lab 1.3**.
4. Create a separate variable for each blank in your story.
5. Ask the player to provide values for each variable. Use the **ask for number** or **ask for string** block as needed.
6. Use **splash** blocks with **join** blocks to tell the silly version of your story.

## Part 2: Enhancements

Now, it’s time to get even more creative with your project! Consider these ideas to enhance your project. Feel free to come up with your own ideas, too!

1. Illustrate your story with a background and sprites.
2. Is there a hero sprite and, if so, will the player be in control of it? Will any sprites move automatically?
3. Expand your story. Perhaps include multiple scenes.
4. Add sound effects to your story.
5. Add background music to your story. Explore the **play melody** block found in the **Music** drawer of the toolbox.

## Reflection

1. Congratulations! You have created your first big project in a typing language! Now, review the project that you created for this lab. Can you identify the different types of variables that you have created in MakeCode Arcade?
2. What is the syntax for creating the different types of variables that you have used?
3. What is the syntax for changing the value of a variable after you have created it?
4. What skills from the Blocks environment are you looking forward to learning in your typing language the most? Why?

## Rubric

|  |  |
| --- | --- |
| Lab 1.3 criteria | Point values |
| Part 1 | (4 total points) |
| Project contains a basic story | 2 points |
| Player provides values for variables | 1 point |
| Program uses variables when presenting story | 1 point |
| Part 2 | (2 total points) |
| Story is presented in a creative way | 2 points or more |
| Project total | 6 total points |