

Create offline project

1. Open the Scratch 2 Offline Editor on your computer.
2. Select File → Save As and type Flapping Bat.

Delete the cat

Every time you create a new Scratch project, it will include one sprite, the Scratch mascot: Scratch Cat.

I am so not a fan of that smiling cat that most chapters in this book begin with two instructions:

1. Create a new project.
2. Delete the cat.

You can delete the cat (or any other sprite) by holding the Shift key on your keyboard while clicking it directly. A small menu will appear with the option to delete whatever you Shift-clicked. You will be doing a lot of Shift-clicking to save you time while working on Scratch projects.

So go ahead ... Delete that smiling Scratch cat!

If you are used to right-clicking with a mouse or trackpad, you may use that technique as an alternative to Shift-clicking.

Choose Player sprite

A sprite is any graphic element in a Scratch project other than the Stage, which represents the background. For our game, we will create three sprites: Player, Ground, and Pipe.

1. Look for the New Sprite area beneath the Stage and click the first icon: Choose Sprite from Library.
2. Select the sprite named Bat2 and then click OK.
3. Shift-click the Bat2 sprite and choose Info.
4. Change the name from Bat2 to Player because, in your game, the player will control the bat sprite.
5. Click the Back button (white triangle on blue circle) to close the Info window.

Paint Ground sprite

1. In the New Sprite area, click the second icon: Paint New Sprite.
 2. Shift-click the new sprite, choose Info, and change the name to Ground.
 3. Click the Costumes tab.
 4. Click the Rectangle tool on the Paint Editor canvas beneath the Costumes tab.
 5. Click the Solid rectangle option.
 6. Click a green color swatch.
 7. Click near the bottom-left corner of the Paint Editor canvas and then drag up and to the right side until you have a rectangle all the way across the bottom.
- If the ground sprite appears off-center on the Stage, click and drag it into place. (I'll also drag mine down a bit to give the bat more room to fly.)

Paint Pipe sprite

The goal of your game is to flap the bat's wings and fly through holes between two pipes. You will use a cool programming trick so that you only need one Pipe sprite.

1. Click the Paint New Sprite icon.
 2. Shift-click the sprite, choose Info, and change the name to Pipe.
 3. Click the Costumes tab.
 4. Click the Rectangle tool on the Paint Editor canvas beneath the Costumes tab.
 5. Click the Solid rectangle option.
 6. Click a gray color swatch.
 7. Click and drag across the middle of the Paint Editor canvas to draw a vertical pipe.
- To make a hole for the bat to fly through, click the Select tool, click and drag across the middle of the pipe, and press the Delete or Backspace key on your keyboard. (Don't worry if your bat is too big; you will fix that soon.)

Good work! Now you have all three sprites you need to make your game. What comes next? That white background is a bit plain. Here's a quick way to make a realistic sky.

Paint sky gradient on the Stage

The term gradient may be new to you. Scratch includes three types of gradients, which

allow you to fade between two colors. Part of what makes a sky look realistic is when it appears brighter toward the horizon and darker at the top of your game screen.

1. Click the Stage button.
2. Click the Backdrops tab.
3. Select the Fill with Color tool.
4. Click the Horizontal Gradient button.
5. Select the white color swatch.
6. Click the Swap Colors button.
7. Select a light blue color swatch.
8. Click anywhere to fill the Paint Editor canvas with the color gradient.

Doesn't the sky look more realistic now? Gradients can be used for all sorts of effects (like making something look metallic), so you will be using gradients quite a bit in the coming chapters. (Did somebody say, "Vector Robots"?)

Bring Game to Life with Code

I just HAD to sneak a bit of coding into this first chapter! Before Scratch, to program a videogame, you had to learn a bunch of commands, type them out, and make sure everything was in the right place. Well, no more, my Scratch Friend! Now, you just drag a few blocks into the Scripts Area of your selected sprite and you can have that bat moving around, responding to keys being pressed, slamming into pipes, and causing endless frustration ... I mean FUN!

Add flapping wings animation

If you click the Player sprite and then click the Costumes tab, you should see two costumes: one with wings up and one with wings down. If you click each costume, you will see the Player bat flapping its wings on the Stage. If you want the bat to keep flapping its wings, you need to add some code blocks.

Click the Scripts tab and you see ten categories listed: Motion, Events, and so on. Notice how all the blocks in each category are the same color.

As you follow the steps below, use the color of each block shown in the image to guide you toward the category where you will find it on the Scripts tab.

1. Select the Player sprite by clicking once on its icon beneath the Stage.
2. Click the Scripts tab.
3. Drag the following blocks into the Scripts Area and snap each one into place:
4. Click the Green Flag button on the top of the Stage to test your code.

Your bat should be flapping awfully fast. How do you slow it down?

Adjust flapping speed

Click the Stop button (beside the Green Flag button) to stop the code from running. See how the NEXT COSTUME block is inside a FOREVER block? You need to add another block inside that FOREVER block to slow it down. Can you find the right block in the Control category (same color as the FOREVER block)?

Drag and snap a WAIT block inside the FOREVER block, then click the Green Flag button again to test your code.

Now the bat is flapping too slowly, right? (This is beginning to feel like Goldilocks and the Three Bears!) What can you change if you want the wings to flap faster than 1 time every 1 second?

By default, the WAIT block has a value of 1. The white background means you can click and type to change the value. Try changing the value from 1 secs (seconds) to .2 secs, then click the Green Flag button again to test the change in your code.

How does it look? I'm going to stick with .2 seconds, but you can adjust the WAIT SECS to whatever value works best for you because YOU are the GAME DESIGNER!

Add keyboard controls

Here comes one of my favorite moments in game design! So far, your flappy project just seems like a simple animation, but these next steps will transform your static project into an interactive game. Let's start by allowing the player to control the bat. Like that other flappy game, you will tap one key to make the bat move up a bit; let's use the spacebar because it is the largest key and can easily be tapped by righties or lefties.

1. Click the Player sprite and then click the Scripts tab.

2. Drag the following new blocks to the right of the other blocks already in the Scripts Area:

3. Click the Green Flag button to test your code.

Each time you tap the spacebar, the bat should move a bit higher, so the CHANGE Y block must have something to do with vertical movement. In Scratch, the vertical position of a sprite is represented by the Y value and the horizontal position is represented by the X value. (I cover this more in later chapters.)

See how the shape of the WHEN KEY PRESSED block is the same as the WHEN GREEN FLAG CLICKED block? Each hat-shaped block represents an action, such as clicking the Green Flag button or pressing the spacebar. When the action occurs, the code snapped beneath its related hat-shaped block runs. If you want to use a different key to control the flapping, click Spacebar and drag down to a new keyboard key inside WHEN KEY PRESSED.

Why does the bat keep flying up until it reaches the top of the Stage?

What do we call that mysterious force that pulls objects toward the ground? Gravity!

Can you think of a good way to simulate gravity in your game using the blocks you have learned so far?

Add gravity to game

The following steps show one of the simplest ways to simulate gravity in your game.

1. Click the Player sprite and then click the Scripts tab.

2. Drag a third set of blocks into the Scripts Area to make the bat fall toward the ground.

3. Click the Green Flag button to test your code.

The player has to keep flapping or he or she will take a nosedive. Notice how there are now two sets of blocks that begin with WHEN GREEN FLAG CLICKED; part of Scratch's power comes from being able to run several sets of code blocks all at the same time.

While you are still on the Scripts tab, you should add a GO TO block with X and Y values each set to 0 to reset the position of the Bat sprite to the center of the screen when the game begins.

Full-screen Stage mode

Use the button in the top-left corner of the Stage to go into Full-screen Stage mode. It is useful to test your game at both the smaller and larger size because players have the option to change the Stage size at any time.

Make pipe move

In the original game, pipes keep moving from right to left, while the player moves up and down. To further distinguish your game from that other Flappy, you will add code which moves the pipe from left to right.

1. On the Stage, click and drag the pipe all the way to the left edge.

2. Click the Scripts tab.

3. Drag the following blocks into the Scripts Area to make the pipes move left to right:

4. Click the Green Flag to test your code.

The pipe should zoom right past the bat. The blocks should look familiar now. Can you figure out how to slow down the pipe? Click the Player sprite icon if you need to refresh your Scratch-block memory.

You could add a WAIT block to slow down the Pipe sprite movement. The more elegant solution would be to try a lower X value in CHANGE X BY. Click the default value of 10, change it to 4, and click the Green Flag button. If the pipe is still too slow, reduce X value a bit. If it's too fast, increase it a bit.

This should give you an idea about how to change the difficulty of your game, right?

The game won't be very difficult if the pipe passes right over the player as if the bat were not even there.

Add Collision to Your Game

Collision is at the heart of most videogames. Whether it's Pac-Man colliding with a pink ghost, Mario jumping onto a platform, or your character picking up a new tool in Minecraft, a game designer decides what will happen when each collision occurs.

What collisions does your game need to detect? Does the bat touch the pipe? Does the bat touch the ground? If either collision happens, what should occur? The game must

end.

Detect collision with the Ground sprite

1. Click the Ground sprite and then click the Scripts tab.
2. Drag the following blocks into the Scripts Area and select Player in the TOUCHING block.
3. Click the Green Flag button.

If the player floats to the ground, the game should end as soon as they collide.

You will usually put an IF THEN block inside a FOREVER block so the program will keep checking whether the condition is true or false and act accordingly (like a parent who keeps watching you to make sure you finish your homework before allowing you to fire up the Xbox or PlayStation). Now your program keeps checking to see whether the Player sprite is touching the Ground sprite from the instant the Green Flag button is pressed until the collision causes the STOP ALL block to end the game.

Since you will need the same code on the Pipe sprite, you can save time by copying the code blocks.

Copy blocks from Ground to Pipe

Click the top block (WHEN GREEN FLAG CLICKED) and drag the blocks from the Scripts Area directly onto the Pipe sprite icon beneath the Stage.

Once you release the mouse/trackpad button, you should see the original code snap back into place on the Scripts tab. If you click the Pipe sprite icon beneath the Stage, you should find the code has been copied there.

If the new blocks are overlapping the previous code, click and drag the top block to the right or beneath the other code.

Dragging the top block moves all the connected blocks. If you drag a middle block, only the blocks snapped beneath it stay connected.

Click the Green Flag button, and you should find that the game ends when the Player sprite touches the Pipe sprite or the Ground sprite.

Unless you got lucky, the size of the hole in your Pipe sprite is either too small for the bat to fly through or too large to make the game challenging.

Adjust Pipe Size and Location

Before increasing the hole size, consider one other thing the pipe should do. Part of the challenge from the original game comes from not knowing where the hole will be when each new pipe appears. So far, the Pipe sprite is always in the same vertical position, so the hole remains in the same spot.

Randomize vertical position

In Scratch, you can use a PICK RANDOM block inside the GO TO block so the pipe (and its hole) will appear in a different vertical position (Y) each time you run the game.

1. Click the Pipe sprite and then click the Scripts tab.
2. Drag a PICK RANDOM block into the Y value of the GO TO X Y block. (Notice how the round blocks can fit inside the round sockets of other blocks.)
3. Change the values in the PICK RANDOM block to -75 and 75.

Click the Green Flag button several times and you should see the hole appear in a different position each time. You will also see the pipe is too short to span the entire Stage when shifted vertically.

Increase sprite size on the Stage

To increase the size of sprites on the Stage, you can use the Grow tool.

1. Click the Grow tool (above the Scripts, Costumes, and Sounds tab).
2. Click ten times on the Pipe sprite on the Stage.
3. Click the Green Flag button to test your game.

The Pipe sprite should span the entire Stage no matter where it is positioned vertically. But I still cannot flap my bat safely through the hole. If you try to click any more with the Grow tool, you will find the sprite will no longer increase in size because you reached the limit. Instead, why don't you try shrinking the bat?

Decrease sprite size on the Stage

To decrease the size of sprites on the Stage, you can use the Shrink tool.

1. Click the Shrink tool (above the Scripts, Costumes, and Sounds tab).
2. Click five times on the Bat sprite on the Stage.
3. Click the Green Flag button to test your game.

It took me a few tries, but I was FINALLY able to flap my bat through the pipe hole successfully!

You may need to use the Shrink and Grow tools to adjust the size until it's possible to fly through the pipe, but not too easy. Then you will be ready to add more pipes.

Include additional pipes

Since you only need to have one pipe appear on the Stage, you do not need to create additional Pipe sprites. Instead, once the player successfully flies through the hole and the first pipe reaches the other side, you can reset the position of the Pipe sprite back to the left side of the screen.

1. Click the Pipe sprite icon beneath the Stage.
2. Click the Scripts tab.
3. Snap the following code blocks inside the first FOREVER block and change the values of your blocks to match the values shown.

The additional GO TO X Y block should look familiar. It has exactly the same values as the GO TO X Y block above it (right under the WHEN GREEN FLAG CLICKED block). So if the X position of the Pipe sprite is greater than 280, it resets the horizontal position to -200 and chooses another random Y position so the hole appears in a new location.

You