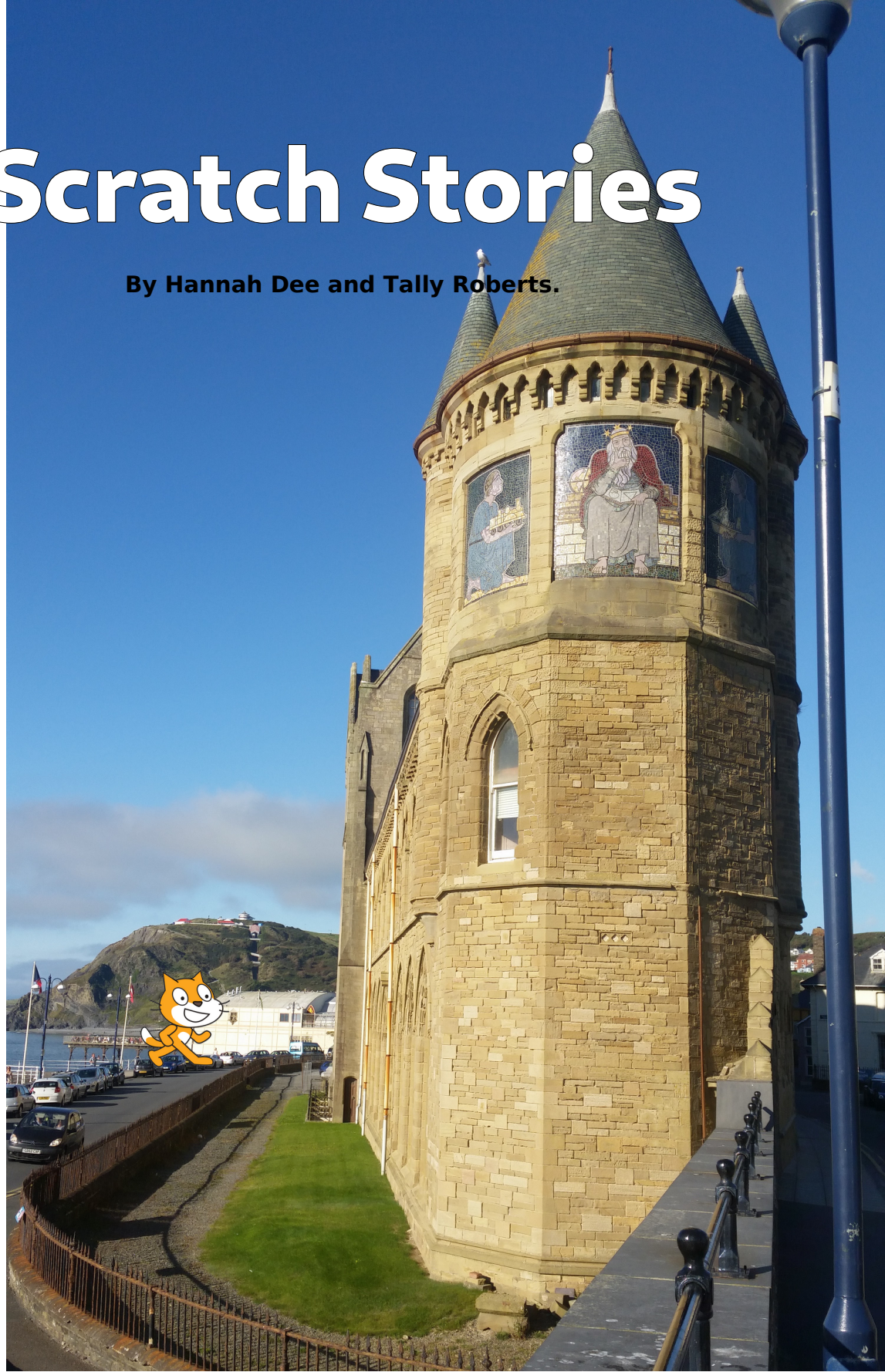


Scratch Stories

By Hannah Dee and Tally Roberts.



1872 PRIFYSGOL
ABERYSTWYTH
UNIVERSITY

1872
150
2022



Cyngor Sir
CEREDIGION
County Council



Story planning.....	2
An introduction to Scratch.....	5
Choosing your character.....	8
Making your character say something.....	11
Making your character move.....	13
Changing the Backdrop.....	16
Scene changes.....	17
Animating two or more characters at the same time.....	17
Appearing and disappearing.....	18
Keeping time.....	19
Sending messages.....	19
Making your character turn.....	21
Making your character shrink and grow.....	22
Making your character change appearance completely.....	24
Some example animations.....	26

This workbook helps you create animations in Scratch. It has some information about story planning, then programming characters to move and say things, with scene changes, multiple characters interacting, and some graphics. It was written as part of Aberystwyth University’s 150th Birthday celebrations and accompanies some community workshops working on animations about Aberystwyth: we hope you find it useful for making animations about all sorts of things.



Cyngor Sir
CEREDIGION
County Council



Story planning

Stories have a beginning, a middle and an end. They also have characters (the people in the story), and a setting (places where the story happens).

This workbook helps you decide how to organise your story and collect together images you'll need to animate it.



Who are the characters in your story? Are they humans, or animals, or robots? List the names of your characters here:

1

2

3

4

5

Don't worry if you don't have 5 characters. Some great stories only have one or two characters.

Next, think about the setting of your story. Does it all happen in one place? Or are there different places? (It could start in Aberystwyth, then move to a rocket, and end up on the Moon).

My story starts in ...



Then it moves to ...

My story finishes in ...

Because we are animating the story (and we don't have unlimited time) it is best to choose a small number of locations, even if this means only doing part of the story to start with. You can always add more later.

Storyboarding helps lay out your story in pictures. It's a bit like a cartoon of what happens in the story. If you want, you can use this template to draw a rough outline of your story.



Scene 1	Scene 2	Scene 3

You are going to need a picture of each character, and a picture of the background for each location. You'll be making the animation in Scratch which has some Sprites you can choose from. If these aren't the ones you want, you can either find some images on the Internet, or you can draw them yourself.

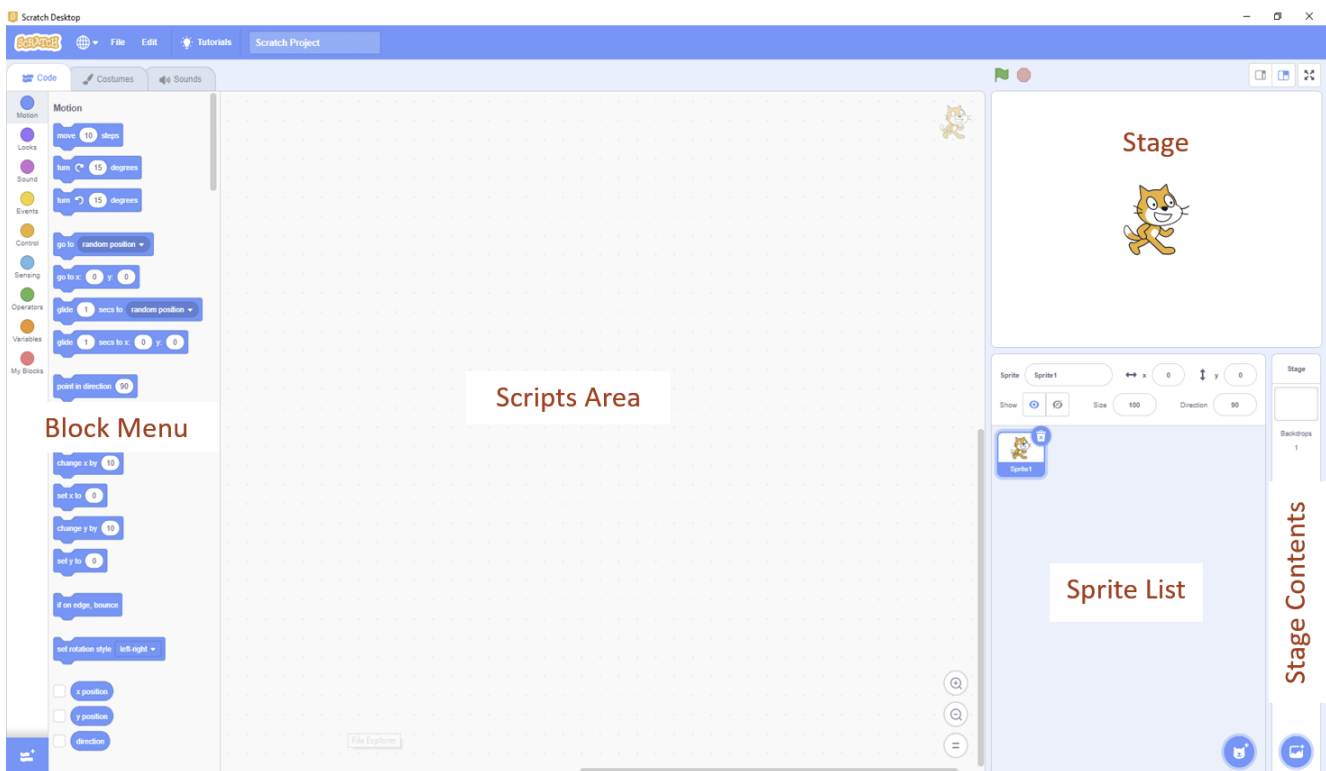
In computing and in animation we call the images (and sounds) that make up our animation "Assets". It's a good idea to collect these before you start coding, although you can continue to build a collection of story assets as you go, and it's easy to change your mind about the appearance of a character, an object or a background.

An introduction to Scratch

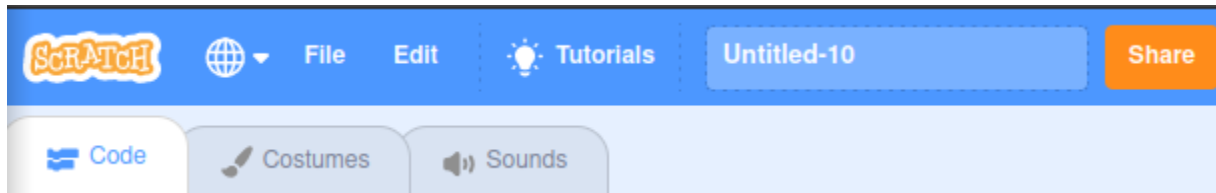
Scratch is a programming language designed to help people learn about computers, animation and programming. It makes it easy to move images around a screen, and incorporate sounds. You can use it for all sorts of projects (games, art, interaction, maths), but this booklet will concentrate on scratch for storytelling and animation.

You can use browser-based Scratch or download the desktop version. The browser version can be found at <https://scratch.mit.edu>.

See the image below for the general layout of Scratch. We have added labels for the key areas of the screen to help you find your way around - look out for the highlighted text to show when these label names are used



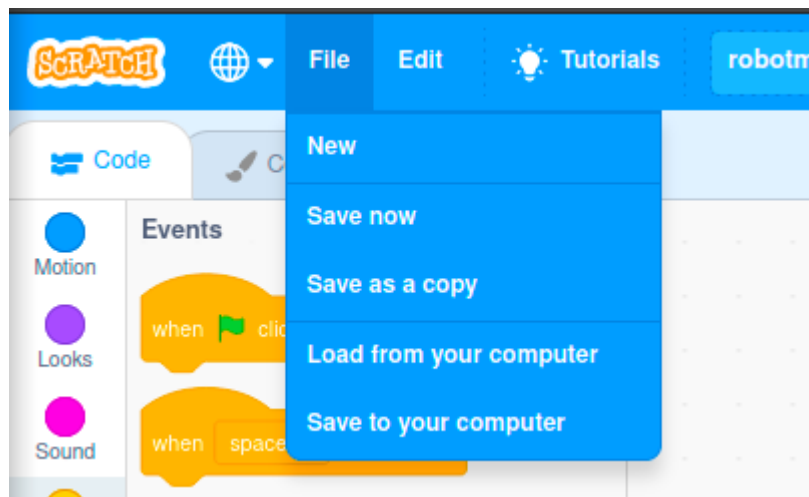
The first thing is to create a file name for your animation. You can do this by entering your chosen file name in the box next to the Share button. In the image below it contains the text “Untitled-10” – yours will say something similar, and you can just replace that text with your filename.



Whilst we are looking at the top row of the Scratch window it is worth mentioning the File menu. This is different if you're using Scratch offline or scratch on the web.

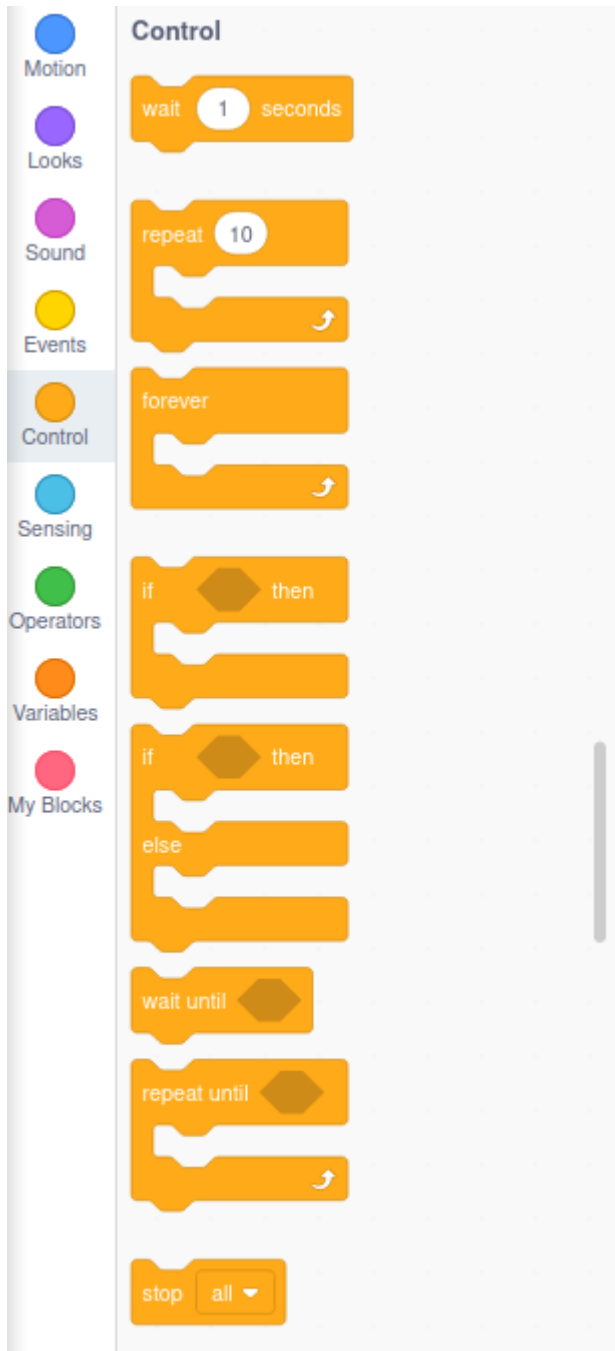
Scratch offline: This is a piece of software you can download, and it has a File menu like most software. You can save your Scratch animations in the same way you would save a word-processing document.

Scratch on the web: this is where you can save, make a copy, load from your computer or save to your computer. By default, Scratch saves things online on the Scratch site.



Now take a little while to look at the 'blocks' in the **Block Menu**.

They are organised into different categories which are colour coded. This makes it a bit easier to see what is going on.



You can scroll down through all the different blocks using the scroll bar, or you can click on one of the coloured circles to jump to the blocks in a specific category.

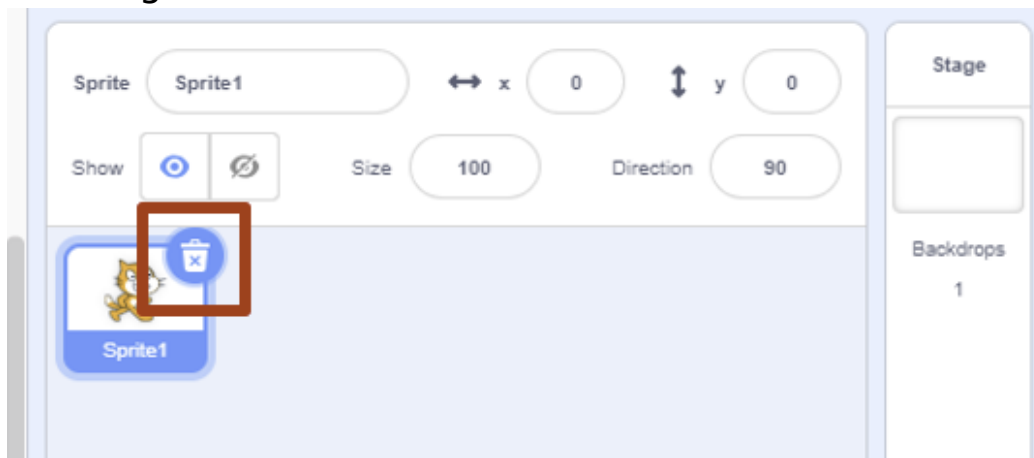
The picture to the left shows what you'll see if you jump to the "Control" blocks, which are coloured a dark yellow.

Choosing your character

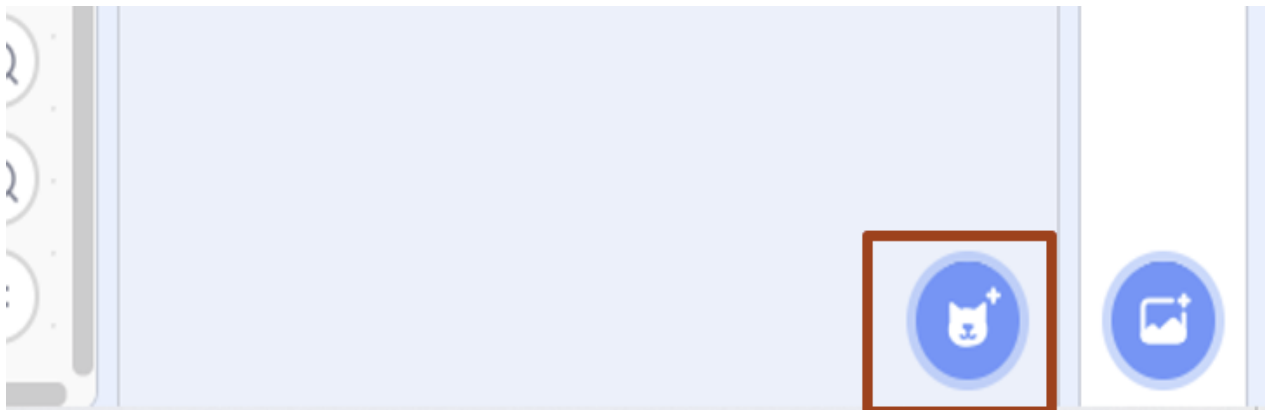
If you look at your **Stage** you will see a cat image. This image is called a **sprite**. You can also see it in the **Sprite List**. In Scratch, all code and instructions are attached to Sprites - you can think of a Sprite as a character in your story.

You probably do not want to use a cat as your main character, so let's change this sprite now.

This means we need to delete the cat from our **Sprite List** using the bin image shown here.

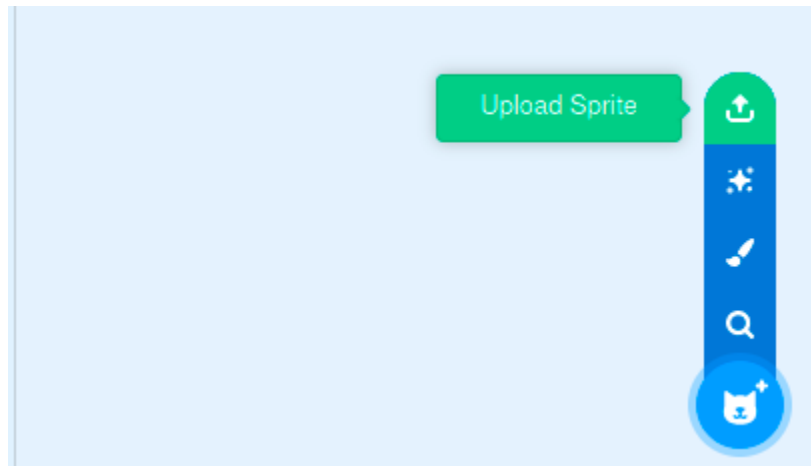


Now our **Stage** is empty. We can add a new sprite using the icon in the **Sprite List** as shown below.



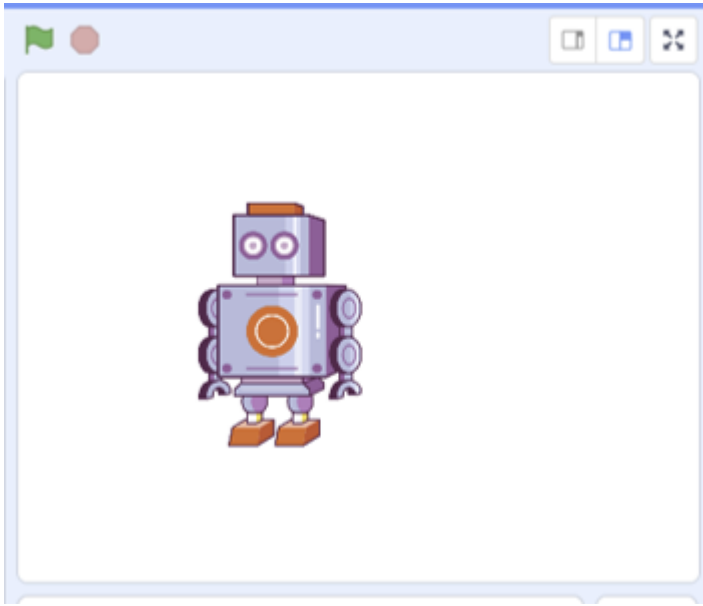
This will show you a selection of different sprites. If you want to use one of these, that's great! You can select the image and it'll appear as a sprite.

If you want to upload your own image to use as a sprite, hover the mouse over the cat's head "New Sprite" icon and a menu will appear. This menu lets you draw a sprite using an editor (the paintbrush), get a random sprite (the stars) or upload an image (the arrow+tray).



You can upload most types of image file to use as a sprite. If your image has a background, though, it is easier and better to remove that using an image package (like Paint, or Gimp) than to upload it to Scratch and try to tidy it up there.

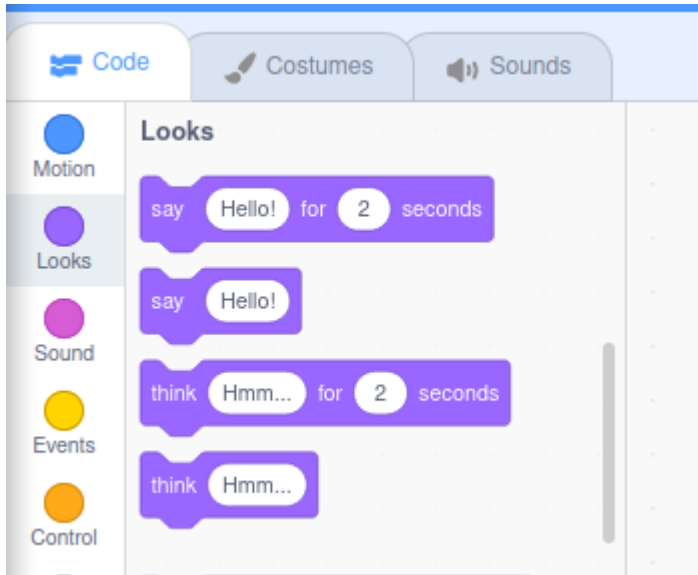
Your **Stage** should now look like this:



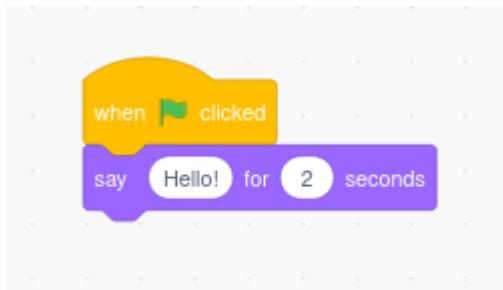
but with your image in the place of the retro robot.

Making your character say something

To make your character say something, we use blocks from the “Looks” section. These are purple in colour. Drag one of these (for example, the “Say Hello for 2 Seconds” block) from the **Block List** to the **Scripts Area**.



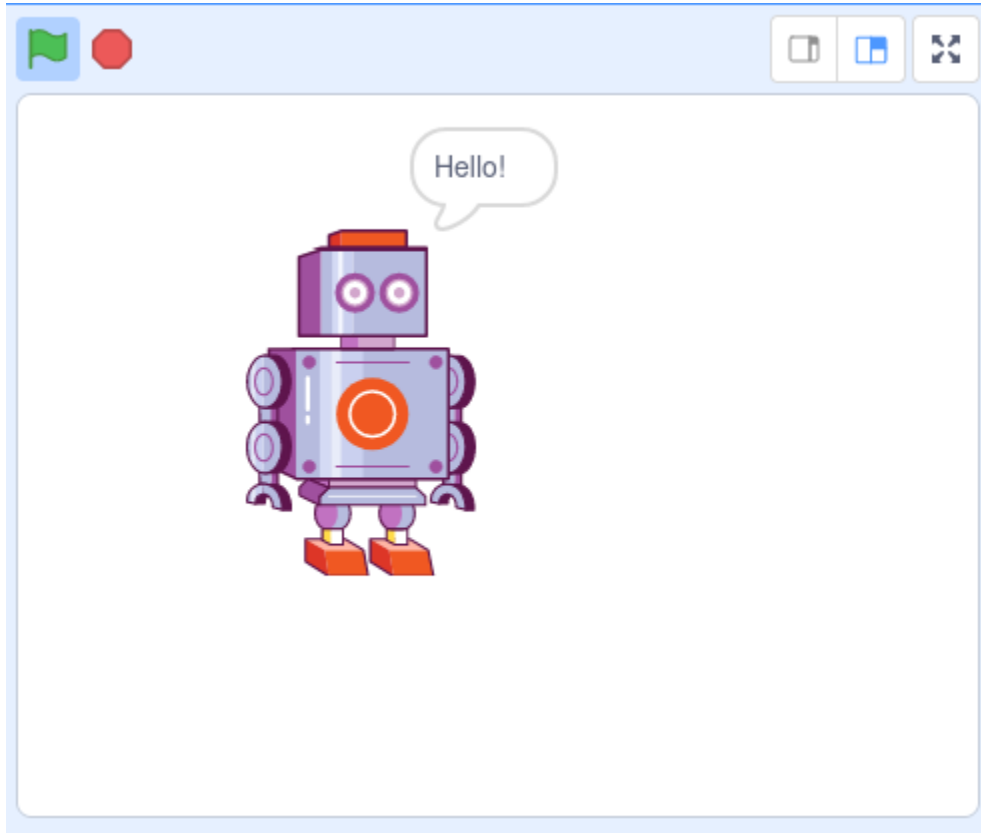
Moving blocks to the **Scripts Area** is how we tell Scratch what we’d like our sprites to do. You can’t just put the blocks there though, you also have to tell Scratch *when* you’d like it to do things. We’ll do this by dragging something in the yellow “Events” category over.



“When green flag clicked” is the block we normally use to start a scratch program, so drag that block in and place it so it clicks into place above the purple “Say hello!” block.

The **Stage** is where you can test your program. When writing a program it is good to test it after every **script** change. Below is an image of the **Stage**.

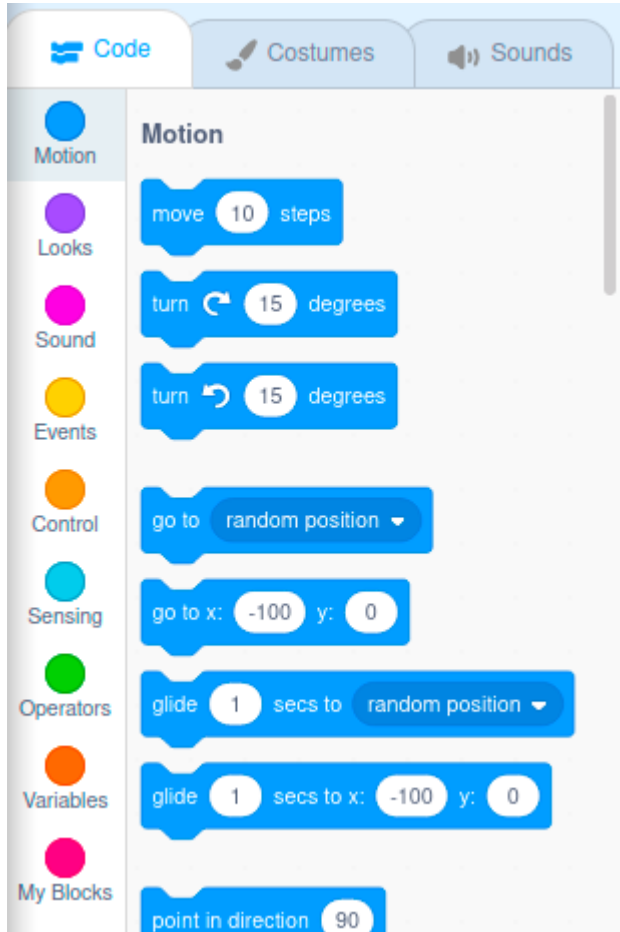
The green flag in the top left is how we start our program. Try clicking this now, and your sprite should say “Hello!”.



Try changing the “say hello” block for a “think hello” box. What is the difference?

Making your character move

It is now time to make our character move. Motion blocks in Scratch are blue, and appear at the top of the **Block List**.



Find the 'Go To x: y:' block in the **Block List**, as highlighted in the image to the left. Now click and drag the block into the **Scripts Area**.

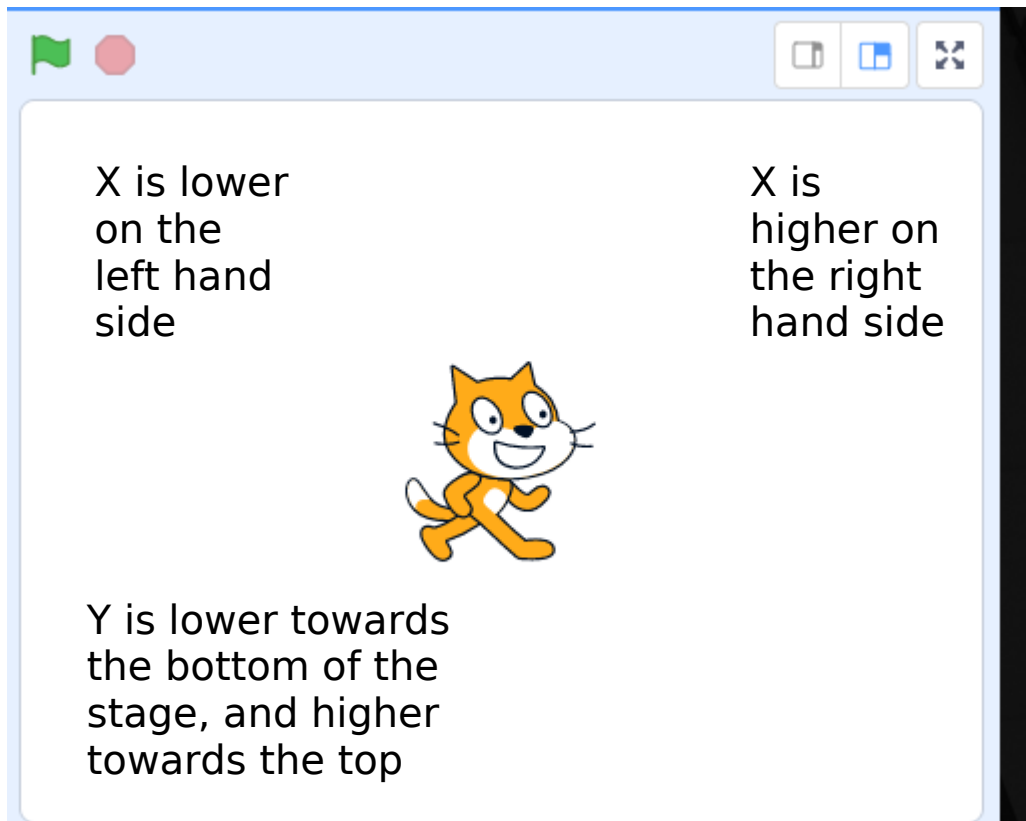
You should be able to drag it inbetween the two blocks you already have, so that it sits below the "When green flag clicked" block and above the "Say Hello" block.

Try clicking the green flag, and work out what happens. If nothing happens, try changing the numbers next to x: and y: - you can type in new numbers.

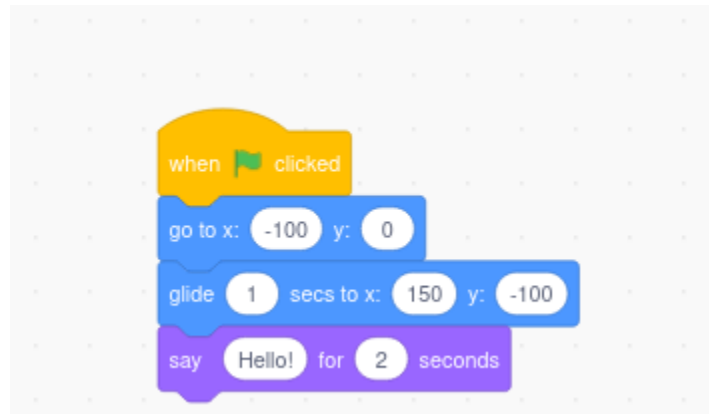
Experiment to see what values of x and y work best for you.

What do minus numbers do? What is the biggest x you can use and still have your **Sprite** on the **Stage**?

You might have worked out that if the sprite is already at the location x,y it doesn't move. You might also have worked out that minus numbers for x are on the left hand side of the **Stage**, and minus numbers for y are the bottom half of the **Stage**.



Now add a new block to your script. Drag a “glide 1 second to x: y:” block to your **Scripts Area**. Put it above or below the “Say Hello” block - it’s up to you. Change the values of x and y to things that make sense for you, and keep testing (keep clicking the green flag to see how it works).

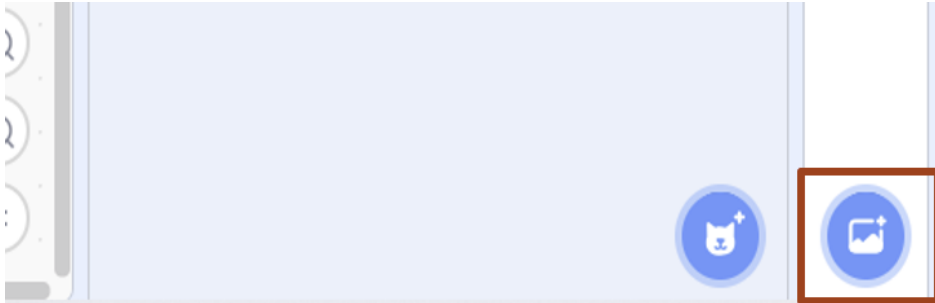


Your blocks should look a bit like this: And when you click the green flag, your Sprite should move and say something.

We now have the start of an animated story.

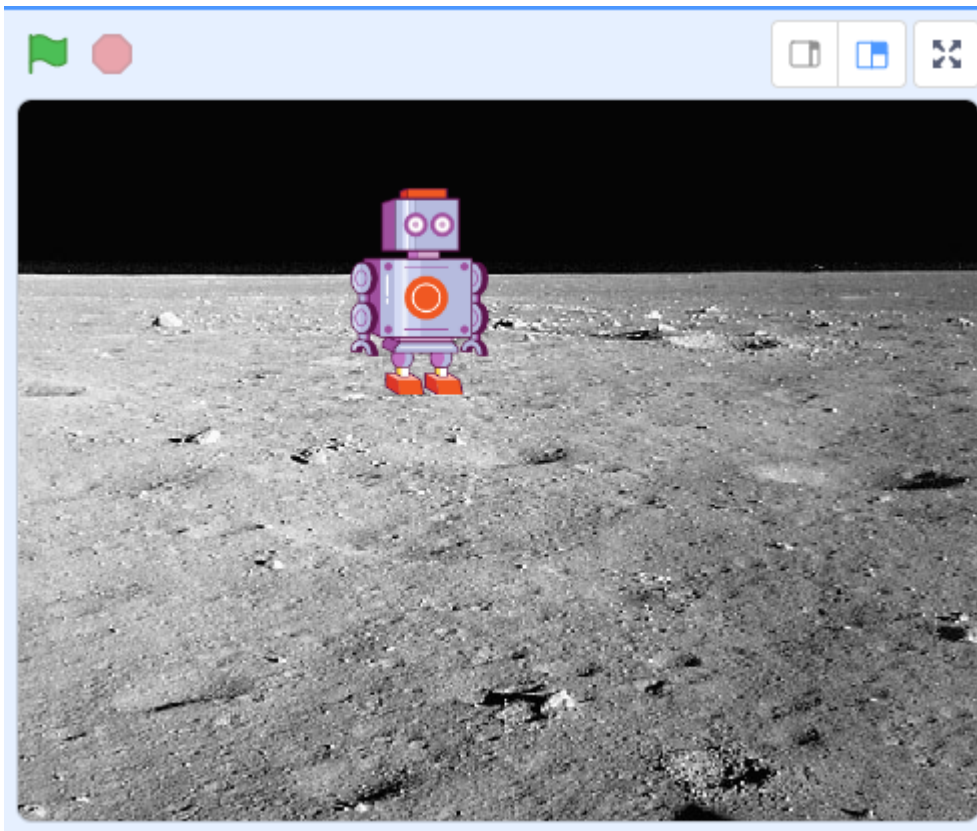
Changing the Backdrop

We now need a Backdrop for your story. We do this in a very similar way to adding new a new sprite. There is an add Backdrop icon at the bottom of the **Stage Contents**. See image below.

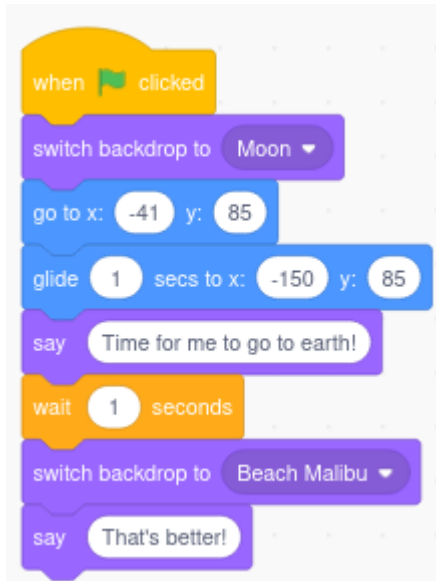


Select a Backdrop which works for your story. In this example, we've selected the moon surface image from the list that comes up. If you want to add your own Backdrop the method is the same as for adding your own sprite.

Now your **Stage** should look something like this:



Scene changes



We have seen already with motion that Scratch will remember the last thing you did – so if you move a sprite or change a Backdrop you’ll end up with that being the starting state next time you click the green flag. For this reason it’s good to initialise your Backdrop with a “switch backdrop to” block at the same time as setting the location of your sprites.

The animation script shown here does this – it sets the backdrop, moves the sprite, and then gets on with the animation.

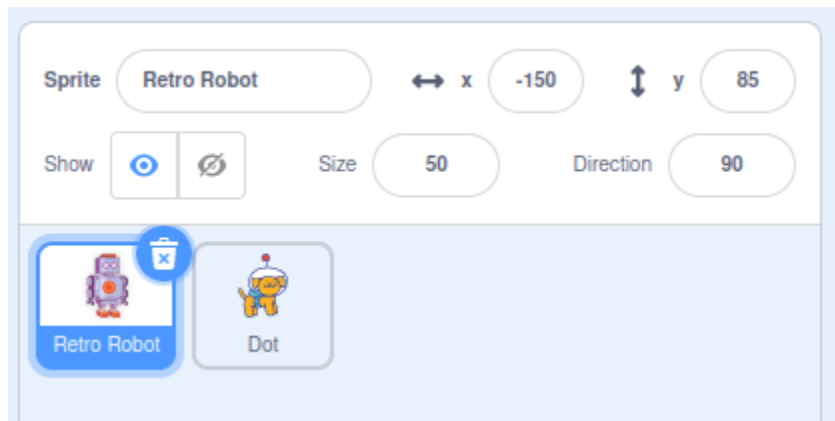
If you only have one character, this is great and probably enough to build a story where that character moves around and explores. However, most stories have more than one character, and we want these characters to talk to each other.

Animating two or more characters at the same time

Each Sprite has their own set of scripts. You can add lots of different sprites (characters) to your animation and add different scripts for each. You could think of it like a play, with each Sprite remembering their own lines and stage directions.

This means that each sprite will have their own **scripts area**. So if you have two sprites you have two completely different sets of instructions. The **sprite list** is how you switch between these – if you add more sprites, they’ll all appear in the sprite list and you can select one of them by clicking on it. This gives the active sprite a blue border, and the “delete sprite” bin icon appears.

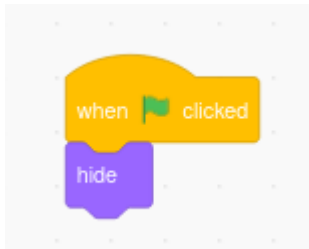
The picture below shows a sprite list containing two sprites (“Dot” and “Retro Robot”) with “Retro Robot” selected.



One thing to bear in mind is that any change of the backdrop happens to the scene, so we advise having one **main** character who's there all the time (or most of the time) and putting all the scene change code blocks in that Sprite's **scripts area**. You don't have to do this, but it does make it easier to find any errors.

If you give each sprite an action to do "When green flag clicked" then you know that all your sprites will start off at the same time.

Appearing and disappearing

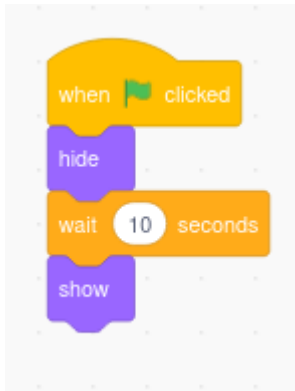


You might want particular sprites to appear or disappear in a scene. You can do this with show and hide blocks. This can be useful for sprites which you want to appear later in an animation, or it can be useful for sprites who have done their actions and take no further role in the story.

The blocks shown here hide a sprite when the green flag is clicked, which means that the sprite starts off invisible.

If you have a sprite which starts off visible then hides, you might need to begin your script with a "show" block as Scratch will remember how the sprite finished the script. If you find your sprites disappearing and not reappearing, this is something to check.

Keeping time



One way to make sprites work well together is to have them keep time. If you know that a scene lasts a particular number of seconds, you can get any sprites who have action in later scenes to appear after delaying the right number of seconds.

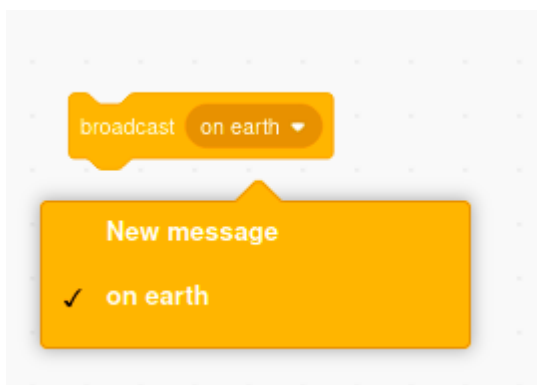
This works well if you are able to plan your animation carefully, and add up how much time things take. There are other ways to coordinate sprites, though.

Sending messages

Perhaps an easier way to make sprites work well together is to have one sprite do some stuff (say things, move around) and then **broadcast** a message which the other sprites can listen out for. You could think of this as being like listening out for a cue in a play.

On the next page is an example of this. The animation as a whole has Retro Robot starting off on the moon, moving a bit, and saying “Time for me to go to Earth!”. Then the background changes to a beach, Retro robot says “That’s better” and Dot the space dog appears jumping up and down.

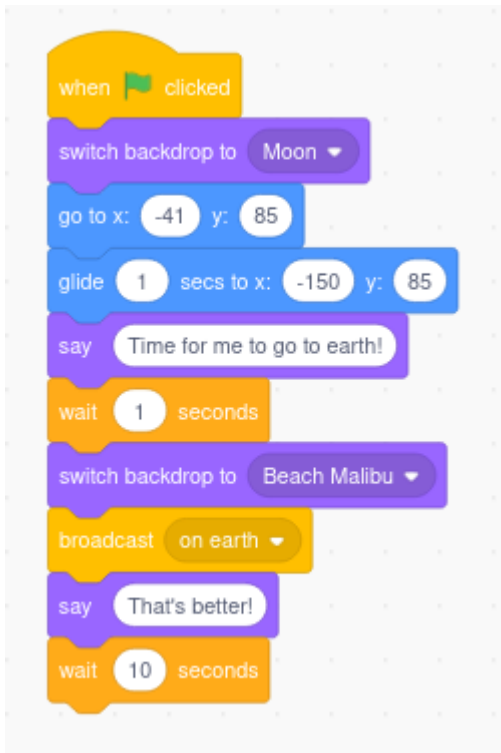
The way we do this is that the code for Retro robot sends a message when the background changes, and the code for Dot the space dog listens out for that message.



To broadcast a message, go to the Events blocks and choose a “broadcast” block. This has a drop-down menu where you can choose between messages which have been set up, or you can choose to create a New message.

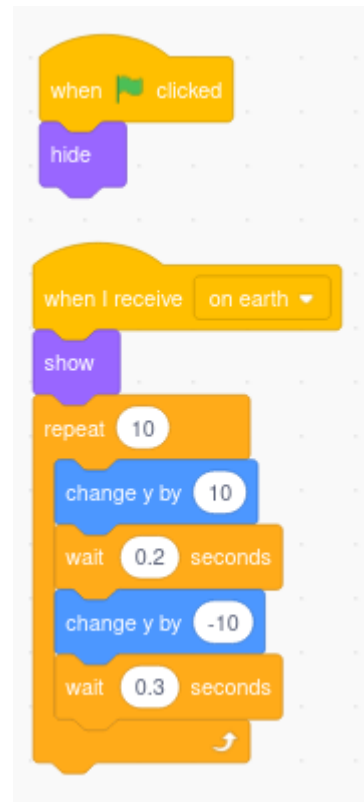
Here's the code for two sprites, with messages being passed from Retro Robot to Dot the space dog.

Retro Robot's script blocks



```
when green flag clicked
  switch backdrop to Moon
  go to x: -41 y: 85
  glide 1 secs to x: -150 y: 85
  say Time for me to go to earth!
  wait 1 seconds
  switch backdrop to Beach Malibu
  broadcast on earth
  say That's better!
  wait 10 seconds
```

Dot the space dog's script blocks



```
when green flag clicked
  hide

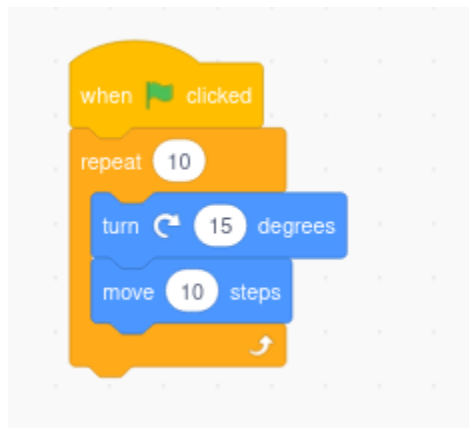
when I receive on earth
  show
  repeat 10
    change y by 10
    wait 0.2 seconds
    change y by -10
    wait 0.3 seconds
```

Making your character turn

Now you have Backdrops changing, characters moving and saying things, and interacting we can look at a couple of more advanced ideas about moving around and changing appearance.

Each Sprite remembers where it is, and what direction it is pointing in. This means you can tell sprites to “move forwards 10 steps”, and “turn 15 degrees”, and other commands for motion. We didn’t start with these, though, as they can be a bit confusing.

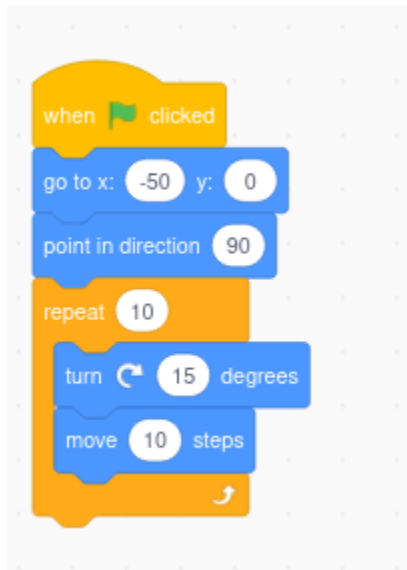
Add a sprite, and try a set of blocks like this:



When you click the green flag to test this code you’ll see the Sprite turn around. However, next time you click the green flag, the Sprite will start wherever it ended up. Experiment with different numbers in the “Repeat 10” loop, the turn, and the steps.

Because Scratch remembers the direction and location of Sprites each time it runs, it can be hard to make things repeatable. This is OK for some applications but not for animations or storytelling – when you’re building an animation, you generally want it to start in the same place each time.

This is called **initialisation** in computing and the way we do it is to explicitly state where we want things to start – we specify the initial position or value for things so that we don’t end up accidentally remembering old values and starting in the wrong place.

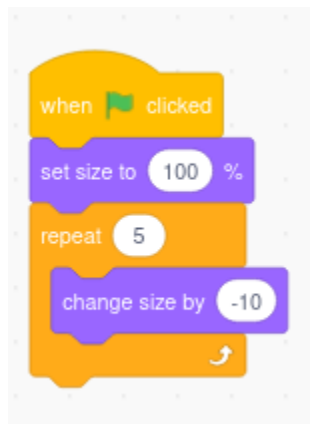


Try adding some initialisation blocks to your test of rotation – tell the sprite where you want to start with a “Go to x: y:” block, and what direction you want it to be looking with a “Point in direction” block.

Making your character shrink and grow

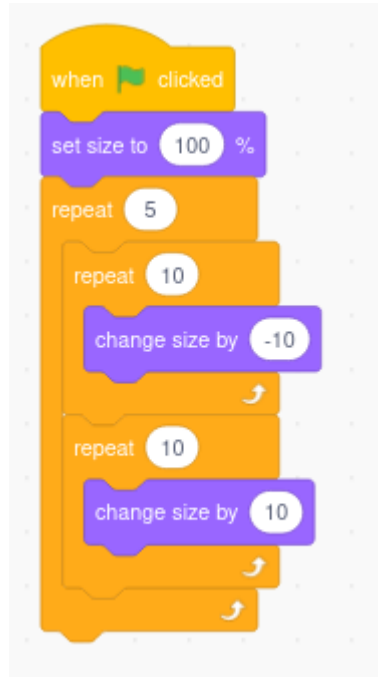
You can make your character shrink and grow in a similar way, but you have to think about initialisation in a similar way too.

Changing the size of your character comes under the “Looks” category. Try this set of blocks and see what the effect is:



People quite often use size changes in animations to make it look like a character is moving away or coming closer. (Things get smaller when they get farther away, and larger when they get closer.)

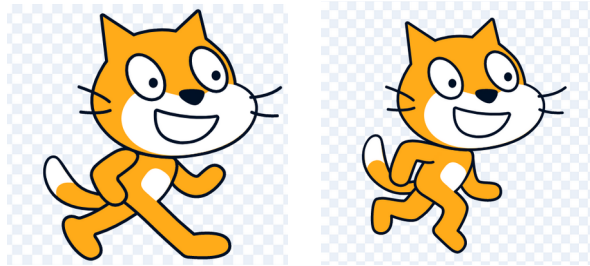
We've used some repeat loops in the last few sections to change things multiple times. To round off this section on growing and shrinking your Sprites, try out this nested repeat loop where you have a shrink (change size by a negative number) and a grow (change size by a positive number).



This might create a strange effect for a Sprite representing a character, but if you have a sprite representing a fire which moves, or some other background object that changes, these kinds of loops can be very useful.

Making your character change appearance completely

Scratch's built in sprites often have different costumes you can use to change their appearance. These might be the same creature in a different pose, or a related creature. As an example, here are the different costumes for the Cat sprite:



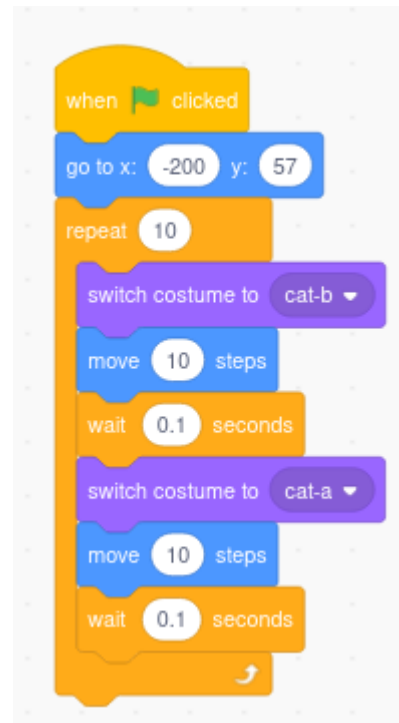
You can see that it is the same character, just slightly different in pose.

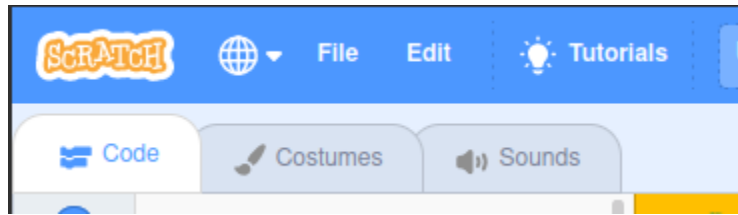
To move between costumes there's a block in the looks category: the drop-down menu in the "switch costume to:" block will change so it matches the costumes available for the current sprite.

The set of blocks shown to the left make our cat sprite animate walking (well, the cat's paw moves).

You might have noticed that there are some tabs at the top of the Scratch page. Up until now we've been using the Code tab to move things around and change Backdrops using blocks.

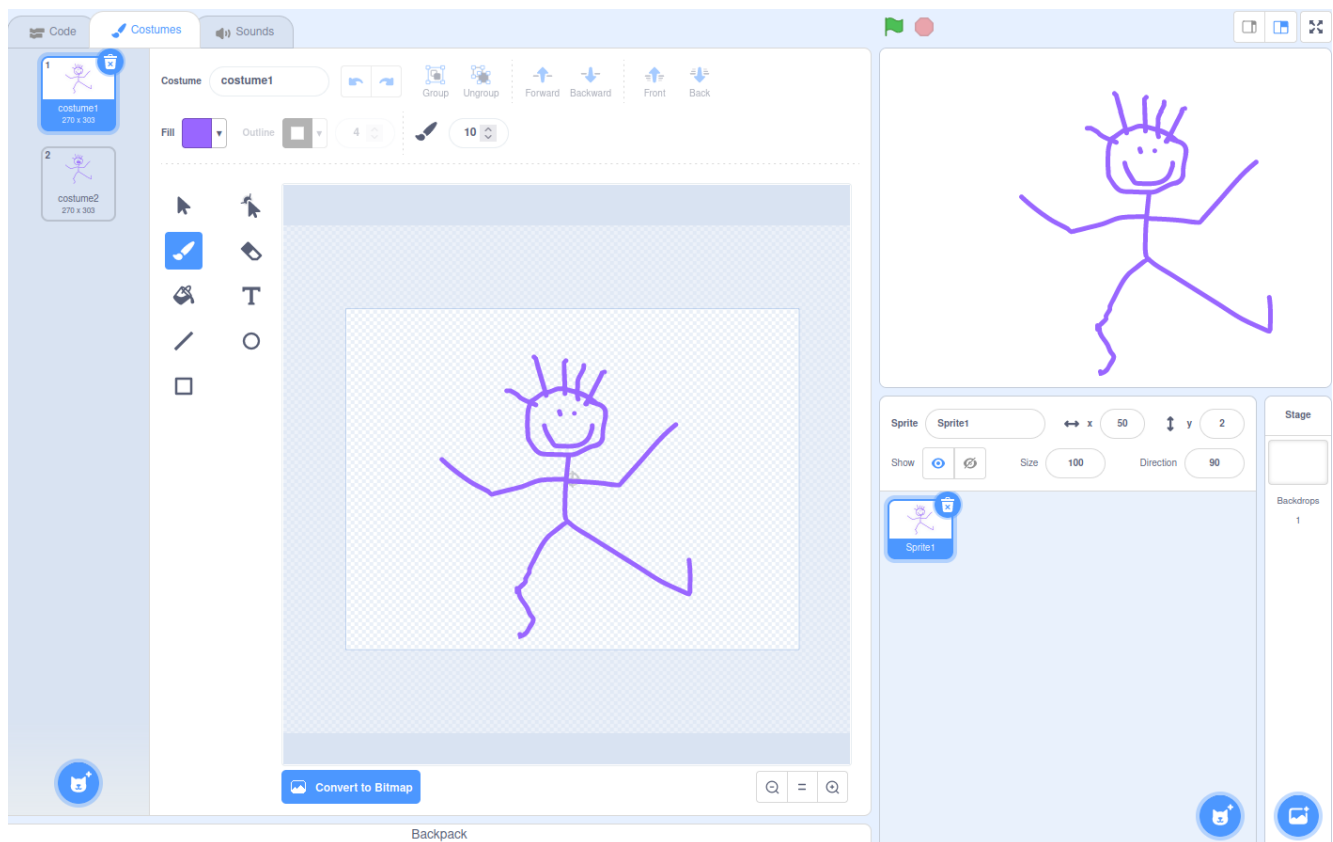
The "Costumes" tab lets you add extra costumes to a Sprite, edit Sprites, and draw your own sprites. The in-built sprites which come with Scratch often have multiple costumes. Up until now we've been thinking about each sprite as an image, but really, it can be a collection of images.





Click on “Costumes” now and take a look around.

On the left hand side of the window is a list of all the current Sprite’s costumes. If you want to make small changes to a costume, you can duplicate existing costumes by right-clicking and selecting “duplicate”, then use Scratch’s built in editor to make minor changes.



Some example animations

Finally if you would like to see some example short animations using the techniques in this book we have a few to show you.

- <https://scratch.mit.edu/projects/747885291> has a cat looking for a dinosaur
- <https://scratch.mit.edu/projects/727210103/> A hungry dragon sets fire to the Aberystwyth Town Library
- <https://scratch.mit.edu/projects/739605624/> A lonely robot goes to the beach to find its dog.

For any of these you can click “See inside” to take a look at the blocks and how they fit together.

