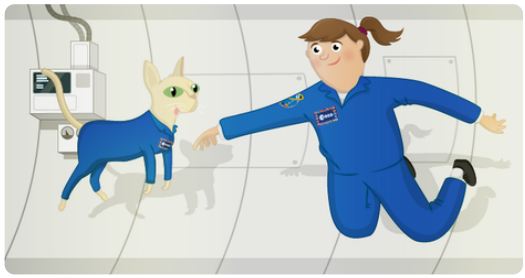
Projects

Weightlessness space game

Learn how to simulate the effects of weightlessness in space with this Scratch game

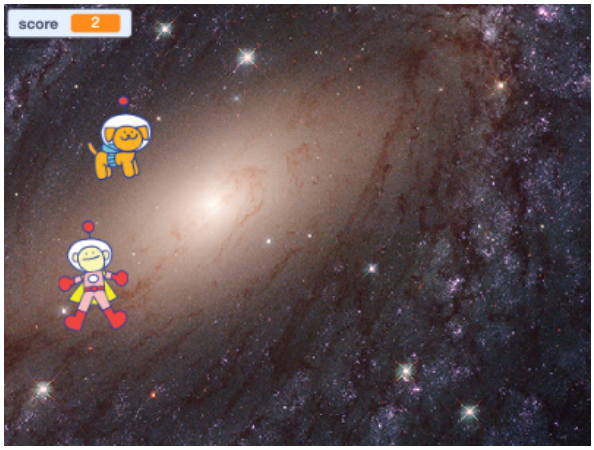


Step 1 You will make

Create a Scratch game to simulate objects floating in space.

You will:

- Simulate objects floating in space using a `forever` loop and `point towards` blocks
- Add and edit a 'Game Over' backdrop then `switch backdrop` using code
- Use the Scratch `timer` variable and make your own `score variable` to create a game



> Tim Peake was the first British ESA astronaut to visit the International Space Station. You can find out more about his mission, the forces of gravity, and the effects of weightlessness on The National STEM Centre website (http://www.nationalstemcentre.org.uk/timpeake).

Step 2 Set the space scene

On Earth, gravity pulls everything and everybody down towards the ground. In space it feels like everything is floating because it is no longer being pulled by our planet's gravity.

You will set up your weightlessness space game with a space backdrop and a space character.

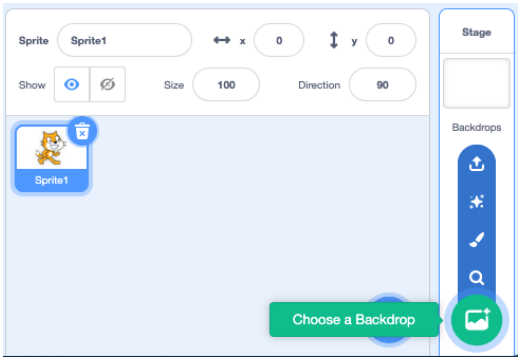
Open a new Scratch project (http://rpf.io/scratch-new). Scratch will open in another browser tab.

Working offline

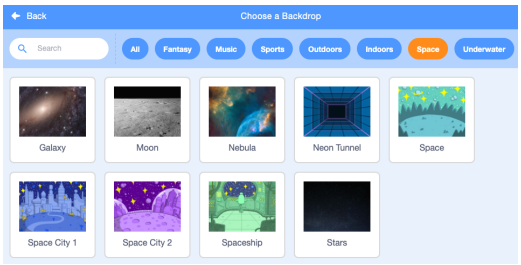
To set up Scratch for offline use, visit our Scratch guide (https://projects.raspberrypi.org/en/projects/getting-started-scratch/1).

A backdrop sets the scene by changing the way the Stage looks.

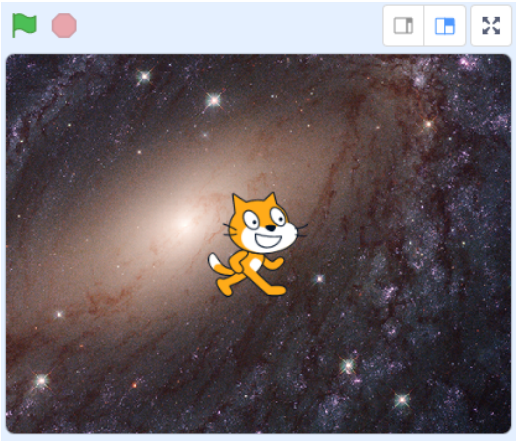
Click on Choose a Backdrop from the Stage pane:



Select the Space category and choose the backdrop you like best:

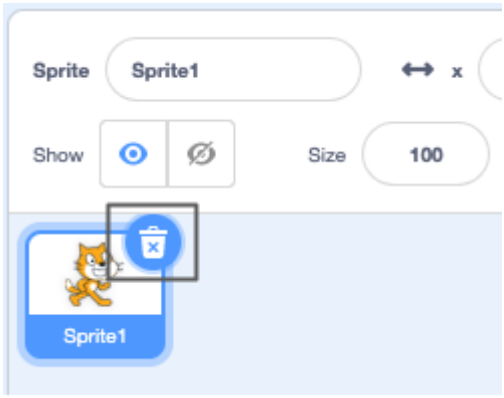


Your Stage should now look something like this:

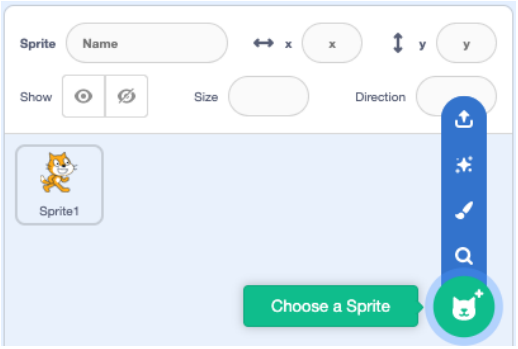


Sprites are characters and objects in Scratch. Can you find a sprite that has already been included in your project? That's Scratch the cat, but we need a sprite with an astronaut suit.

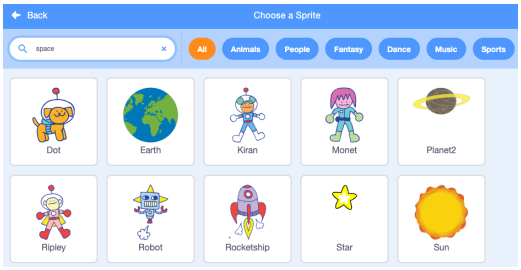
Delete the Sprite1 cat sprite by clicking the trash can in the Sprite list:



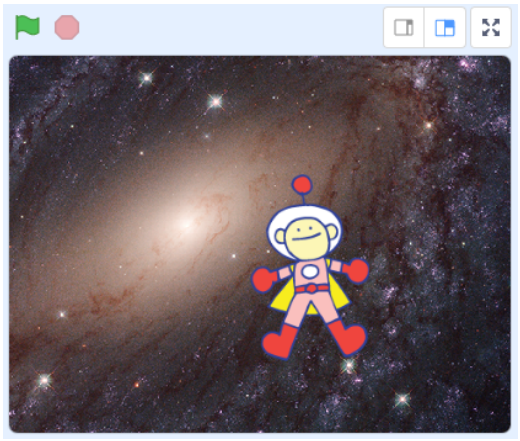
Click on Choose a Sprite from the Sprite list:



Click in the search box and type `space` then select your favourite character. We've chosen the Ripley sprite:



Your stage should look something like this:

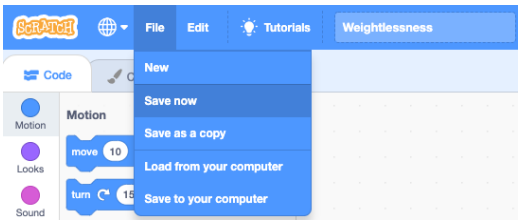


Save your project

Type `Weightlessness` into the project title box at the top of the screen:

Tip: Give your projects helpful names so that you can easily find them when you have lots of projects.

Next, click on File, and then on Save now to save your project:



If you are not online or you do not have a Scratch account, you can click on Save to your computer to save a copy of your project.

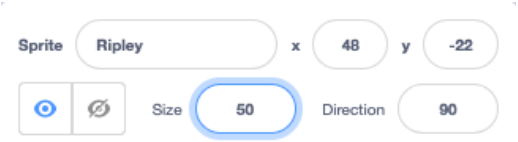
> It takes astronauts on board The International Space Station some time to get used to the feeling of weightlessness. These Weightlessness videos (https://www.stem.org.uk/elibrary/resource/29076) from NASA show how people and objects behave in space.

Step 3 Make your sprite weightless

Your character is in space and will be floating around feeling weightless.

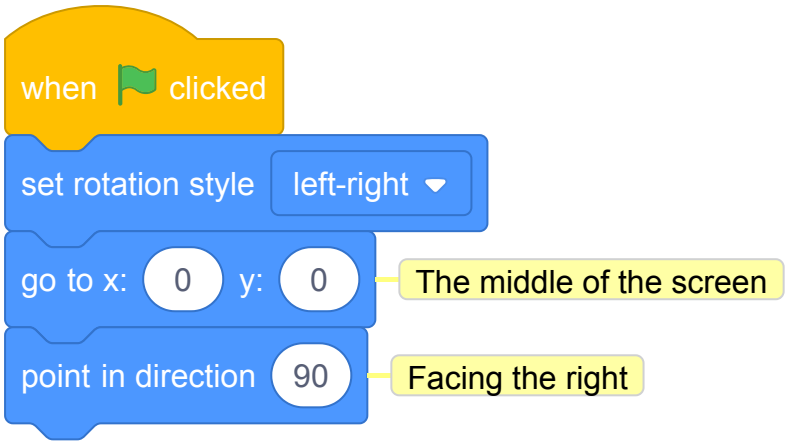
At the moment, your character sprite is quite big so your game will be too easy!

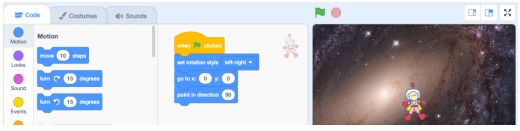
From the Sprite pane, click on the Size property and change the value to `50`%:



A group of connected blocks in Scratch is called a script. You will add a script to your character sprite so it starts in the middle of the Stage then moves `forever` through space.

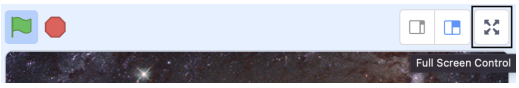
Click on the `events` blocks menu and drag a `when flag clicked` block to the Code area. Connect some `motion` blocks to the bottom of your script to get your character sprite ready to move:





Test: It is a good idea to test your project regularly in full screen mode using the Full Screen Control. This view will let you see your game the way a user playing your game would see it.

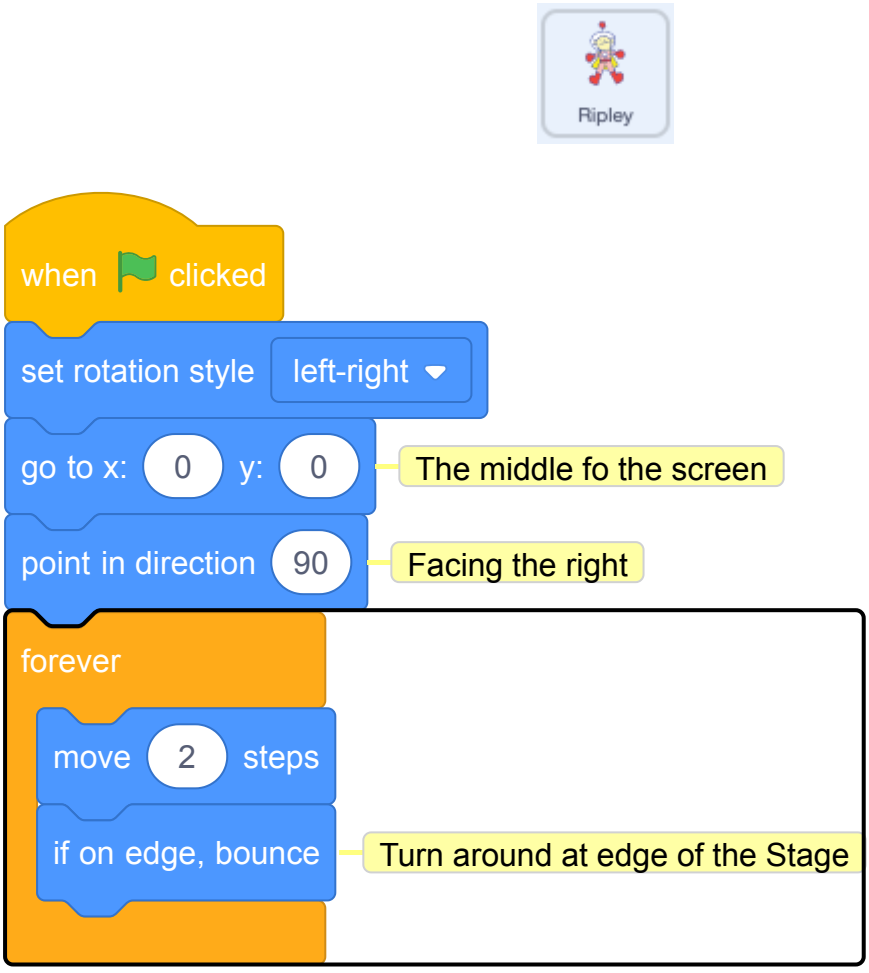
To run your project in full-screen mode in Scratch, go to the area above the Stage and click on the icon with four arrows that point outwards. This is the Full Screen Control icon:



Click on the `green flag`, your character sprite will move to the middle of the Stage.

To exit full-screen mode, click on the Full Screen Control icon again. It will have four arrows that point inwards.

Add a `forever` loop to the bottom of your script. Insert `motion` blocks inside your `forever` loop to create weightless movement:



Test: Enter full screen mode then click on the `green flag`, your character sprite will float across the Stage.

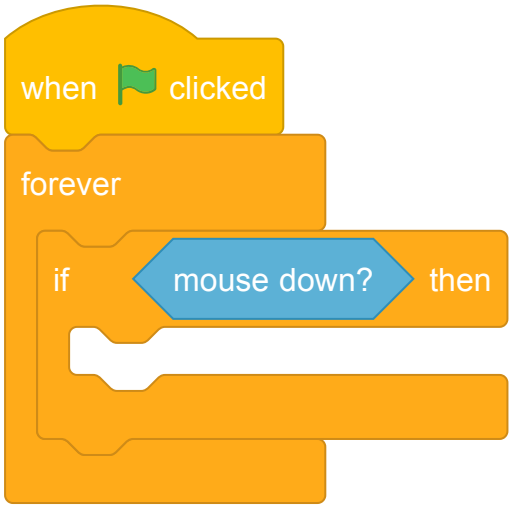
Save your project

Step 4 Control direction

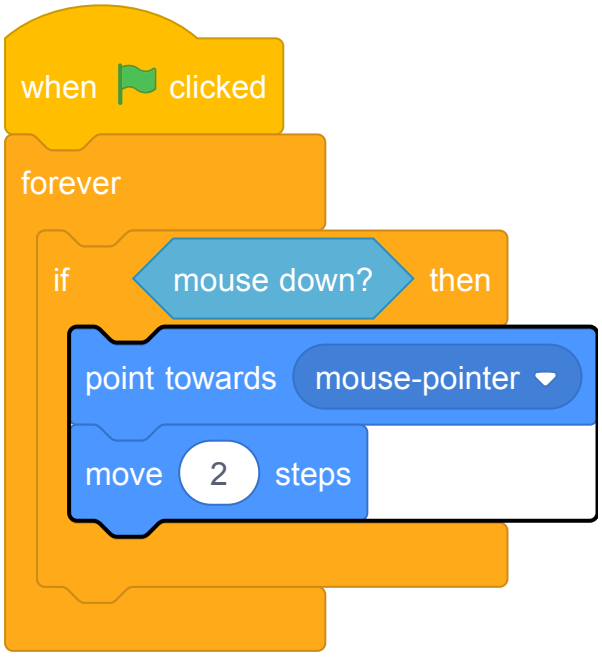
Your astronaut is wearing a spacesuit with thrusters, allowing them to control the direction of their movements.

Your program needs to continuously detect when the left mouse button is pressed. To do this, start a new script with a `when flag clicked` block then add a `forever` loop to your script.

Scratch's `mouse down` block works for fingers on mouse buttons and on touchscreens! Inside the `forever` loop add an `if ... then` block to detect if `mouse down`:



To get the character sprite to respond to where the user clicks, add the following blocks so the character sprite will `point towards mouse-pointer` and `move 2 steps`:



Test: Enter full screen mode then click on the green flag to run the program to test that your character faces and moves towards your mouse-pointer.

Make sure that your character moves in all directions.

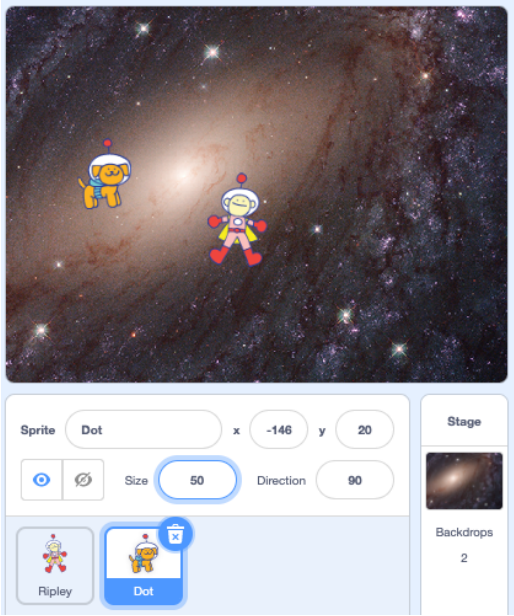
Save your project

Step 5 Catch a sprite

Your character needs an object to chase. Your object could be an item or another character. What will you choose? We used the Dot space dog sprite, but you can use whatever you like.

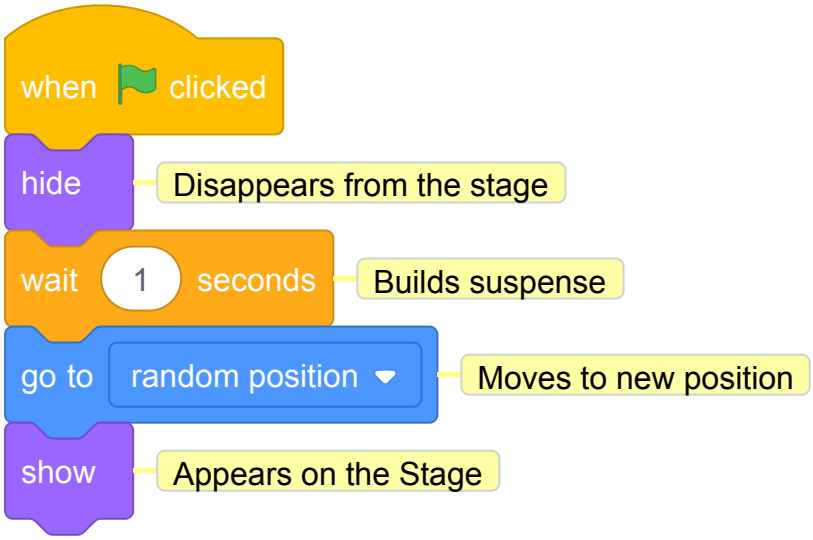
Click on the Choose a Sprite button to see the available sprites. Find one you like, and click on it to add it to your project:

Tip: You'll need to change the size again so it suits your game:



Add code to make your object sprite appear in different places each time the `green flag` is clicked.

Drag a `when flag clicked` block to your object sprite. Use `hide`, `go to random position`, `wait`, and `show` blocks so your object sprite appears in a surprise position:

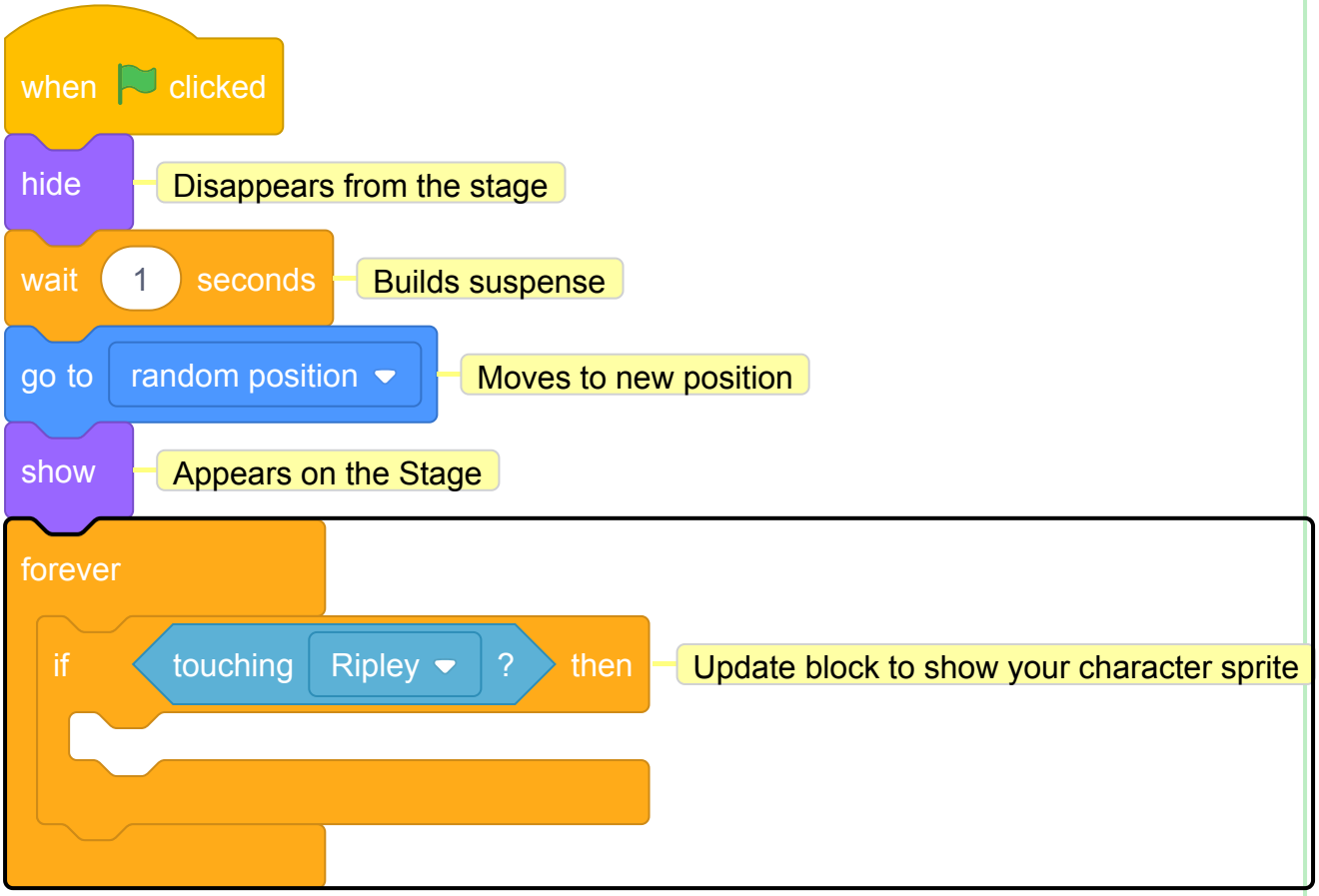


Test: Enter full screen mode then click on the green flag and run your project a couple of times. Does your object sprite move to a different position each time?

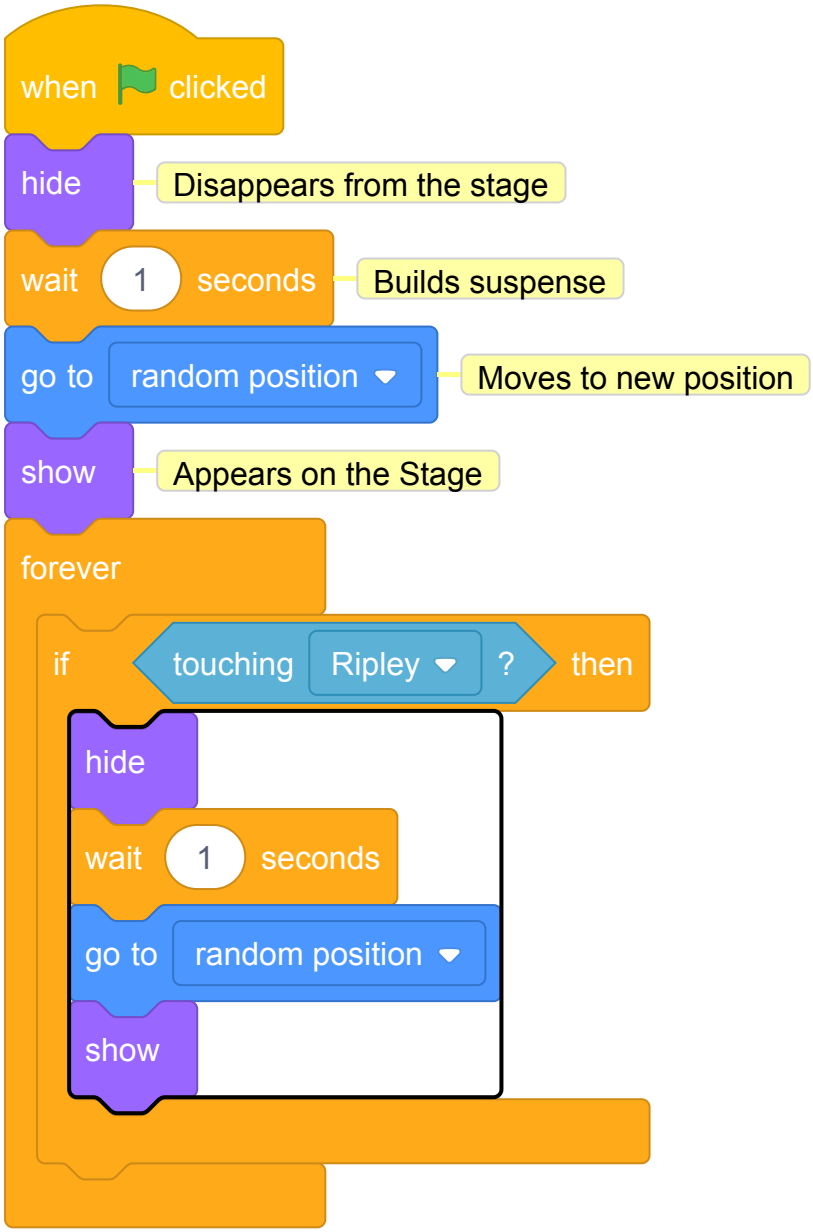
To turn your project into a game, the script needs to repeat every time your character sprite and object sprite touch.

Add a `forever` loop to the bottom of your object sprite script and insert an `if` condition block inside.

To sense `if` your sprites are touching, add a `touching mouse-pointer ?` block and select your character sprite name from the drop-down menu:



Inside the `if` block, add the same four blocks you used earlier:



Test: Enter full screen mode then click on the green flag and move your character to the object. When your sprites touch, your object sprite will hide and re-appear in a new position.

Save your project

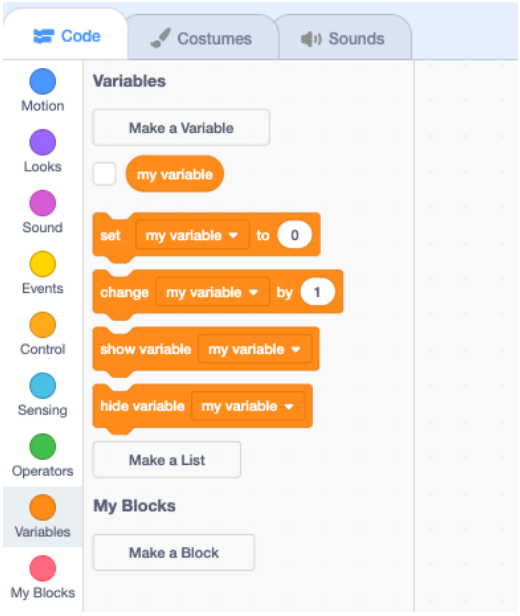
Step 6 Make a game of it with variables

Games often have features to show players how well they did against other players or themselves. Can you think of a game that shows you how well you did?

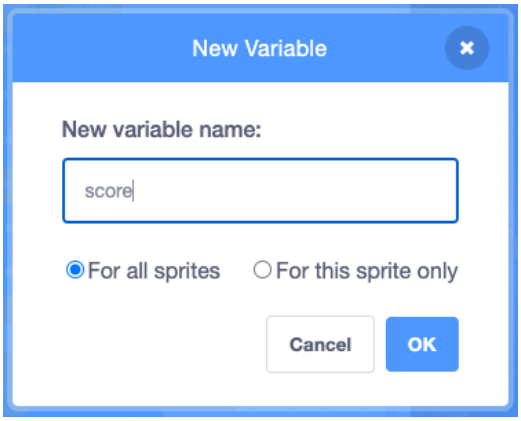
In your weightlessness game, you'll use the built-in Scratch `timer` variable and create a `score` variable to show how many objects the character caught in 30 seconds.

First, create a `score` variable.

Go to the `variables` blocks menu and click on the Make a Variable button:

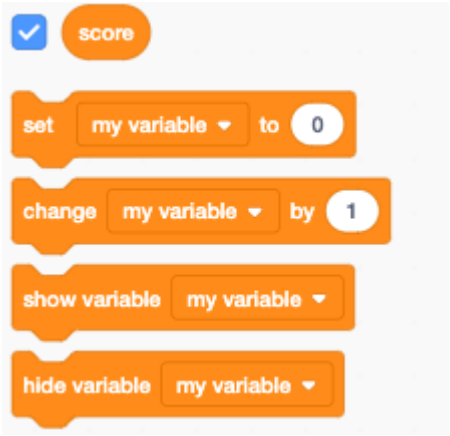


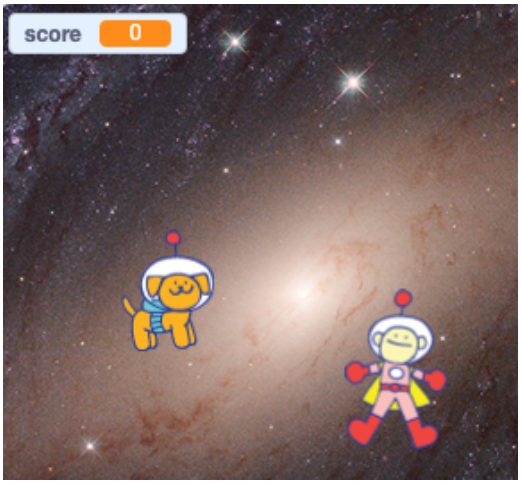
Type `score` as your New variable name and select For all sprites:



The `score` variable appears in the `variables` blocks menu. It has a tick next to it, which means it appears on the Stage.

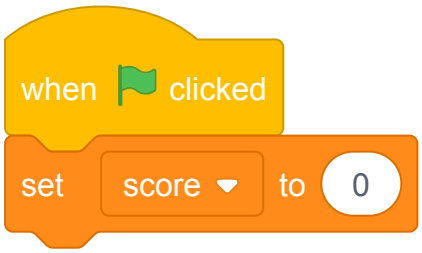
Try unticking the box to see what happens. You can choose whether or not you want to display your `score` on the Stage during the game.



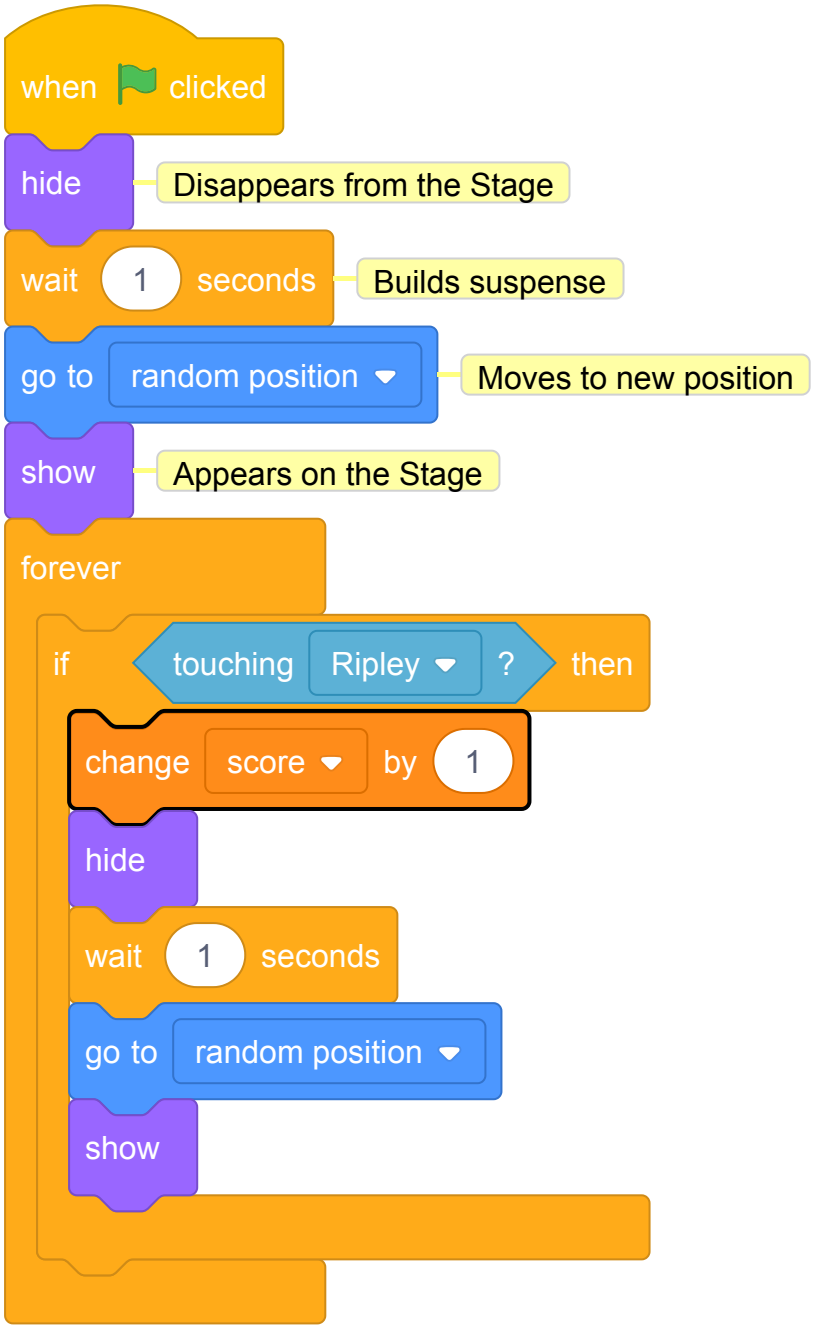


The `score` variable will start at '0' `when flag clicked` and store the number of times your sprites `touch`.

Click on your character sprite and add a new script to reset the score:



Click on your object sprite and find the `if touching` script. Insert a block to `change score by 1`:

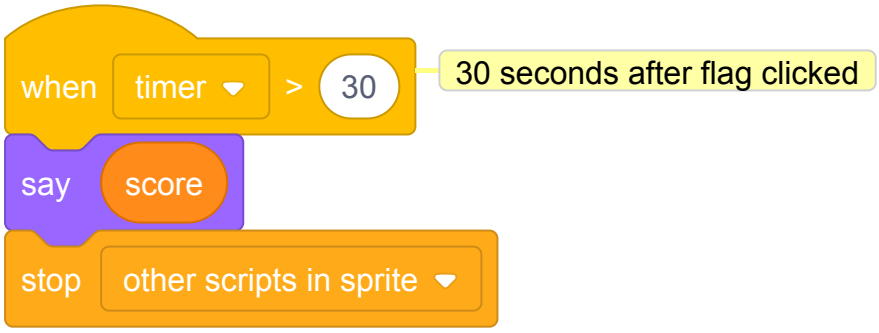


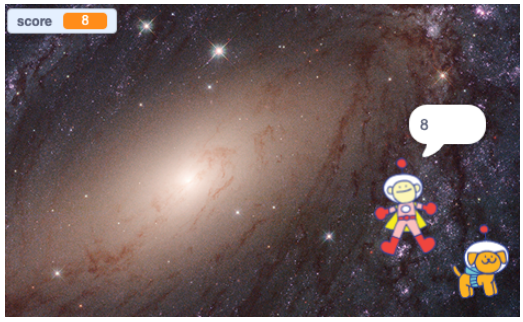
Test: Enter full screen mode then run your project. Your score will update each time the sprites touch. Run your project again to test that the `score` goes back to zero when you start a new run.

Now that you have a `variable` to store the score, you can use this in other ways. Your character sprite can `say` the `score` at the end of the game.

Click on your character sprite. Add a new script to end the game `when timer is greater than 30`.

Tip: To find the `when timer >` block, use the `when loudness >` block from the `events` blocks menu and change the drop-down option to `timer`.





Save your project

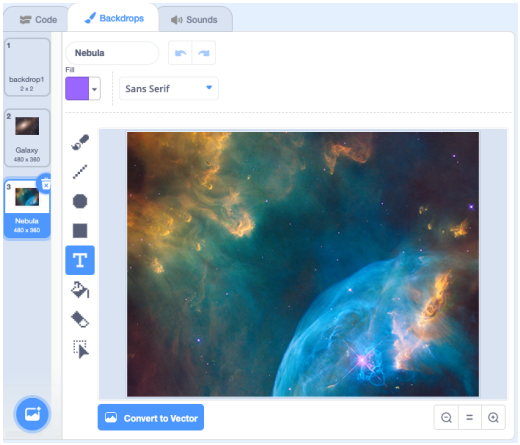
Step 7 Create a Game Over screen

Game over screens are a great way to end video games. Some have a simple 'game over' message whilst others have details of who made the game, and some may even feature all the sprites involved in a game.

To create a game over screen, you'll need to add a backdrop and edit it with colours, shapes, or text.

Click on the Choose a Backdrop button. If you want to continue your space theme, click on the Space category. We chose Nebula, but you can choose any backdrop you like:

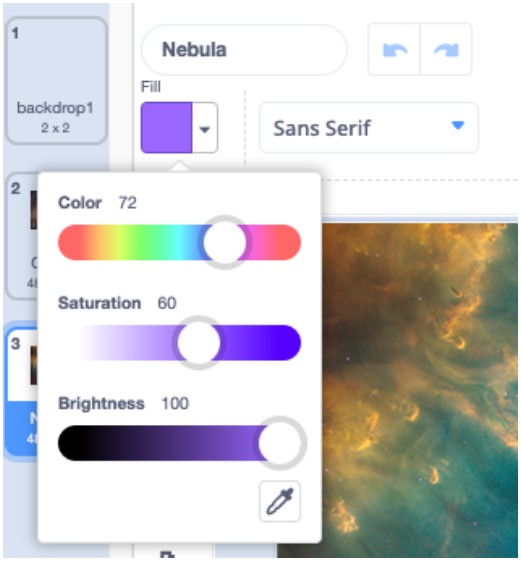
Click on the Backdrops tab and you will see the backdrops currently in your project with your new backdrop at the bottom of the list.



Edit your backdrop to include some text.

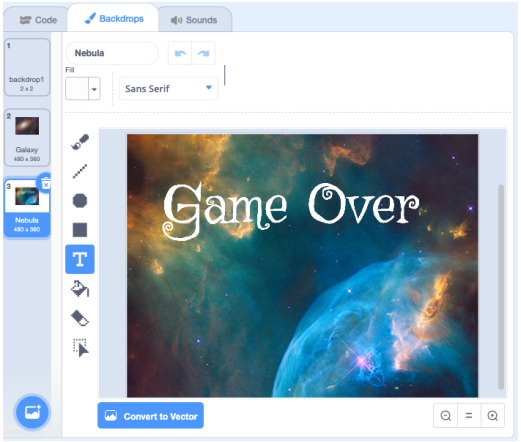
Click on the Text tool:

Choose the colour and font style you want from the drop-down boxes:



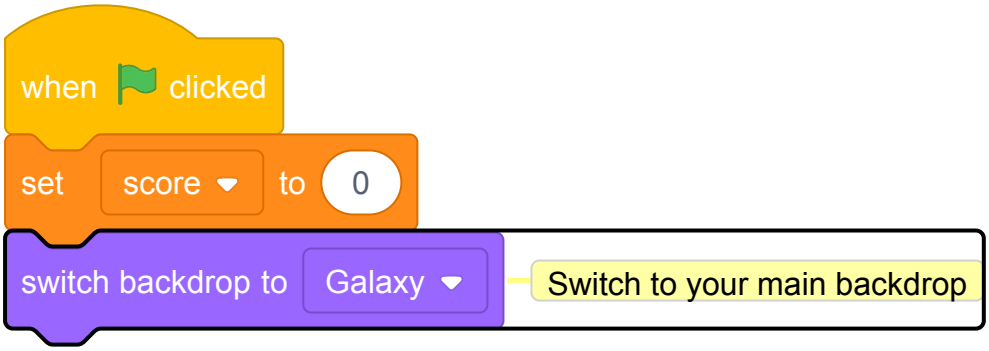
Tip: Use the sliders in the Fill colour chooser to pick your colour. If the colours turn only to black or white, you might need to adjust the Saturation or Brightness sliders.

Click where you want the text to appear on the backdrop, then type `Game Over` or words of your choice.

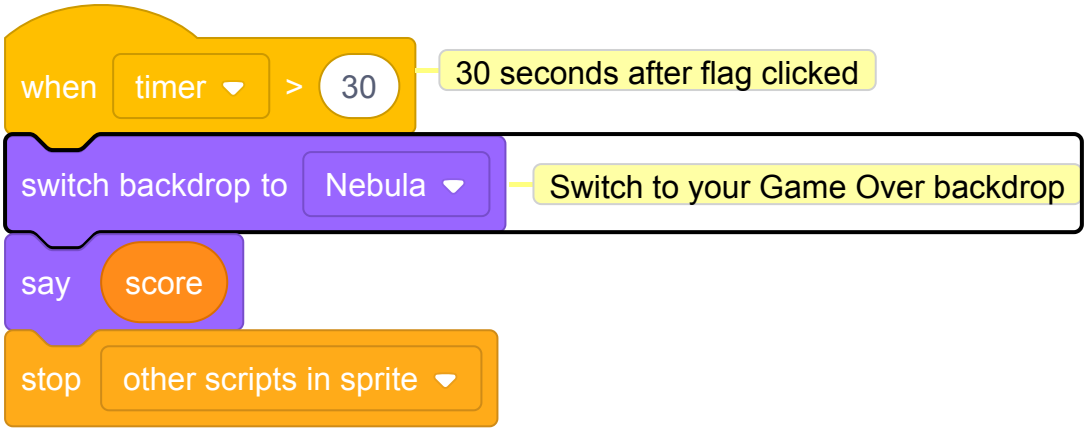


Set up your game to start on your first backdrop, then change to the Game Over backdrop at the end.

Click on your character sprite, at the end of your `set score to 0` script, add a `switch backdrop` block:



Find your `when timer > 30` script and insert a `switch backdrop` block at the top.



Test: Enter full screen mode then run your project. Does your backdrop change to the Game Over screen? When you run it for a second time, does it switch back at the start of the game?

Save your project

Step 8 What next?

- Could you add extra sprites to catch? Perhaps they appear less frequently but score more points when you catch them.
- Could you give your character a speed boost for a few seconds when you press the space key?

Play

What happens when you catch the robot?

Weightlessness extended: See inside (https://scratch.mit.edu/projects/501858568/editor)

- Check out some of our other space-themed projects (https://projects.raspberrypi.org/en/projects?interests%5B%5D=space).