

MAKING YOUR OWN GAMES

Just like writing a story, creating a game takes planning. You can plan in your head, on paper, or by coding. However you do it, think about these ideas

SETTING

Where is your game set? Is the game background just a picture, or will it need to stop the player from moving?

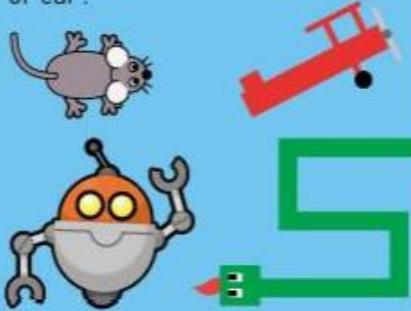
Look at page 15, step 5, or page 24 for how to draw a background. See page 54, step 1 for how to choose a background from the library.

```
repeat until touching color [green]?
  move (5) steps
end
```

You could use a **Repeat until** loop to move a sprite until it touches a particular color. *Drive Me Crazy* on pages 18–21 does that. *Snake* on pages 44–45 uses a more powerful method.

PLAYER

Who or what is the player? An animal, human, robot, or car?



AIM OF THE GAME

What is the aim of the game? To collect lots of objects? To avoid bad guys for as long as possible? To get to a particular place? To do something as fast as possible? To score as many points as you can in a certain amount of time?



Turn to *Catch the Donut* on pages 30–31 for an example of a collecting game. In *Cat and Mouse* on pages 52–53, your sprite has to avoid bad guys. In *Cross the Road* on pages 24–25, you need to reach a particular place. Page 23, step 5 shows how to put a time limit on a game.

MOVEMENT

How will the player move? By following the mouse? By pressing keys to move up, down, left, and right? By steering left and right, like a car?



To follow the mouse:

```
forever
  point towards [mouse-pointer v]
  move (5) steps
end
```

Add this code to your sprite to make it follow the mouse. The number in the **Move steps** block controls how fast the sprite moves.

To press a key:

```
when [right arrow v] key pressed
  point in direction (90) v
  move (10) steps
end
```

Choose your key using the drop-down menu on the **When key pressed** block. Pick the direction to move in with the drop-down menu on the **Point in direction** block.

To steer and rotate:

```
if [key [left arrow v] pressed?]
  then
    turn (5) degrees
end
```

Add this code to your sprite to make it rotate. Choose the key to press, and how far (or fast) to turn. You'll also need to add code somewhere else to make the sprite move forward.

VARIABLES AND SCORE

Score 30

Lives 1

What will you need variables for? To calculate the score? To change the speed of something? Will you set a maximum score to create a time limit for the game?

For help creating variables, see page 26.

Data

To reset the score at the start:

set Score to 0

To increase the score:

change Score by 1



Take a look at a simple game with a score variable by flipping to *Dog and Bone* on pages 28–29. Getting more complicated, *Flappy Fish* on pages 38–39 uses a variable to change speed. To use variables to simulate jumping, try page 67.



SOUND EFFECTS AND ANIMATION

Will you add sound effects? Will these effects be a sound file that is played when something happens, or a tune at the start of the game? Will you use variables to make the sound pitch change as the speed or score goes up?

To play a sound when a key is pressed:

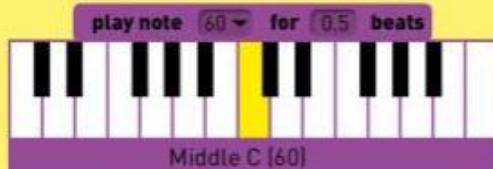
when space key pressed
play sound boing

To play a sound when two sprites collide:

if touching Cat2 ? then
play sound meow2 until done

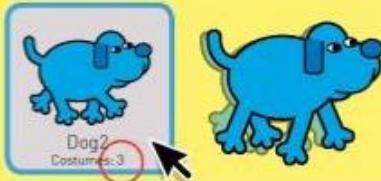
To play a short tune:

set instrument to 18
play note 60 for 0.5 beats
play note 64 for 0.5 beats
play note 67 for 0.5 beats
play note 72 for 0.5 beats



Animations can bring your games to life. Make your sprites look as if they are moving by making wheels turn or wings flap.

If you want to use a sprite from the library with ready-made animations, look for one that has more than one costume. See *Dog and Bone* on pages 28–29.



To draw your own animated sprite, turn to page 27. There is also a tip on how to add animations to a library sprite on pages 40–41, steps 3–4.



FINALLY... TEST AND TEST AGAIN

Don't try to code your whole game straight from start to end. Start by making short programs to test out different parts. For example, create a piece of code that makes your player move around. Experiment with it before you add other parts to the game.

Once you're happy with the basics, you can add details like animations or adjust the scoring system.

Ask other people to try out your game. Does it need instructions? Is it too hard or too easy? Keep experimenting, and don't be afraid to start again.

