Introducing Scratch

8 – Flappy Bird

The last few games in this series of exercises will all make use of some of the more advanced features of Scratch to create more complex (and more fun) games. For this one we will create a version of the popular Flappy Bird game which includes some modifications.

Like the normal game, we will have a bird that flies along and keeps flying by pressing a key (the space bar in our game). The bird needs to avoid touching the ground, while avoiding obstacles that move toward it.

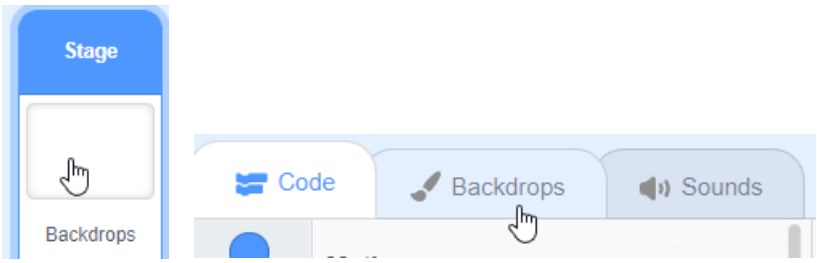
Exercise 1. Create the Background

We'll create a simple background that will have three versions. As a player advances further in the game, the background will change to give the feeling of advancing to a new level. Some parts of the game will also change to make it more difficult at each change in "level".

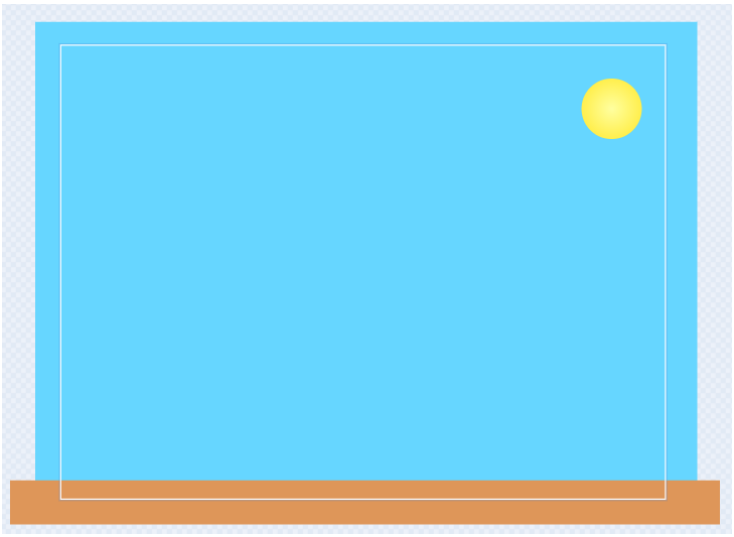
Each version of the background will have a strip of "ground" along the bottom that the player needs to avoid. This will be done by using the colour of the ground.

1. Start a new project and remove the default cat sprite.

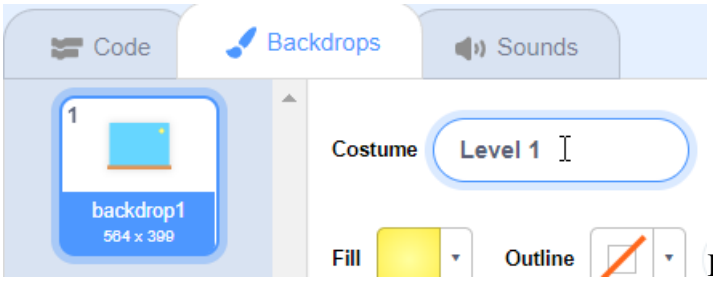
2. Make sure the backdrop is selected from the side of the sprites and then select the Backdrops tab to modify the backdrop.



3. Create a simple background which includes a strip of ground along the bottom. The colour of the ground will need to be the same the whole way along the bottom since the colour will be used to detect if the player has touched the ground. You can include additional decorations on the background but try to ensure that they won't make it difficult to see things that are important to the game (you can always come back and modify the backdrop once the rest of the game is finished).

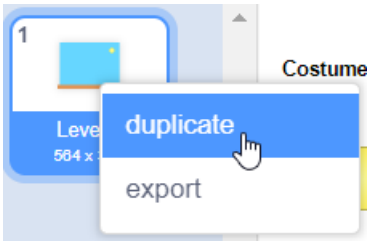


4. Change the name of the backdrop to **Level 1**.



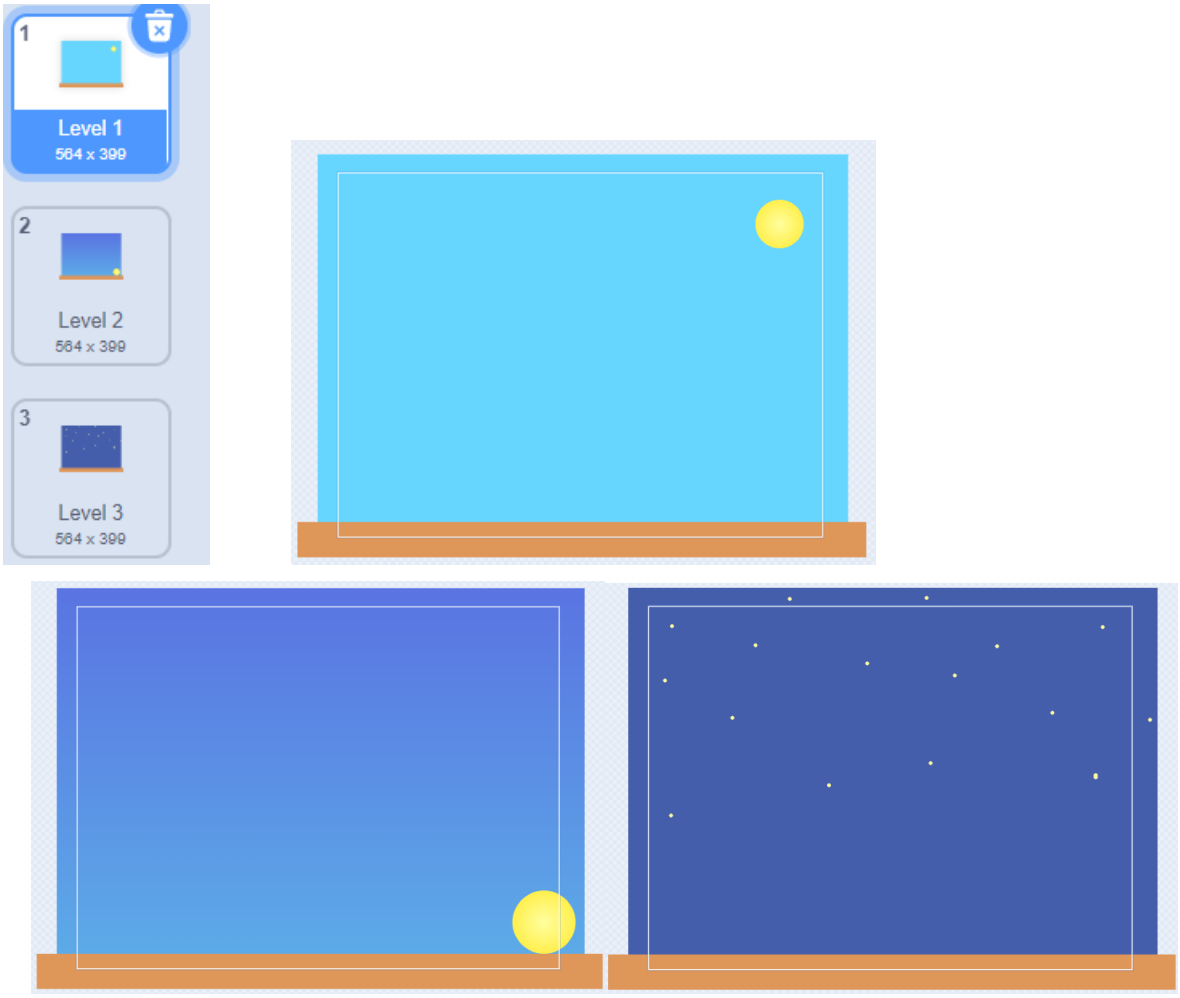
Right

5. Right click on your backdrop and duplicate it.



6. Modify your new backdrop slightly and name it **Level** 2 (scratch might automatically rename it for you.
7. Create a third backdrop and name it **Level 3**.

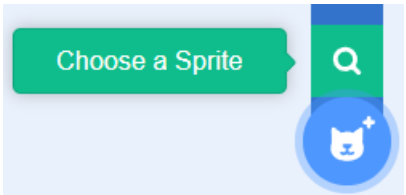
You should now have 3 different backdrops.



Later on, we will add some code for the backdrop but for now, we will create the bird.

Exercise 2. Creating the Bird

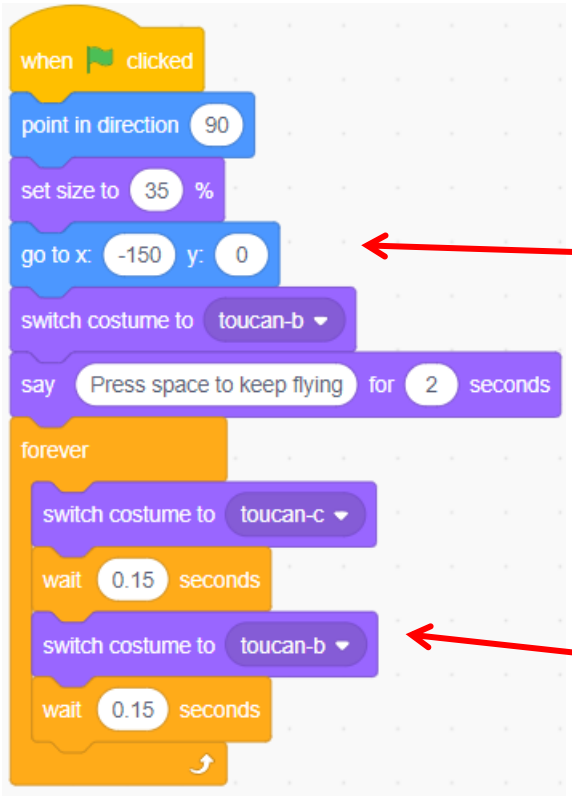
1. Choose a new sprite.



2. From the **Animals** category select the **Toucan** sprite. This one will be a good choice for our game since it includes 2 costumes for the flapping animation and a third costume that we will use when the bird crashes.

3. Rename the sprite to **Bird**.

4. Add the code blocks below and then click the **Go** button to test it.



Set starting conditions for the sprite

Flapping wings animation

Exercise 3. Adding the Lose Conditions

1. Add the following code blocks.

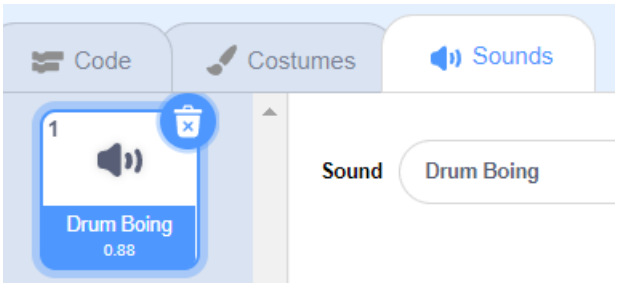
Wait while the bird says the instructions from the previous code block

Create gravity effect by constantly moving downward

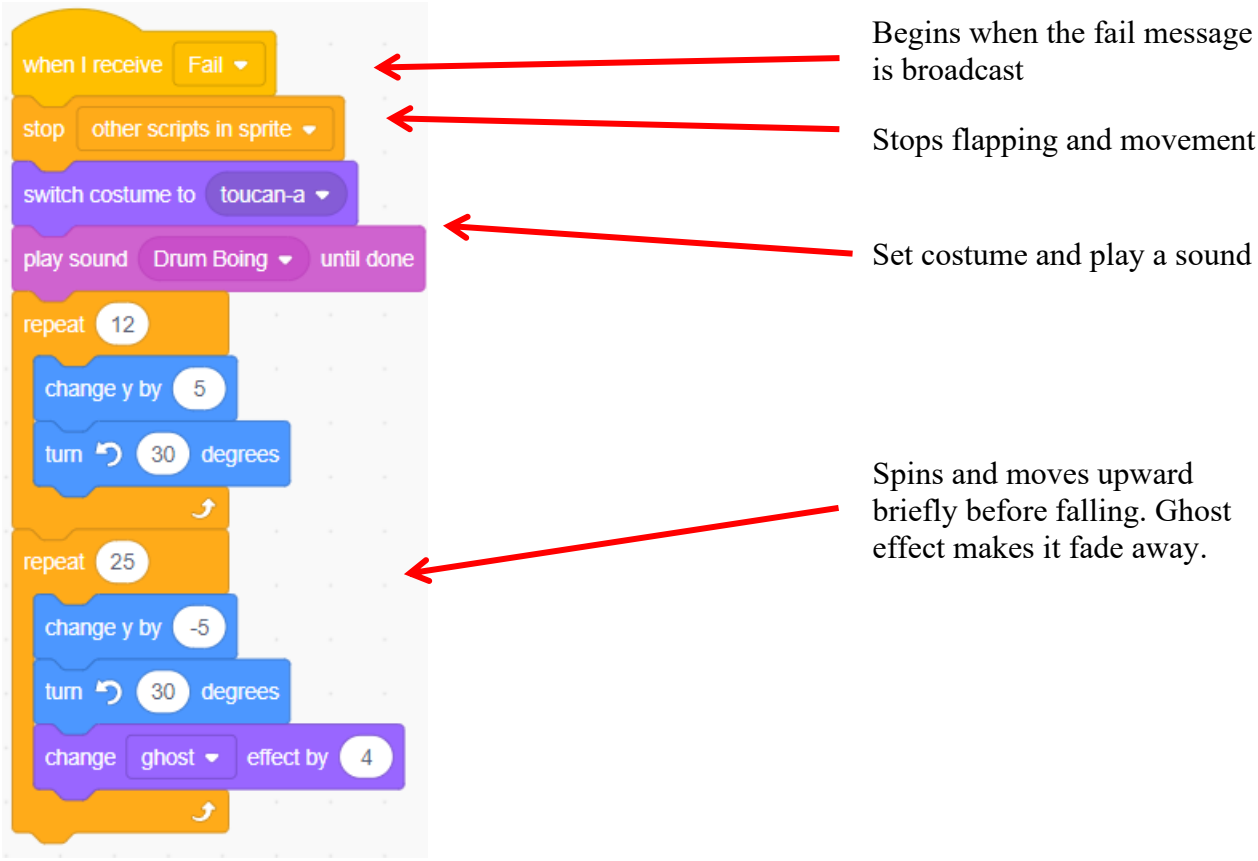
Move upward each time the space bar is pressed

Broadcast when the colour along the bottom of the backdrop is touched. This broadcast will be used to trigger the end of the game.

2. From the sounds tab, import a sound that would be suitable for when the bird crashes on the ground or collides with an obstacle.

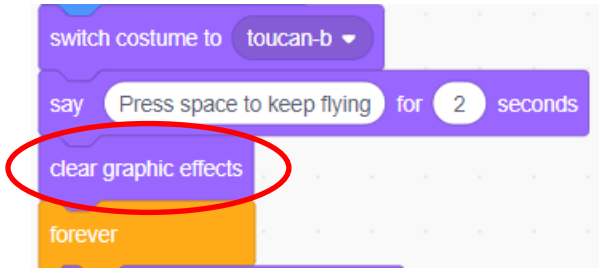


3. Add the following code blocks.



Since we've used a graphic effect to make the sprite fade away, we will need to make sure graphic effects are reset when a game starts so the bird is not still in "ghost" mode.

4. Add a **Clear Graphic Effects** block to the first starting code group.



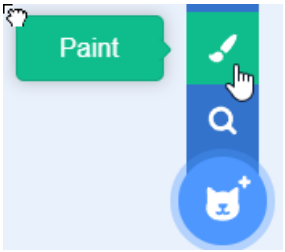
5. Test the game so far. Test the space bar to make sure it keeps the bird flaying and test what happens when it touches the ground.

Tip	This one **Clear Graphic Effects** block can reset all graphics effects for a sprite, including changes to colour or brightness.

Exercise 4. Extra Decorations

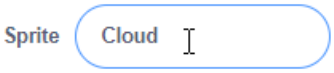
We're going to add to the illusion of movement by having clouds move across the stage while the game is playing.

1. Create a new sprite

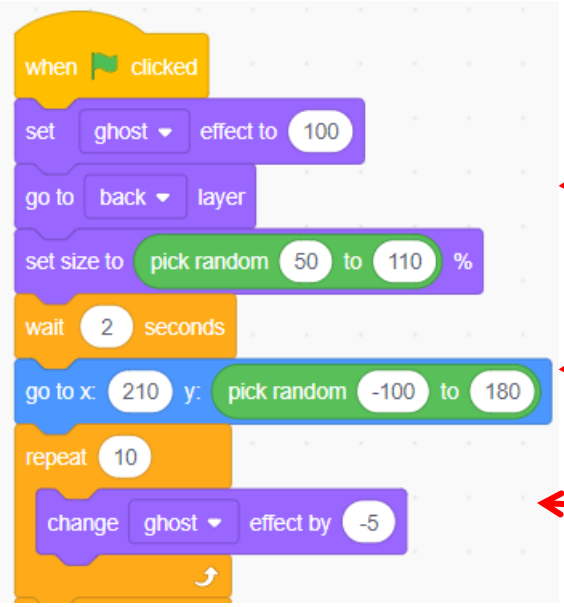


2. Draw a cloud for the new sprite (this might be easier if you convert to bitmap first).

3. Rename the new sprite.



4. Add the following code blocks for the cloud sprite.

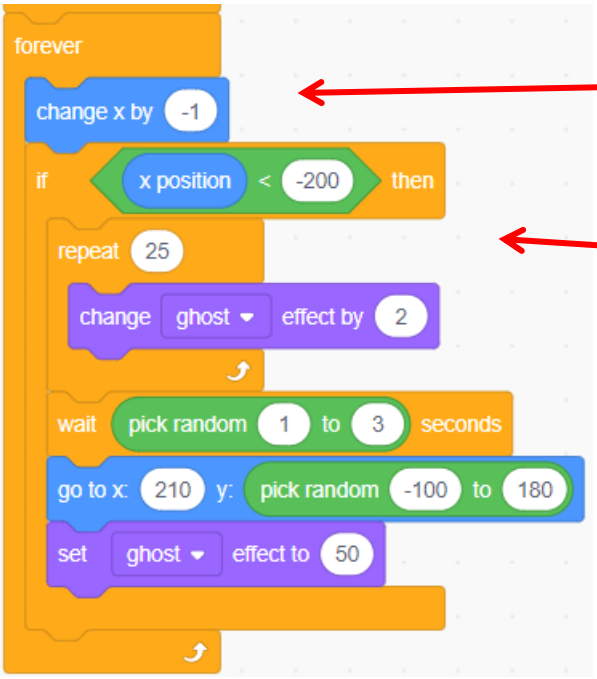


Set cloud to start as transparent. Back layer so it appears behind other objects. Random size so each cloud is slightly different

Start on the right with random vertical position.

Uses a loop to "undo" the ghost effect so that it gradually appears until it is 50% transparent

(continued next page)

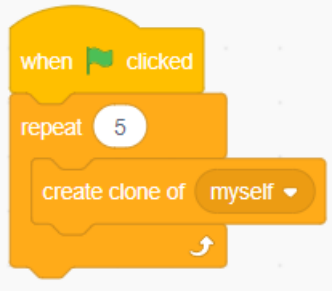


Slowly moves to the left

When it reaches the left side, fades out and re-appears on the right side

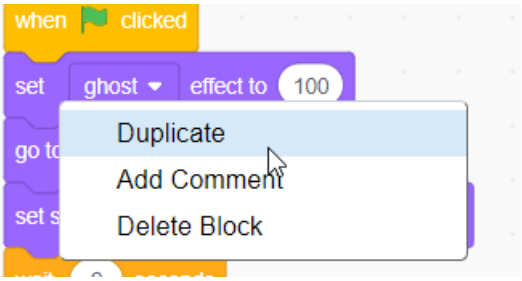
We will add some blocks to clone the cloud so that it clones itself and creates several clouds to scroll across the background.

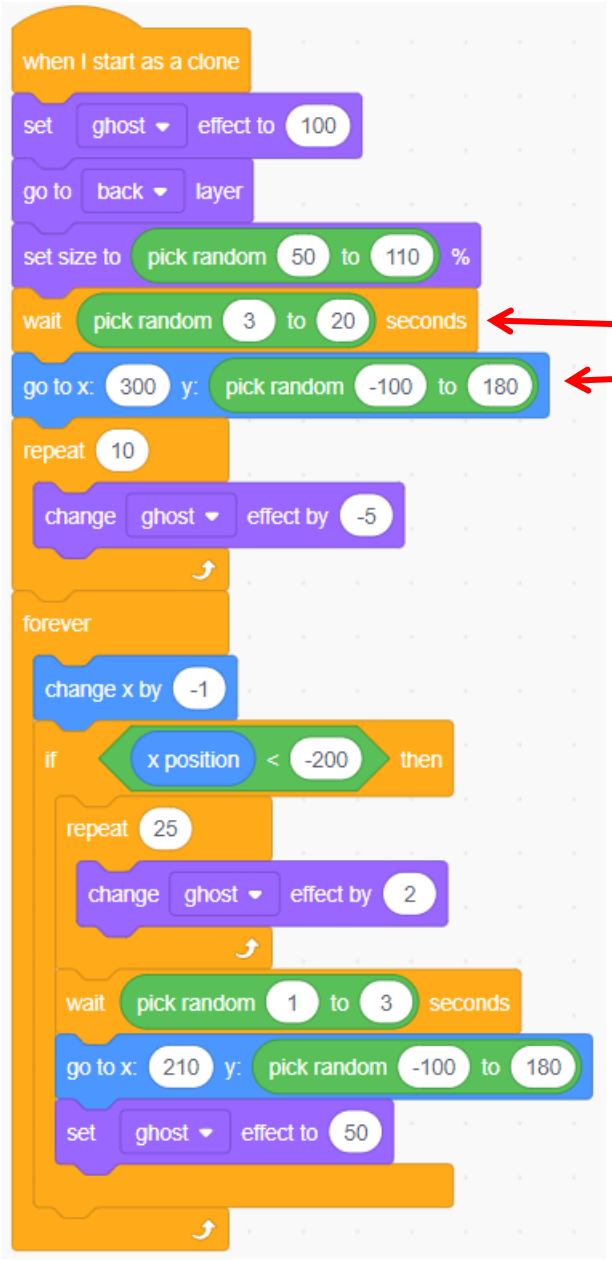
8. Add the following blocks to the cloud sprite.



9. Add the blocks on the following page to tell each clone what to do.

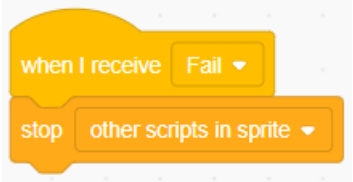
Tip Since most of this is similar to the code block, you can duplicate most of it and then change the parts that are different.





Each clone waits a random number of seconds before beginning and starts a bit further over than the first cloud

10. Lastly, we will add some blocks that pause the movement when the bird crashes.



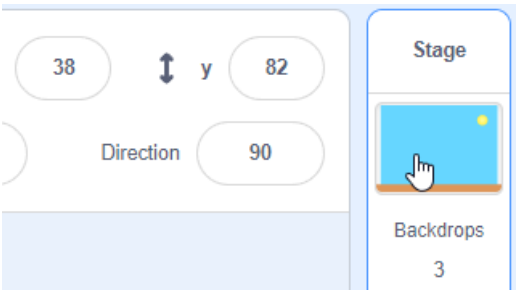
11. Test your game  now that you have moving clouds in the background.

Exercise 5. Adding Levels

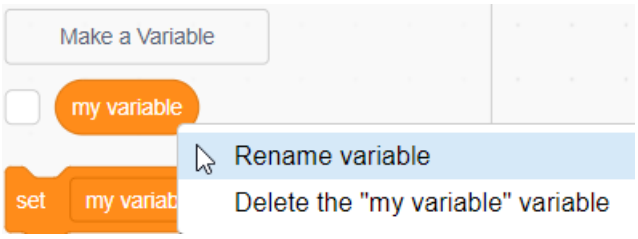
Our game will have 3 levels. The second level and third level will both begin once a certain amount of time has passed. As each level begins, the background will change and a broadcast will be put out so that changes can be made to our obstacles which we will create later.

The variable that will be used to determine when each level will change will also be used for the player's score.

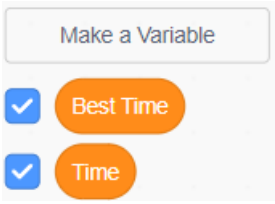
1. Select the **Stage**.



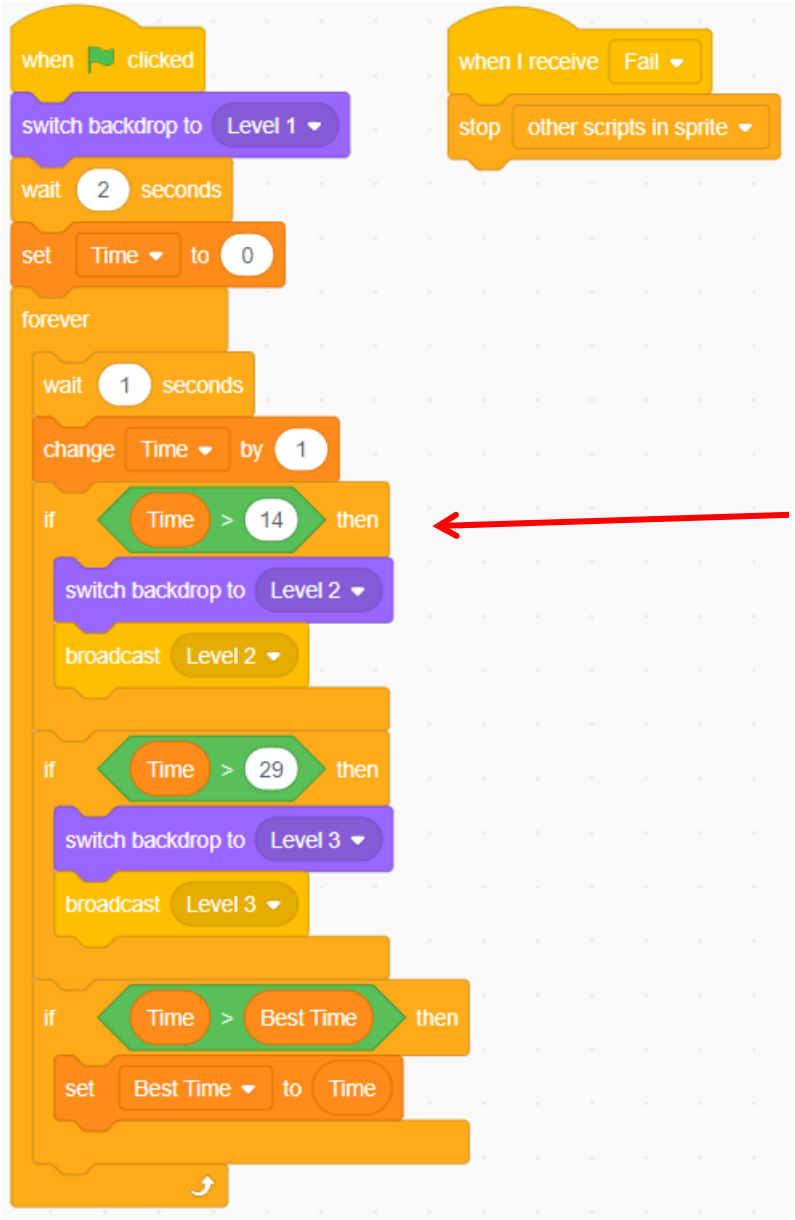
2. Create new variables for **Time** and **Best Time**.



Tip You can delete the default "My Variable" if you want, or rename it to use it as one of your new variables.

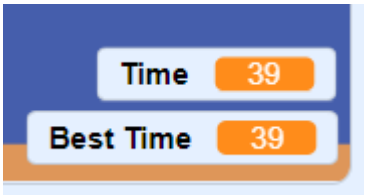


3. Add the following script blocks.



14 seconds plus the 1 second wait when the loop starts adds up to 15 seconds before the level changes

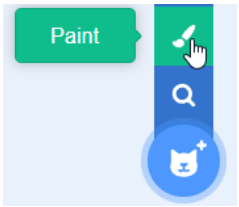
4. Test it  to make sure the backgrounds are changing after 15 and 30 seconds and to make sure the Time and Best time are working. It's pretty easy when there is nothing but the ground to avoid but now, we're going to add some obstacles to avoid.
5. Reposition the Time and Best time so that they are out of the way in the bottom corner.



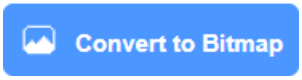
Exercise 6. Adding the Obstacles

The first obstacle we will add is a **Tree** that will move along the bottom part of the screen. Each time it appears the height will be random so that sometimes it will be taller and create a bigger hazard than other times. We will also create a variable for the speed it moves so that it can move faster with each level.

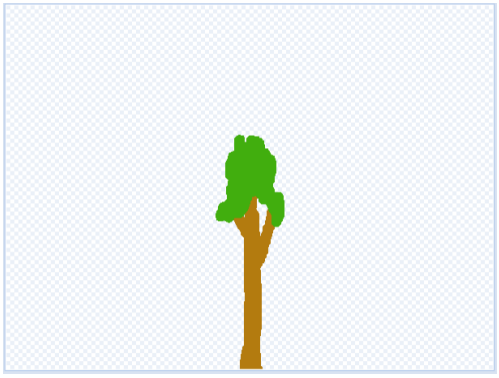
1. Create a new sprite.



2. Switch to Bitmap mode.

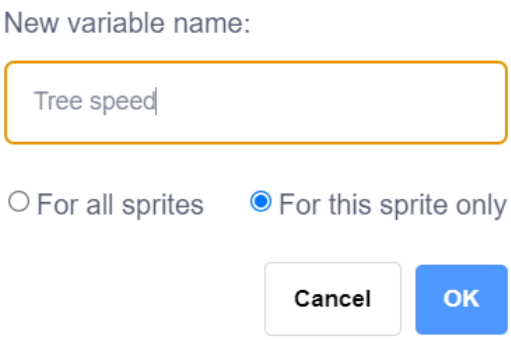


3. Draw a narrow tree.

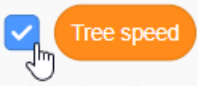


4. Rename the sprite as **Tree**.

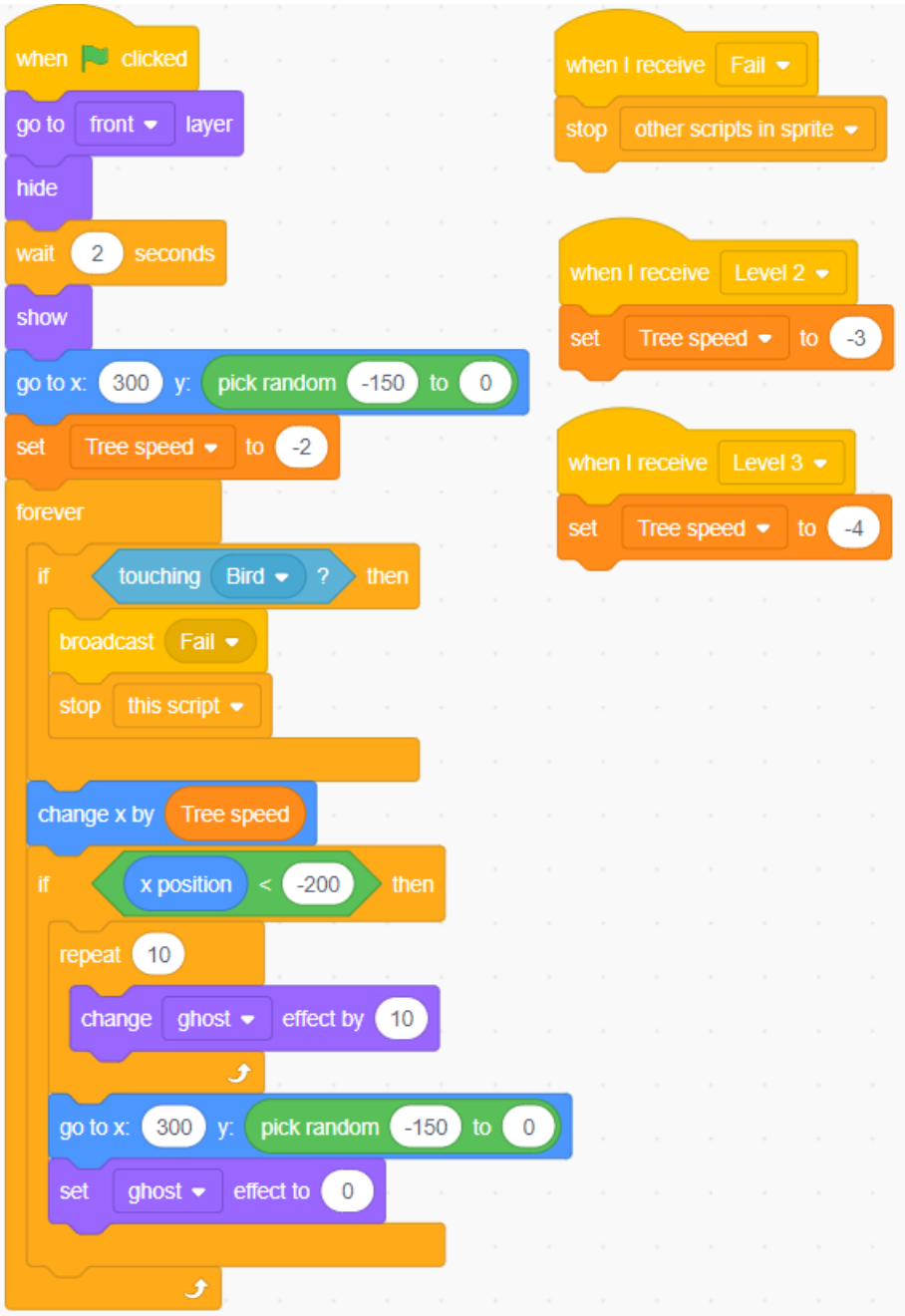
5. Make a variable called **Tree Speed**.



6. Untick the new variable in the blocks section so that it is not showing on the stage.

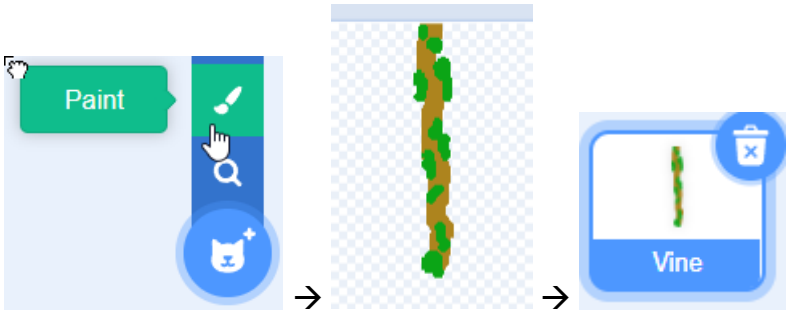


7. Add the following code blocks.



When Level 2 begins, a **Vine** will appear hanging from the top. Like the tree, it will appear randomly at different heights. It will also get faster when Level 3 begins.

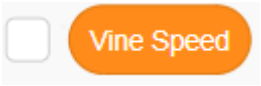
8. Create a new sprite for your vine.



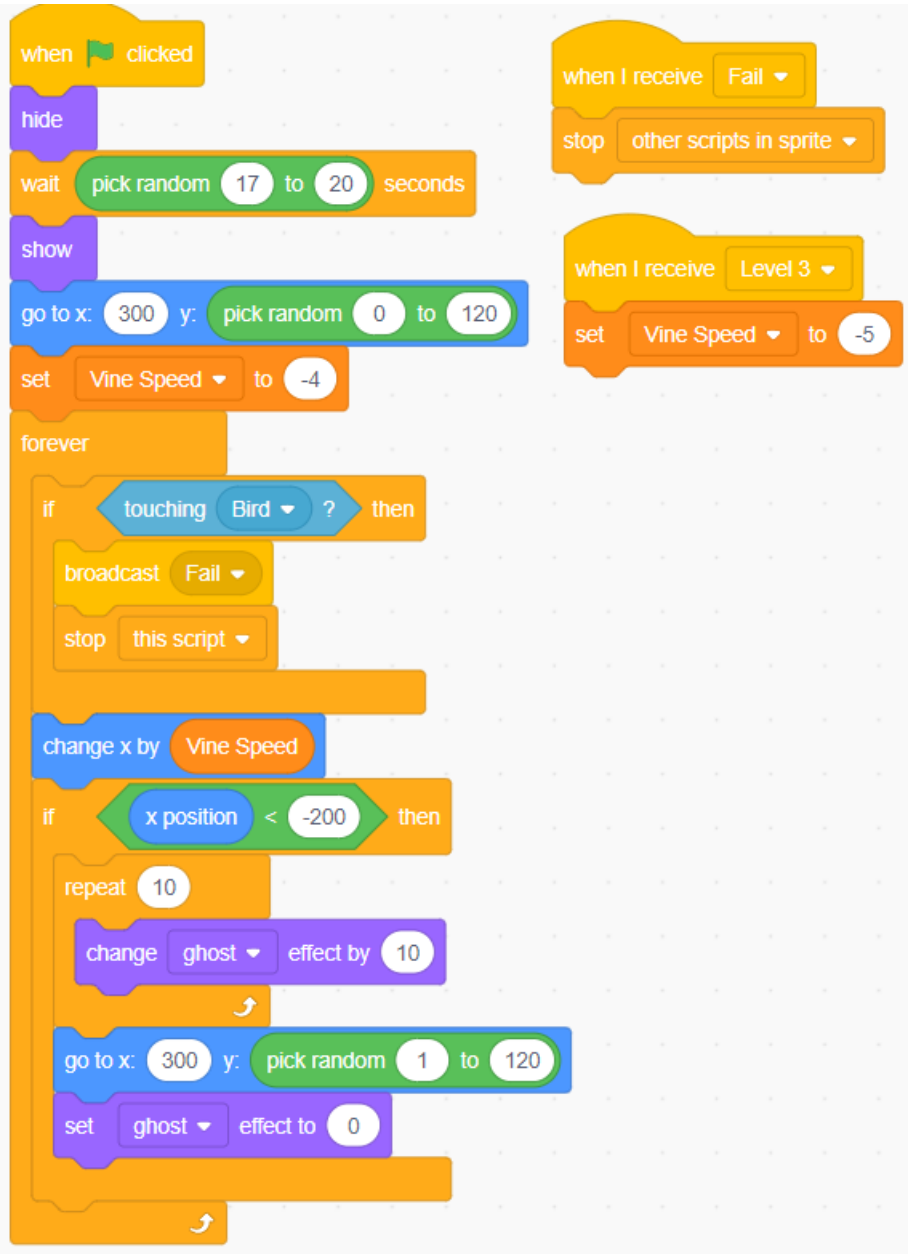
Add the following code blocks.

Tip	When you have code blocks in different sprites that are similar, you can drag code blocks from one sprite on to another sprite to copy them. Then you only need to change the parts that are different.

9. In the **Vine** sprite, create a variable called **Vine Speed**.

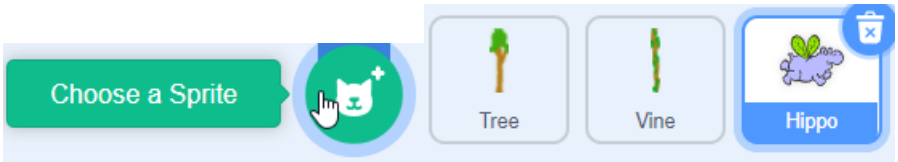


10. Add the following code blocks.

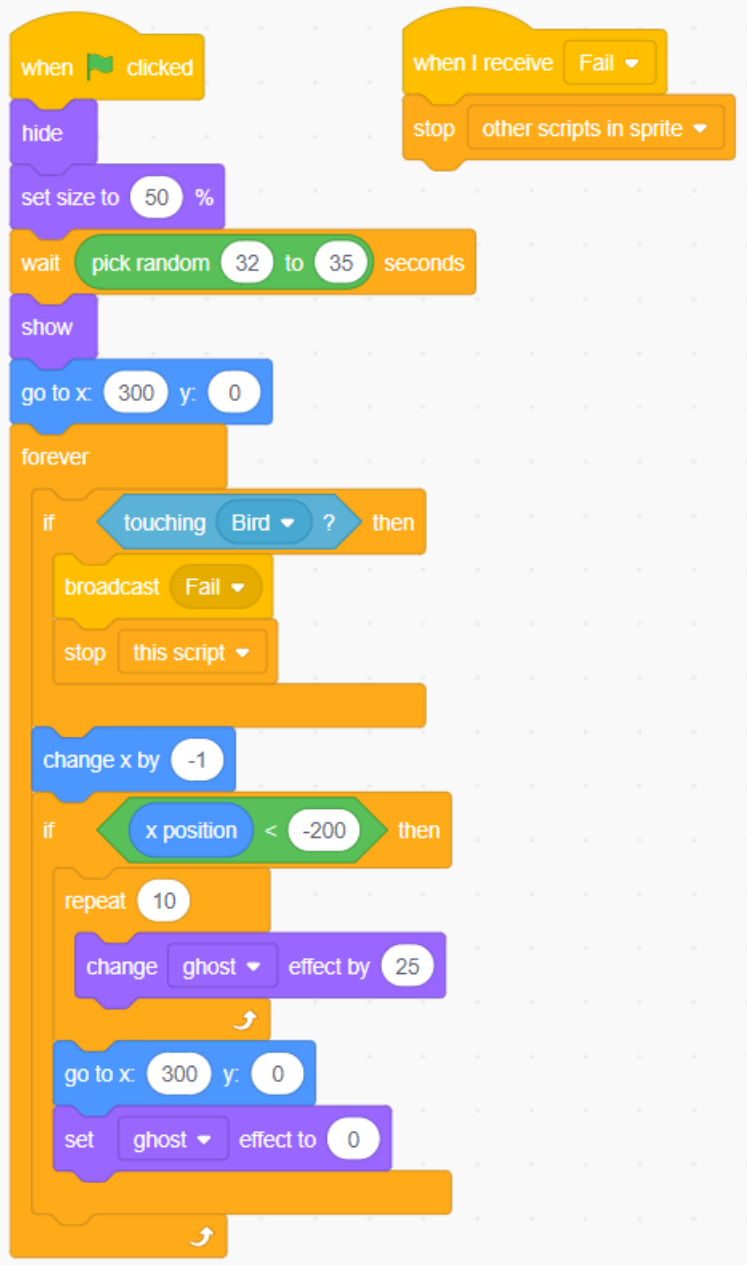


Lastly, when level 3 begins, we'll be adding a flying hippo as an extra obstacle to avoid. Since it will only appear in one level, there will be no need for a speed change.

11. Import the **Hippo** sprite from the Animals category in the Scratch sprite library.



12. Add the following code blocks for the **Hippo** sprite.



That's it! Try it out and remember. If it's too hard or two easy, then try adjusting some of the numbers that control speed or random changes.