Sean McManus

2nd Edition

Scratch Programming

updated for Scratch 3

in easy steps

PLAIN ENGLISH

EASY TO FOLLOW

FULLY ILLUSTRATED

IN FULL COLOR

DESIGN & SHARE YOUR OWN PROGRAMS

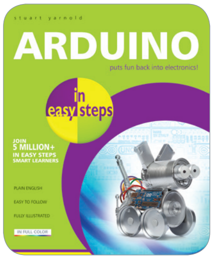
CREATE ARCADE GAMES & QUIZZES

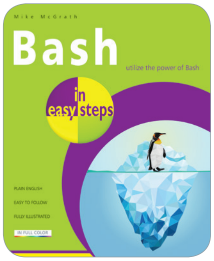
MAKE COMPUTER-GENERATED ART

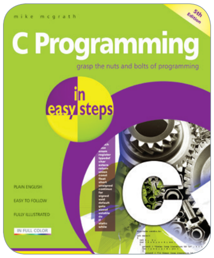
LEARN TIPS AND TRICKS

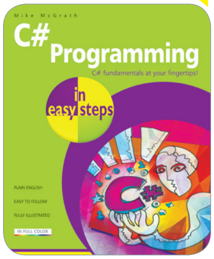
Liked this book? Learn more in easy steps

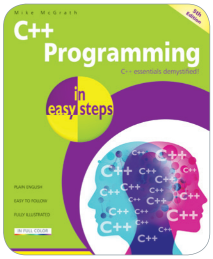
Age 9+

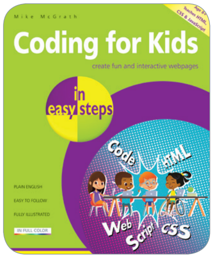


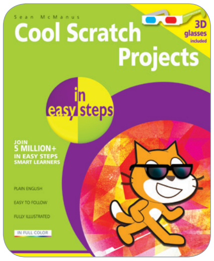


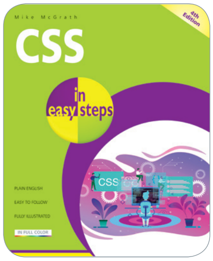


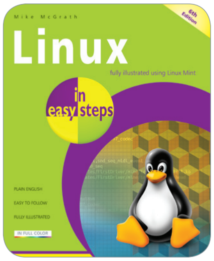


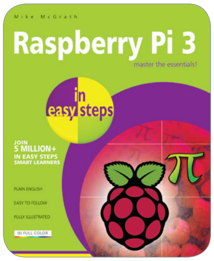


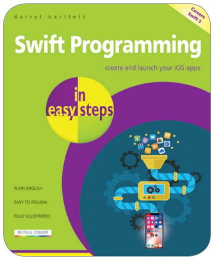


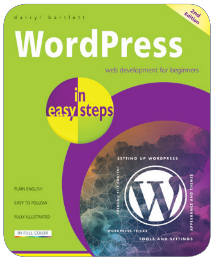










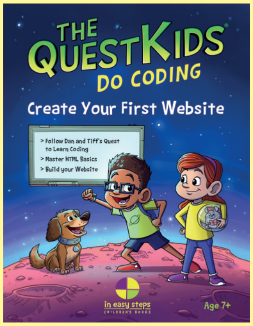


In Easy Steps ebooks
now available to download onto your computer, tablet and smartphone from
www.ineasysteps.com

kobo amazonkindle nook

Get it on Apple Books GET IT ON Google Play

For children aged 7+



With a captivating storyline, this book will teach aspiring young coders how to build their first website in a fun way, and in easy steps.

visit www.ineasysteps.com for more information on titles and updates

Sean McManus

Scratch Programming

in easy steps

Second Edition
updated for Scratch 3

In easy steps is an imprint of In Easy Steps Limited
16 Hamilton Terrace · Holly Walk · Leamington Spa
Warwickshire · United Kingdom · CV32 4LY
www.ineasysteps.com

Notice of Liability

Every effort has been made to ensure that this book contains accurate and current information. However, In Easy Steps Limited and the author shall not be liable for any loss or damage suffered by readers as a result of any information contained herein.

Trademarks

All trademarks are acknowledged as belonging to their respective companies.

In Easy Steps Limited supports The Forest Stewardship Council (FSC), the leading international forest certification organization. All our titles that are printed on Greenpeace approved FSC certified paper carry the FSC logo.

FSC
www.fsc.org
MIX
Paper from
responsible sources
FSC® C020837

Printed and bound in the United Kingdom

ISBN 978-1-84078-859-4

Contents

1 Introducing Scratch

In this chapter, you'll get started with Scratch, including meeting some of the blocks used to give instructions, and creating your first program. You'll also learn how to save your work, and load projects created by others.

What is Scratch?

Programming is the art of writing instructions to tell a computer what to do. A set of instructions is called a program. The instructions are written in what's known as a programming language, and there are thousands to choose from.

Scratch is a programming language that is perfect for making games, animations, interactive stories and other visually rich programs. It provides a great introduction to programming for people of all ages. It's widely used in schools and colleges, but Harvard University has also used it in higher education at its Summer School. I've led workshops for adults where Scratch provided a friendly introduction to the kind of creative problem solving that programmers do all the time.

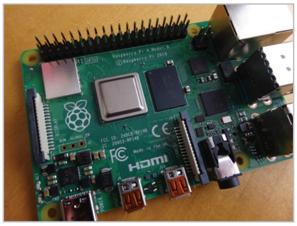
Scratch is easier to use than most other programming languages for a number of reasons:

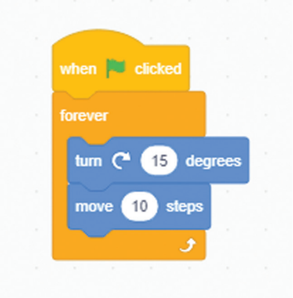
- You don't have to remember or type any commands: they're all on screen, so you can just drag and drop them.
- Commands fit together like jigsaw pieces, so there are strong visual hints about how you can combine them.
- Error messages are rare. Because Scratch commands lock together, programs usually make some kind of sense. It is possible to still write programs with logical errors in, if they don't do what you expected, but Scratch guides you to write things that work, rather than nagging you when they don't.
- The commands are color-coded and categorized, so you can easily find a command when you need it.
- The commands in Scratch simplify common activities in games, such as testing whether a missile has hit an alien (collision detection), or rotating a character on screen.

In short, Scratch is designed for your success. It enables you to quickly see results from your work, and even includes graphics and sounds you can use so you can get started right now. Many other programming languages require you to learn text commands, and have strict rules about how you can use them. Scratch doesn't. That means you can focus your energy instead on the fun stuff: dreaming up ideas for new programs, working out how to build them, designing them, and sharing them with friends.

Hot tip

Scratch doesn't cost anything and it works on Windows, Mac and Linux computers, as well as tablet devices. For the latest version of Scratch, the Raspberry Pi Foundation recommends a Raspberry Pi 4 computer (pictured below) with at least 2GB of RAM.





```
when 🏴 clicked
forever
    turn ↻ 15 degrees
    move 10 steps
```

Above: A simple Scratch program, showing how the color-coded commands lock together.

Creating a Scratch account

Before you begin to program with Scratch, I recommend you create an account for the Scratch website. Here's how:

1. Open a web browser, such as Google Chrome

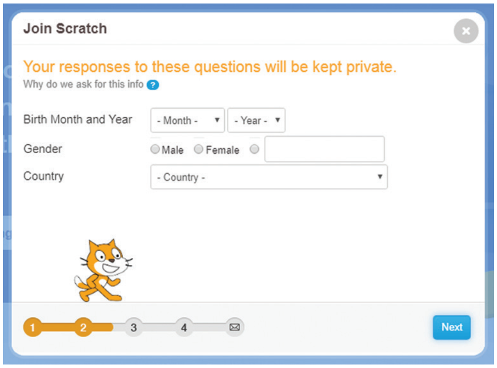
2. Enter the website address **https://scratch.mit.edu/** in your address bar, usually at the top of the screen

3. Click **Join Scratch** in the top right

4. Make up a username. The site advises you to protect your privacy by not using your real name. You can't change your username later, so choose wisely

5. Pick a password and enter it twice. The second time is to make sure you've typed it correctly. Use a mixture of uppercase and lowercase, numbers and symbols to make it more secure. Click the **Next** button

6. Enter your date of birth, gender, country and email address. This personal information is used to help you recover your password if you forget it, and is used by the Scratch team to understand who uses Scratch. Click **Next**



Join Scratch

Your responses to these questions will be kept private.
Why do we ask for this info ?

Birth Month and Year	- Month - ▾ - Year - ▾
Gender	○ Male ○ Female ○
Country	- Country - ▾

7. Enter an email address so they can confirm your account before you can share projects or post comments. Click **Next** and you will be logged in

Hot tip

The design of websites can change from time to time, so don't worry if you see variations in the sign-up process when you do it.

Hot tip

You can try Scratch by just going to the website and clicking Create at the top of the screen. If you use an account, though, the website will automatically save your work for you.

Hot tip

When you return to the site next time, you can click "Sign in" in the top right to get back to your projects.

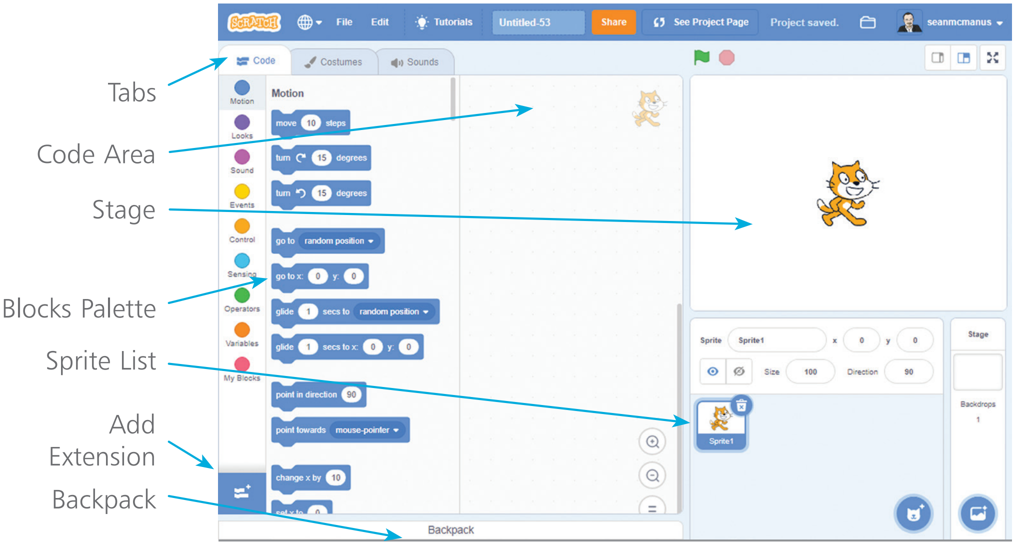
Meet Scratch 3

Scratch 3 adds new features for making projects, a streamlined user interface, and compatibility with more devices.

Scratch 3 is the latest version of Scratch, launched in January 2019. It runs in your web browser and is based on standard web technologies (HTML5). That means it is compatible with many different types of computer. Scratch 3 introduced several new features, including:

- **Tablet support**. You can now build and use projects on a tablet device, and can play (but not build) projects on a mobile phone. Games that require keyboard controls to move a character on screen may not work. You can use your tablet's onscreen keyboard to type information into a project, though.
- **New extensions**. There are new capabilities to work with the micro:bit and Sense HAT devices, to speak out loud, to translate text, and to work with the Lego WeDo 2.0 and Lego Mindstorms EV3 kits. To add an extension, click the button in the bottom left of the screen.
- **Simplified Blocks Palette**. The menu of instructions has been streamlined by moving the music and drawing (Pen) blocks into the extensions section. If you're familiar with Scratch and are wondering where they went, you can find them there!

Older versions of Scratch (Scratch 2, and Scratch 1.4) are still available to download. There's an offline version of Scratch 3 too. Find them all at **https://scratch.mit.edu/download**



Using the Scratch screen

The main parts of the Scratch screen, as shown on the facing page, are:

- **Stage**: This is where you can see your animations and games in action. When Scratch starts, there's a large orange cat in the middle of the Stage.
- **Sprite List**: The cat is a "sprite", which is like a character or object in a game. Your project might include lots of sprites, such as the player's spaceship, invading aliens and a missile. In the Sprite List, you can see all the sprites that are in your project, and click them to switch between them. The Sprite List is underneath the Stage.
- **Blocks Palette**: In Scratch, you give the computer commands by using blocks, which are instructions that fit together like jigsaw pieces. The Blocks Palette presents you with all the blocks you can use. When you start Scratch, you can see the Motion blocks, which are color-coded in dark blue, and are used for moving sprites around the Stage. You can browse a different set of blocks in the Blocks Palette by clicking one of the buttons, such as the **Looks** button or the **Sound** button. You can also scroll through all the blocks in the different categories in the Blocks Palette.
- **Code Area**: The Code Area (previously known as the Scripts Area in older versions of Scratch) is where you make your programs, by joining instruction blocks together. When you join blocks together, the result is called a script. The Code Area expands to fill the screen space available, so if you use a larger monitor, the Code Area will be bigger.
- **Backpack**: The Backpack at the bottom of the screen is handy for copying sprites and scripts between different projects. Click the Backpack at the bottom of the screen to open it. You can then drag sprites or scripts into it, or drag them out into your current project.

The blue stripe across the top of the screen is used for managing your projects. You can use the File menu to save your work, including downloading it to your computer. The online version of Scratch saves your updates automatically, so you won't need this often. The box containing "Untitled" is where you rename your project. Click the folder in the top right to find all your projects.

Don't forget

You'll see all these elements in action soon, so don't worry about memorizing the screen layout. This section is just to help you get your bearings. Remember that you can refer to the screenshot on the facing page at any time.

Hot tip

Use the tabs above the Blocks Palette to switch between the scripts, costumes and sounds on a sprite.

Using Scratch on tablets

You can now use Scratch on a tablet, such as an Apple iPad. Simply visit the Scratch website in your browser. You may find that Scratch runs more slowly than it does on a PC, but you should still be able to enjoy building and using many Scratch projects. If you're using a tablet, make sure you log in to the Scratch website, because you can't save projects to your device.

You can't use projects that require the keyboard, such as games that use keys to move a sprite. If the project includes **key [space] pressed?** or **when [space] key pressed** blocks (to detect any key), the project is unlikely to work well without a real keyboard. From this book, you can't play games like Super Dodgeball or Shop Cat (although they are fine to build). But you can play Spiral Rider, Space Opera, Quiz Break, Evil Robot, Photo Safari and more. I'll warn you at the start of a chapter if a project won't work on a touch-based device.

Using gestures

You'll find the touch gestures feel natural and are easy to remember. Here's how to use a touch-based device instead of a mouse:

- To "click" something, **tap** it on the screen. For example, tap to use a button or start a script.
- To "right-click" something, **tap and hold** it on the screen – for example, to duplicate a sprite in the Sprite List.
- To "hover" over a button, simply **tap** it – for example, to open the menu to add a sprite.
- You can "drag" blocks and sprites by touching them and **dragging your finger** across the screen.
- To scroll through the Blocks Palette, **swipe** it up or down.
- To move around the Code Area, **drag** its background up, down, left or right.
- To scroll in the Paint Editor (see Chapter 4), **drag the scrollbars** at the bottom and side of the canvas.
- To change the values in a block, or type something in response to an **ask** block on the Stage, **tap the box to make your virtual keyboard appear**.

NEW

With the launch of Scratch 3, touch devices are more widely supported. The previous version of Scratch was based on Flash, which didn't work on iPads. The latest version has redesigned blocks that are easier move by hand.

Hot tip

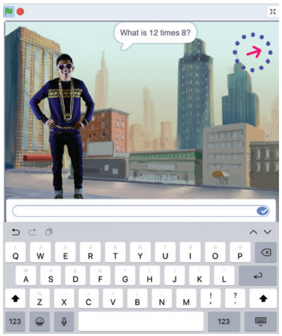
If you have a PC with a touchscreen, you might want to use the keyboard and touchscreen together to create your projects. Some people find it easier to use the touchscreen than the mouse.

Scrolling around the Scratch screen

If Scratch doesn't fit in your tablet's screen, you can scroll left and right by dragging the blue bar at the top. To scroll up and down, drag part of the light blue background – for example, an empty space in the Sprite List.

Using Scratch in portrait mode

If you want to see all of the Stage and the keyboard at the same time – for example, to play Quiz Break in this book – use your tablet in portrait orientation.

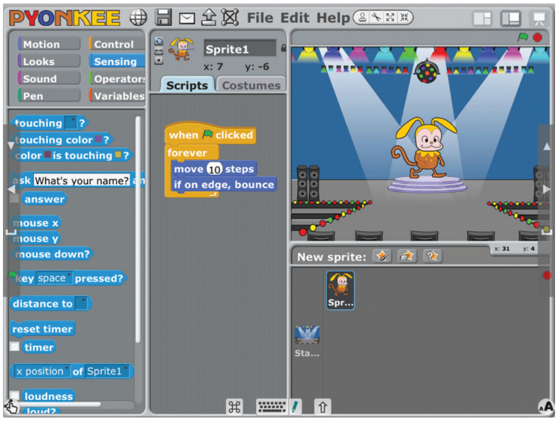


Don't forget

In the rest of this book, I'll mostly assume readers are using a keyboard and mouse, so refer back here if you need a reminder of how to use your touchscreen.

Using Scratch apps

At the time of writing, MIT is creating apps for Scratch 3. There is an unofficial app based on Scratch 1.4, called Pyonkee. It's a fun way to tinker with Scratch, but it doesn't have the cat or certain other sprites, and doesn't have newer Scratch features such as sprite cloning, extensions, and sound effect blocks.



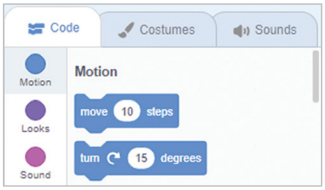
Hot tip

MIT has also created ScratchJr, a reinvention of Scratch for younger children, designed for iPads and Android tablets. For tips on using ScratchJr, see our book Cool Scratch Projects in easy steps.

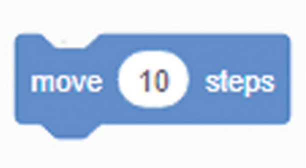
Exploring the blocks

Try experimenting with a few blocks to see what they do:

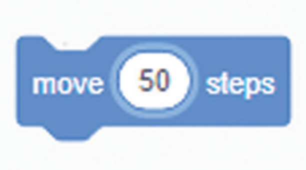
1. Click the round, blue **Motion** button to the left of the Blocks Palette to show the Motion blocks. This button is selected when you first start Scratch



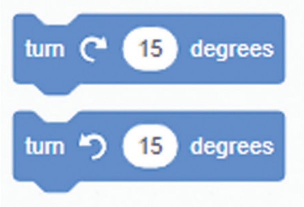
2. In the Blocks Palette, click (or tap) the **move 10 steps** block. The cat on the Stage moves in the direction it's facing, to the right. Each time you click the block, the cat moves once. This block only changes the cat's position, though: you won't see its legs move



3. The number of steps is how far across the screen you want the cat to move. Click the number 10 and change it to something else. Try 50 and when you click the block, the cat moves five times as far. Whenever you see a white hole in a block, you can change what's in it



4. Rotate the cat by clicking the **turn clockwise 15 degrees** block. To change the angle of the turn, change the number. Remember to click the block to actually make the cat turn. When you click the **move 10 steps** block next time, the cat walks in its new direction



Hot tip

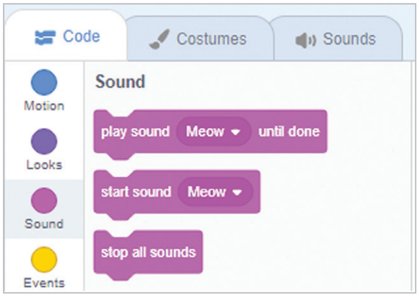
Scratch won't let the cat disappear off the screen completely. If you use numbers that are too big, the sprite will stay at the edge of the screen.

Above: The cat on the Stage, after I clicked "turn clockwise 15 degrees".

5. If the cat gets to the edge of the Stage, drag it back again with your mouse pointer. Click the cat, hold the mouse button down, move the cat, and then release the mouse button to drop it in place

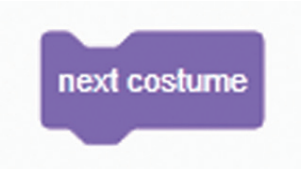
6. Click the round, pink **Sound** button beside the Blocks Palette

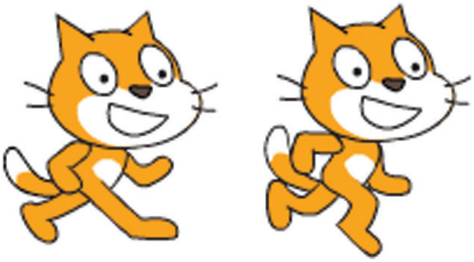
7. Click the **play sound Meow until done** block. You should hear the cat make a meow sound if you have speakers on



8. Click the round, purple **Looks** button. It's beside the Blocks Palette. You can see it in the picture above

9. Click the **next costume** block to see the cat's legs move, so it appears to run on the spot. Costumes are different pictures a sprite can have, and the cat has two that show its legs in different positions (see below). In Scratch, the sprites are like the actors; they wear the costumes (different images), and they move on the Stage





Hot tip

All these blocks are explained later in the book, but for now it's worth spending a few minutes exploring some of the commands you can give the cat.

NEW

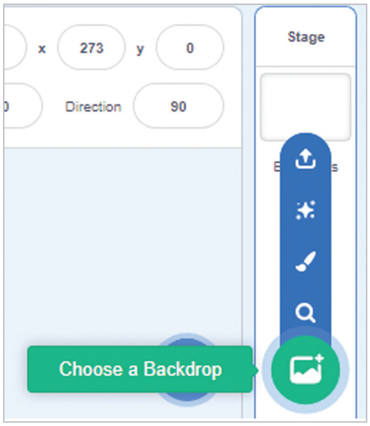
Scratch 3 includes new blocks to change the pitch of the sound. Try changing the numbers in the "change [pitch] effect by 10" block, clicking it, and then clicking the "play sound Meow until done" block again.

Don't forget

If you're using a tablet, you can tap things instead of clicking them.

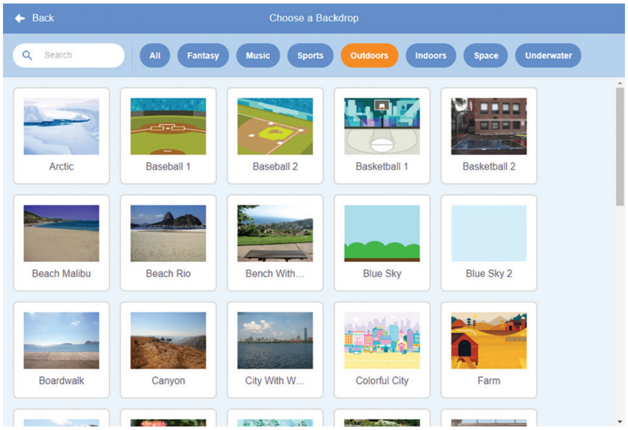
Changing the backdrop

Before we make our first program, let’s change the background of the Stage to something more inspiring. The background is called a backdrop in Scratch.



1 To the right of the Sprite List, there is a panel for the Stage. At the bottom, hover over the round button to open the backdrop options. Use the magnifying glass icon to choose a backdrop from the library. (You can also just click the round button to choose from the library.) The other options enable you to paint a backdrop, get a surprise backdrop, or upload a picture from your computer

2 When the library opens, click the categories at the top to view the different backdrops available, and use the scrollbar on the right to see more designs. Select the Outdoors category



3 Scroll to find, and then select, the Hill image

Hot tip

Before you begin building each program in this book, it’s a good idea to start a new project. Click the File menu and then click New. If you’re using Scratch 3 and you can’t see the File menu, click Create instead at the top of the screen.

NEW

Scratch 3 adds some new backdrops, including some great space cities, illustrated sports fields, and a Jurassic skyline.

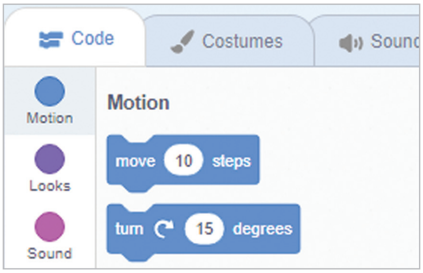
Creating your first program

When you click blocks in the Blocks Palette, the cat moves immediately, so this is good for testing what blocks do, but not useful for making a program. A program is a set of repeatable instructions that you can store up to carry out later. For our first Scratch program, let's make the cat walk down the hill:

1. We're going to write a program for the cat, so click the cat in the Sprite List

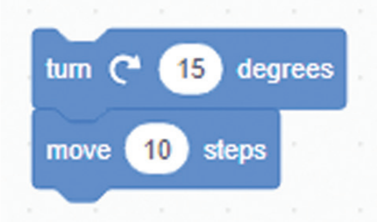
2. Next, check that the Code Area is open. If the middle pane shows costumes or sounds instead, click the **Code** tab above the Blocks Palette

3. Click the **Motion** button to the left of the Blocks Palette (see screenshot, right)



4. You make a program by dragging blocks into the Code Area from the Blocks Palette. To do this for your first block, click the **turn clockwise 15 degrees** block in the Blocks Palette, hold the mouse button down, move the mouse pointer into the Code Area and then release the mouse button. This first block will point our cat downhill, ready for its walk

5. Click the **move 10 steps** block in the Blocks Palette, drag it into the Code Area and drop it underneath the **turn clockwise 15 degrees** block. They will snap together. When blocks are joined like this they make what's known as a "script". A sprite can have more than one script, and a program might include lots of sprites with several scripts



```
turn ↻ 15 degrees
move 10 steps
```

Hot tip

If you click and drag a block that's joined to other blocks above it in the Code Area, it will break away from them, and carry all the blocks underneath it with it.

Beware

You can move blocks around the Code Area, but if you drag them into the Blocks Palette, they'll be deleted.

Don't forget

Make sure the blocks snap together, otherwise they won't work as one script. If they don't snap together, they're not close enough.

...cont'd

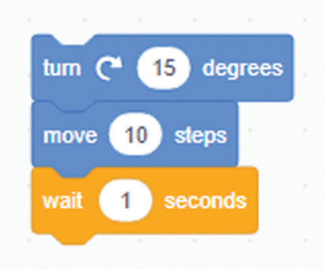
Beware

I wouldn't usually recommend you add the same blocks repeatedly, but we're right at the beginning of learning Scratch here. There is a more readable and elegant solution you'll discover in Chapter 2.

Hot tip

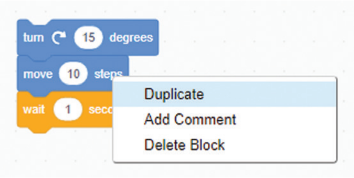
To right-click on a touchscreen device, tap and hold. When you duplicate a script, the copy is placed in the Code Area. You can drag it to the bottom of your script.

6 Click the **Control** button beside the Blocks Palette. Control blocks are used to decide when things should happen. Drag the **wait 1 seconds** block into the Code Area and snap it underneath the other two blocks. This block adds a 1-second delay. Without it, our cat would move so fast, it would appear to just jump from the start of its walk to the end. Slowing it down enables us to see what's going on. You can make it walk a bit faster by changing the delay from 1 second to 0.5 seconds



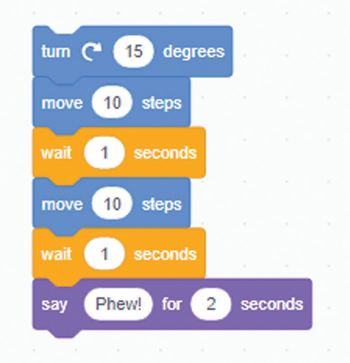
```
turn ↻ 15 degrees
move 10 steps
wait 1 seconds
```

7 Right-click the **move 10 steps** block, and when the menu opens, choose **Duplicate**. This copies the block plus any blocks underneath it in your script. In our example, it copies the **move** and the **wait** blocks. Move the copy to the bottom of your program, and click to place the blocks there. You can repeat this step several times to make the cat walk further



```
turn ↻ 15 degrees
move 10 steps
wait 1 secc
```

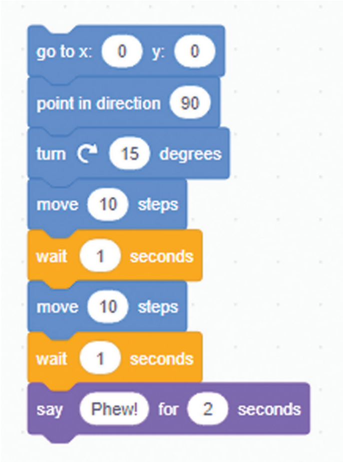
8 It's a long walk for a tiny cat, so let's make it finish its walk with a exclamation of "Phew!" in a speech bubble. Click the **Looks** button beside the Blocks Palette. Drag the **say Hello! for 2 seconds** block into the Code Area and join it to your program. Click Hello! to edit what the cat says to Phew!



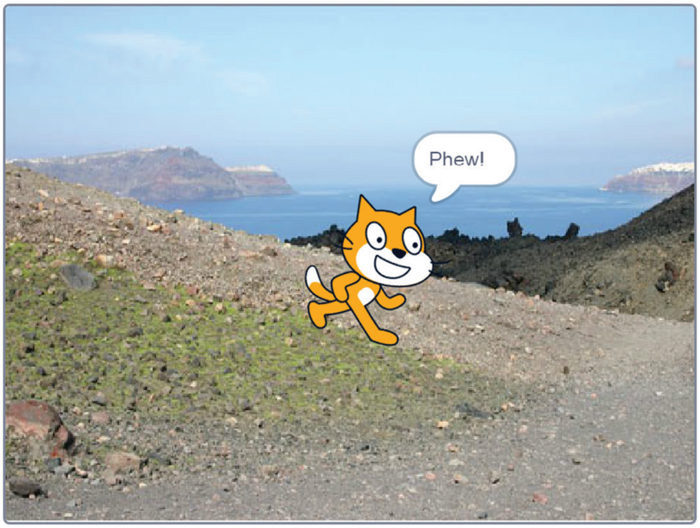
```
turn ↻ 15 degrees
move 10 steps
wait 1 seconds
move 10 steps
wait 1 seconds
say Phew! for 2 seconds
```

9 When you start a script's commands, it's called "running" the script. To run your script, click any of the joined-up blocks in the Code Area. Scratch carries out all the joined-up instructions in order, starting at the top and working its way down the blocks

10 What happens if you click the script to run it again? The cat turns again and walks from where it finished last time. Eventually, it'll be walking on its head. Let's add some blocks to put it in the right starting position. Click the **Motion** button beside the Blocks Palette and drag in the **point in direction 90** block and the **go to x:0 y:0** block. If the **go to** block has different numbers in it, edit them both to make them zero. Add these blocks to the top of your script. Can you make the cat's walk longer?



```
go to x: 0 y: 0
point in direction 90
turn ↻ 15 degrees
move 10 steps
wait 1 seconds
move 10 steps
wait 1 seconds
say Phew! for 2 seconds
```



Hot tip

You can insert blocks into an existing script. Instead of dropping a block at the end of your script, drag your block over it. The blocks in the script will make room to let the new blocks in. When the blocks are in the right place, release the mouse button or your finger.

Hot tip

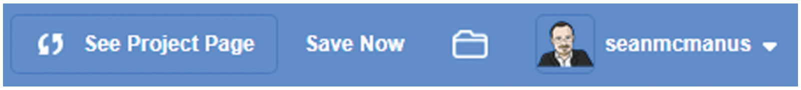
Experiment! If you use a negative number of steps in the "move 10 steps" block, the sprite moves backwards. Why not see if you can make the cat walk backwards up the hill again?

Left: A few blocks and one exhausted cat later, here's your first program.

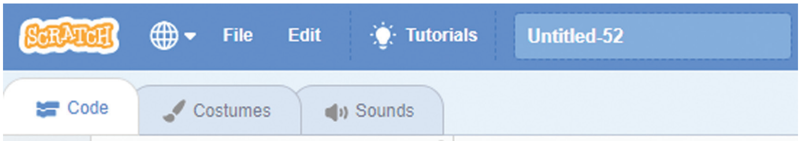
Saving your project

A Scratch project includes all the sprites, scripts and backdrops that are used in it. It's a good idea to save your projects so you can come back to them later to reuse them or modify them.

In Scratch 3 online, your work is automatically saved for you. If your latest changes have not been saved, there will be a link to **Save Now** in the blue bar at the top of the screen, beside the folder icon that gives you access to your projects.



Your project is saved with the name Untitled plus a number. You can choose a more useful name by editing the box in the blue bar, above the Code Area, and to the right of the tabs.



There are additional options for saving your work in the File menu. You can find it above the Blocks Palette, as shown above. These options include:

- **Save as a copy**: This makes a copy of your project with a new name. The previously saved version of your project is left untouched. Use this if you want to experiment with your project without losing a working version of it.
- **Save to your computer**: This enables you to save your project as a file on your computer. If your internet connection fails, use this option straight away to save your work!
- **Load from your computer**: If you previously downloaded a Scratch project to your PC, or used a downloadable version of Scratch to create it, use this option to upload it to the Scratch website, so you can edit and share it there.

Beware

You can't use your projects in an older version of Scratch than you used to create them. You can use projects with a newer version, though.

Hot tip

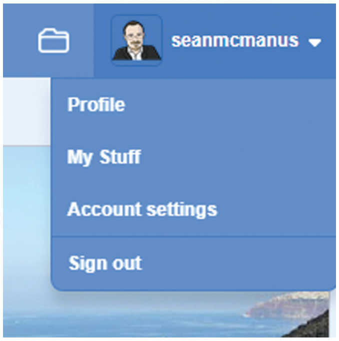
In the downloadable versions of Scratch, your work isn't automatically saved for you. Use the Save to your computer option in the File menu to save your work.

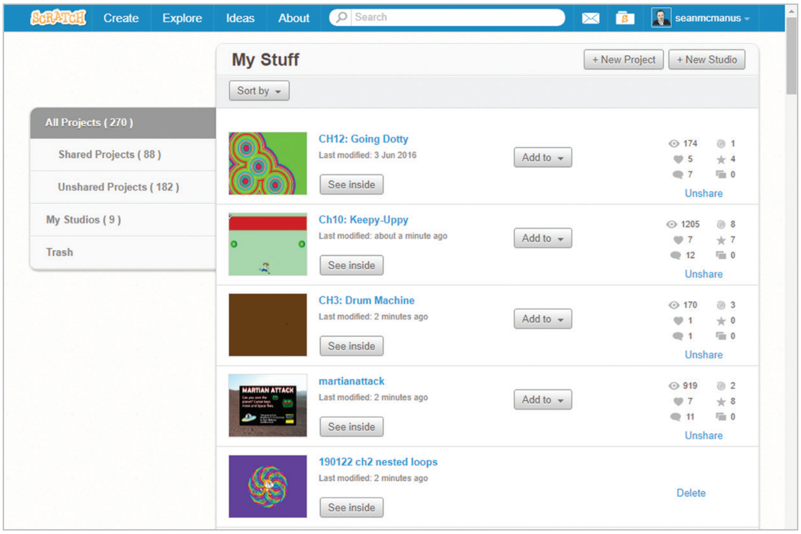
Opening projects

To find your projects on the Scratch 3 website, click the folder icon beside your username in the top right of the screen. You can also click your username and then click **My Stuff**.

If it says “Sign in” in place of your username, click it to sign in first.

The My Stuff section shows all your projects, with those you most recently edited nearer the top. You can see some of my projects in the screenshot below:





My Stuff

- All Projects (270)
- Shared Projects (88)
- Unshared Projects (182)
- My Studios (9)
- Trash

CH12: Going Dotty
Last modified: 3 Jun 2016
174 1 5 4 7 0

Ch10: Keepy-Uppy
Last modified: about a minute ago
1205 8 7 7 12 0

CH3: Drum Machine
Last modified: 2 minutes ago
170 3 1 0 1 0

martianattack
Last modified: 2 minutes ago
919 2 7 8 11 0

190122 ch2 nested loops
Last modified: 2 minutes ago

Use the scrollbar at the right edge of your browser window to find more projects and click the **Load More** button when you get to the bottom of the list. To open a project for editing, click its **See inside** button. To simply run the project, click its screenshot.

Because projects are saved automatically, your My Stuff area quickly fills up with untitled projects. To tidy up, delete unwanted projects by clicking their **Delete** links on the right. If you delete a project by mistake, click the Trash folder on the left, and then click the **Put back** button to recover the project. You can’t delete a shared project without unsharing it first (see page 209 in Chapter 11).

Beware

If you click the Share button on a project, it will be available for anyone to see, use and reuse. See Chapter 11 for more on sharing your projects.

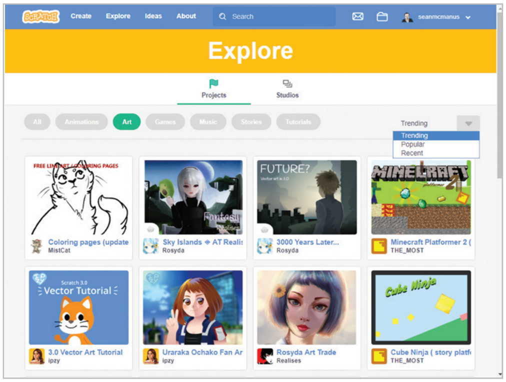
Hot tip

In downloaded versions of Scratch, you open projects by clicking to open the File menu at the top of the screen and then choosing Open.

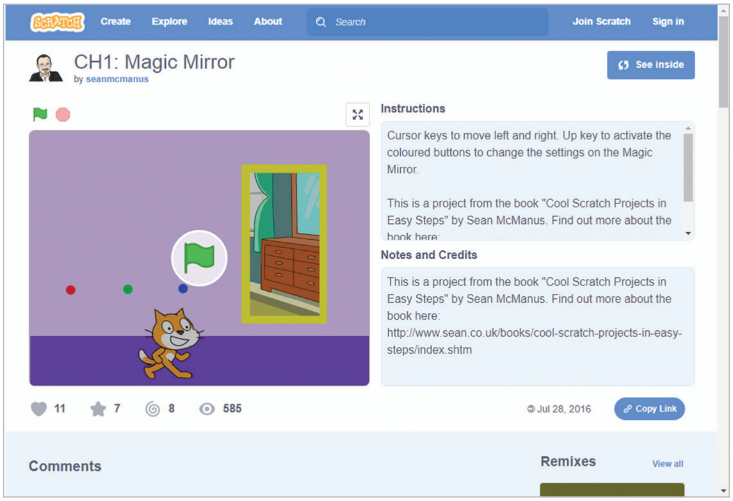
Opening shared projects

You can open the projects that other people have shared on the Scratch website too. Visit the website at **https://scratch.mit.edu** and click **Explore** at the top of the screen.

Use the buttons to choose a category of projects to explore. The triangle to the right of the project categories opens a menu that you can use to find projects that are trending, popular, or recent.



Click a project to go to its page. The instructions on the right tell you how to use the project, and you click the **green flag** button to run it, either the one in the middle of the project or the one above the project's display. If you like what you see, and you want to know how it was done, click the **See inside** button in the top right to go into the editor and see the code.



CH1: Magic Mirror

by seanmcmanus

Instructions

Cursor keys to move left and right. Up key to activate the coloured buttons to change the settings on the Magic Mirror.

This is a project from the book "Cool Scratch Projects in Easy Steps" by Sean McManus. Find out more about the book here:

Notes and Credits

This is a project from the book "Cool Scratch Projects in Easy Steps" by Sean McManus. Find out more about the book here: http://www.sean.co.uk/books/cool-scratch-projects-in-easy-steps/index.shtm

© Jul 28, 2016

Comments

Remixes

Hot tip

If you are logged in, you can click the Remix button to create your own version of a project shared on the Scratch website. All the projects on the Scratch website are shared on the understanding that others can learn from them, and create adaptations of them. You can find a Remix button on the project's page, and another in the blue bar at the top of the screen when you look inside the project.

2 Drawing with Scratch

Find your bearings on the Scratch Stage, learn how to move sprites and draw with them, and discover how loops make it easy to repeat parts of your program. You'll also make an interactive art program called Rainbow Painter.

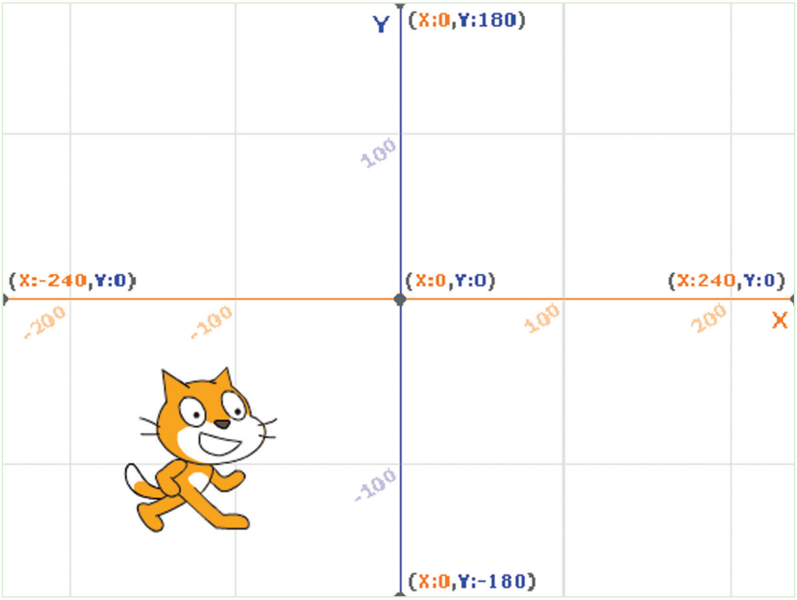
Understanding coordinates

You can use the sprites in Scratch to draw on the Stage. It's a great way to familiarize yourself with how to move sprites, and the technique can be used to create designs that your sprites can interact with, as you'll see when we make a game in Chapter 3.

First, let's take a look at how you position sprites on the Stage. Each position on the Stage has a grid reference, similar to those used on maps. The position across the Stage horizontally is called x, and the position up or down the Stage is called y.

When you start a new project, the cat is in the middle of the Stage, and this position has the grid reference x=0 and y=0.

Here's a map of the grid references on the Stage, after I've moved the cat to the bottom-left quarter of the Stage:



As you can see, the numbers for the y coordinate run from -180 at the bottom of the Stage, up to +180 at the top of the Stage.

The numbers for the x coordinate run from -240 at the left edge of the Stage, up to +240 at the right edge of the Stage.

Hot tip

You can click and drag your sprites around the Stage to reposition them while you're writing your programs. It's a good idea to use the "go to x:0 y:0" block to position them in the program, though, otherwise they might start off somewhere unexpected. If your sprite moves in the program, it won't be in the same position the second time the program is used. Other people can experiment with your program too, and might move sprites around before it starts.

Don't forget

An easy way to remember which way around x and y should be is that "x is a cross" (and "across").

3 Spiral Rider

Learn how to make your first game, including using the green flag to start scripts, using variables to store a number, moving a sprite with the keyboard or touchscreen, and letting Scratch make simple decisions about which blocks should run.

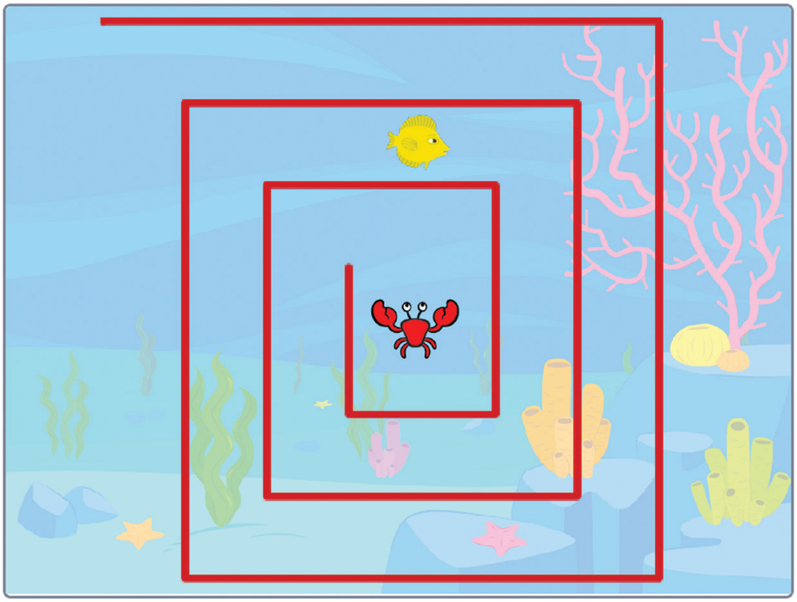
Introducing Spiral Rider

Poor Freddy Fish has bought a new T-shirt and the tag in the back is really itching. Chris the Crab has claws that could clip the tag off, if only the pair could meet up. Chris has crawled through a maze of passages in the rock. Can you help Freddy get there?

Behind this silly premise lies a game that will use what you have already learned about Scratch and take it further to help you make your first real game. You'll learn how to:

- Keep track of a number, so you can make a square spiral with ever decreasing line lengths.
- Use multiple sprites, and use the green flag to synchronize them.
- Move sprites with the keyboard or using a touch control on a tablet device.
- Make a sprite move automatically.
- Detect when a sprite hits something.

In this game, you control the fish using the arrow keys or by tapping it to make it turn. You must turn at the right time on each corner to navigate the spiral successfully.



Hot tip

If you can't get the examples to work, or if you want to take a shortcut, you can find them online. Visit the author's profile on the Scratch website at **https://scratch.mit.edu/users/seanmcmanus/** or follow the link at **https://www.sean.co.uk/scratch/** or at www.ineasysteps.com/resources/download

Hot tip

You can play this game on a tablet. We'll add a special touch control so you can turn by tapping the fish.

4 Super Dodgeball

In this chapter, you'll make a game where you have to dodge bouncing balls and collect ice creams. As you build the game, you'll learn a new way to move sprites, how to make random numbers and use them to add suspense to a game, how to paint your own sprites, and how to clone sprites.

Introducing Super Dodgeball

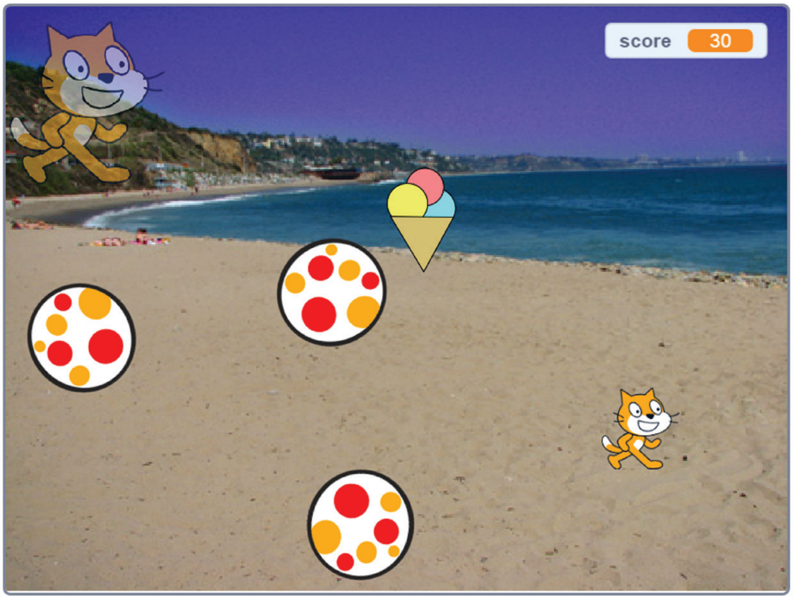
It can be hard to relax at the seaside with so many other people around, playing games and getting in the way. In Super Dodgeball, you have to avoid all the beachballs bouncing around.

Each time a ball hits you, your strength is sapped. Your energy is shown with a semi-transparent picture of your cat character, which shrinks as your strength decreases.

You score by collecting ice creams that pop up on the screen, but they soon disappear again if you're not quick enough. Ice cream also gives you a small dose of energy, to help you recover from being hit by the beachballs.

In this chapter, you'll learn how to:

- Move sprites in a new way, under keyboard control.
- Design your own sprites.
- Add randomness to your game.
- Copy and clone sprites.
- Add sound effects.
- Adjust the game's difficulty.

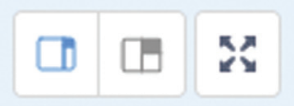


Beware

This game requires keyboard controls to play, so you can't play it on tablet devices.

Hot tip

If you need more room in the Code Area, reduce the size of the Stage. Above the Stage are three buttons. The first makes the Stage small, the second makes it normal size, and the third one is for full screen.



5 Space Opera

In this chapter, you get to conduct an alien opera performance, triggering samples and musical notes. You'll learn how to add sound effects and musical notes to your projects. You'll also learn how you can use Scratch to compose music or play your favorite tunes.

Introducing Space Opera

In Space Opera, you conduct an operatic alien singer, and can jam along on the keyboards, guitar and drum. You use the top row of letter keys (from Q to I) to control which notes the guitar plays, and click the other instruments to play them. The keyboard note varies depending on which part of the keyboard you click, with low notes on the left and high notes on the right. The singer sings a lower note than the previous one if you click them on their left, and a higher note than the previous one if you click them on their right. You can make up tunes by clicking on their left and right hands in different patterns.

Beware

This project mostly works fine on touch devices. The guitar needs a computer keyboard, though, so you might prefer to leave the guitar out if you don't have a computer keyboard.

This project teaches you some important Scratch techniques:

- Two different ways to make music, by playing short recordings, and by playing notes using simulated instruments.
- How to use graphic effects to change a sprite's appearance.
- How to animate a sprite using multiple costumes.

Now that you've completed several Scratch projects, you're an expert on the basics. In this chapter, I'll assume that you know how to add blocks to a script, and where to find the blocks you've used often before, such as the **when green flag clicked** block, and the **go to x:0 y:0** block.

6 Quiz Break

Can you beat the clock? In Quiz Break, you are challenged to answer 10 maths questions in an average of three seconds each. If you do it, the quizmaster dances. In this project, you'll learn how to use the timer, ask the player questions, and join text in speech bubbles.

Introducing Quiz Break

Hot tip

You can play this game using a tablet. I recommend using portrait orientation. Tap the answer box to see your keyboard. You can adjust the difficulty if you find it too hard to answer quickly on a tablet.

In recent years, games based on speed mathematics have become popular on handheld consoles and phones. They're a good warm-up for your brain, as well as being itchingly, frustratingly good fun.

In the game Quiz Break, you're challenged to answer simple mathematics questions against the clock. If you can answer 10 questions quickly enough, your host will be so impressed he'll dance for you. You don't get that on TV quiz shows!

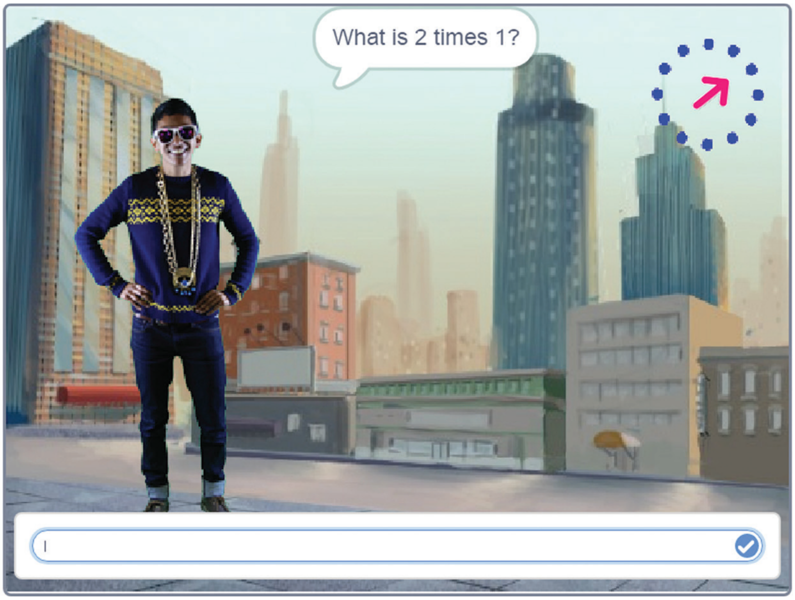
In this chapter, you'll learn how to:

- Ask the player questions.
- Use a timer to measure how long the player is taking.
- Use broadcasts to coordinate actions between sprites.
- Use the Operators blocks to perform calculations.
- Join pieces of text together so you can make sprites say more complex sentences in their speech bubbles.

To complete this project, you'll draw on what you've learned in previous chapters, including using the pen (see Chapter 2), variables (see Chapter 3), random numbers (see Chapter 4), and multiple costumes (see Chapter 5).

Hot tip

This game is set up to test your multiplication tables, but you can easily make it test addition, subtraction or division by changing the Operator blocks used.



What is 2 times 1?

7 Evil Robot

Create a version of the classic word game Hangman, and you'll learn how to structure more sophisticated projects using your own blocks, how to use lists to manage information, and how to write on the Stage with a sprite. You can personalize this game with your own word lists and your choice of cartoon host.

Hot tip

You can play this game using a tablet. I recommend using portrait orientation.

Introducing Evil Robot

In this traditional game, commonly known as Hangman, you have to guess the letters that could be in a mystery word. If you guess correctly, you're shown where your guessed letter appears in the word. If you guess wrongly, another part of the evil robot is assembled. When the robot is complete, it's game over.

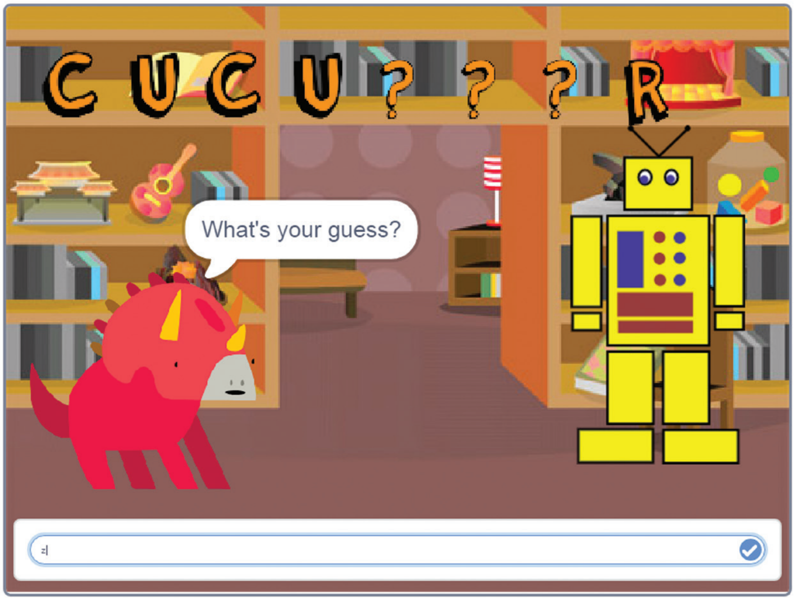
This is a one-player game, with the computer randomly picking a word from a list, and telling you whether your guesses are right or wrong. The game idea might seem simple, but implementing it in Scratch will draw on most of your skills so far.

You'll learn how to:

- Make Scratch talk out loud using the Text to Speech extension, a new feature in Scratch 3.
- Store lists of information, such as lists of letters or words, and use blocks to add, find and remove items in the list.
- Write text on the Stage using a sprite with costumes for each letter of the alphabet.
- Look at individual letters in a piece of text.
- Use functions to create scripts that are easier to understand, and easier to write.

Hot tip

It's okay to experiment with your programs to see what works and what doesn't. I tried making this game another way at first, using a different sprite for each letter in the word. It was clumsy synchronizing between the sprites, so I worked out the simpler solution you'll see in this chapter.



8 Space Swarm

Use what you've learned so far and build on it to make an arcade game. You'll learn how to use loops to create special effects, how to enable a sprite to fire at another sprite, and how to make a high score table.

Beware

This game requires keyboard controls, so you can't play it on tablet devices.

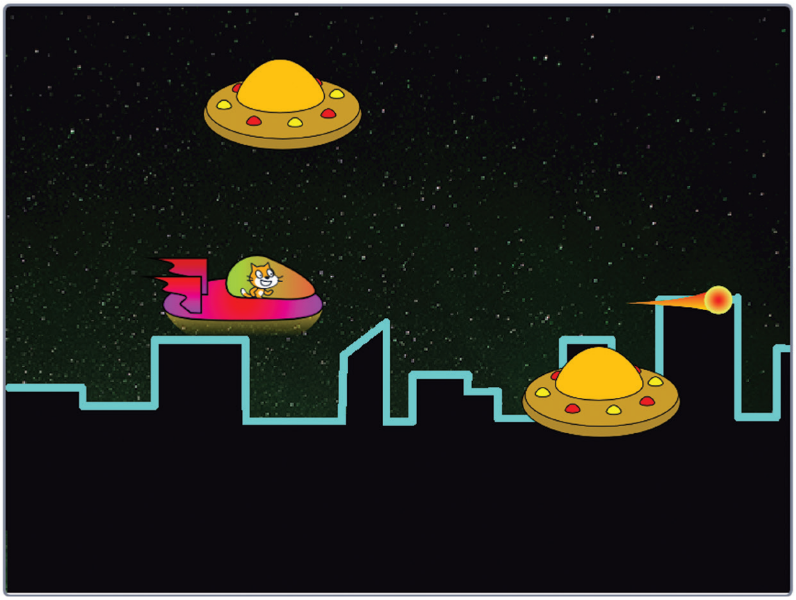
Introducing Space Swarm

We're under attack! As the aliens swarm around the planet, eager to plunder all of its resources, only one person can save us. You!

Space Swarm, in the spirit of classic arcade games, sees you firing on enemies and dodging their invasion. The aliens rush in towards you from the right of the screen, zig-zagging in random directions. To play the game, move your character with the arrow keys and press the spacebar to fire. You lose one of your three lives each time an alien hits you.

This game brings together much of what you've learned in previous chapters, and will also show you how to:

- Add special effects to make sprites materialize and evaporate.
- Enable one sprite to fire upon another.
- Use flags to swap information between sprites.
- Add looping music to your game.
- Make a high score table that remembers the best score anyone has achieved playing your game on the Scratch website.
- Give the player three lives.



9 Physical Computing with Scratch

Design new ways to interact with Scratch using hardware devices such as webcams, microphones, the Sense HAT add-on for the Raspberry Pi, and the micro:bit. This chapter includes two games that use tilt sensors and instructions for controlling a light connected to a Raspberry Pi. You'll also see how to make a clap-o-meter and games that respond to your actions on camera.

What is physical computing?

Physical computing is all about building computer projects that interact with the real world. In this chapter, we'll explore how different sensors can be used to control Scratch projects. We'll also introduce the simple electronics extension for the Raspberry Pi.

You'll see:

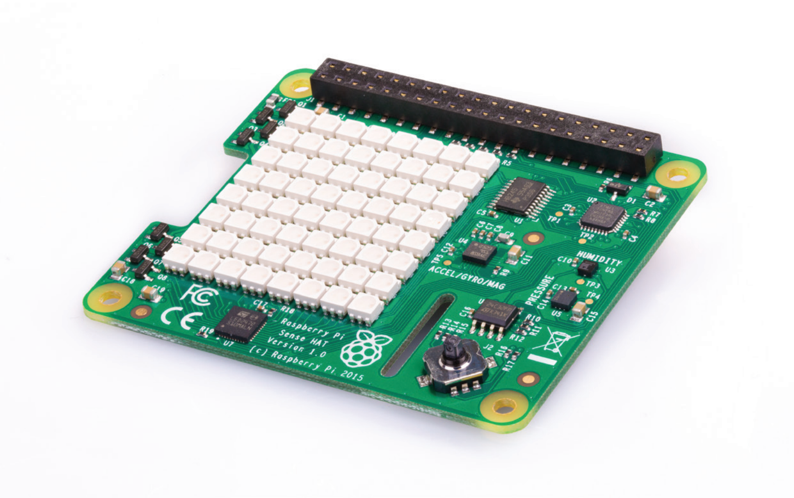
- How Scratch can detect motion on a webcam, and how you can use that to control action on the Stage.
- How Scratch can detect volume using a microphone connected to your computer, and use that to control scripts. We'll make a clap-o-meter, as used on TV talent shows to measure which act gets most audience applause.
- How the Sense HAT add-on for the Raspberry Pi can be used to detect movement, and to display messages using an 8x8 grid of multicolored lights. We'll make a game where you tilt the device to move around the screen, and use the tiny joystick to fire in four directions.
- How the micro:bit device can be used to detect movement and button pushes. We'll make a game that demands physical precision as you tilt the device to control a floating balloon.
- How to connect up an LED to the Raspberry Pi and turn it on and off using the simple electronics extension.

NEW

The new extensions feature in Scratch 3 has been used to add support for the Sense HAT and the micro:bit.

Hot tip

Check out our companion titles, Raspberry Pi in easy steps, Raspberry Pi 3 in easy steps, and Electronics in easy steps at www.ineasysteps.com, for more information on these topics.



Right: The Sense HAT adds sensors to a Raspberry Pi computer. In this chapter, you'll see how to use it as a game controller.

10 Seven shorties

To finish up, here are seven short programs you can build and experiment with, including five games, a chat program and an art generator.

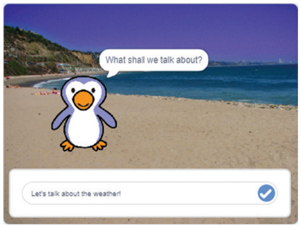
Penguin Patter

In Penguin Patter, you can chat to a friendly penguin:

1. Delete the cat. Add a backdrop and sprite. I've used the sprite Penguin 2 and Beach Malibu backdrop

2. Make the variable *random choice* and the list *sayings*. They can be for all sprites. Uncheck the boxes in the Blocks Palette to hide them

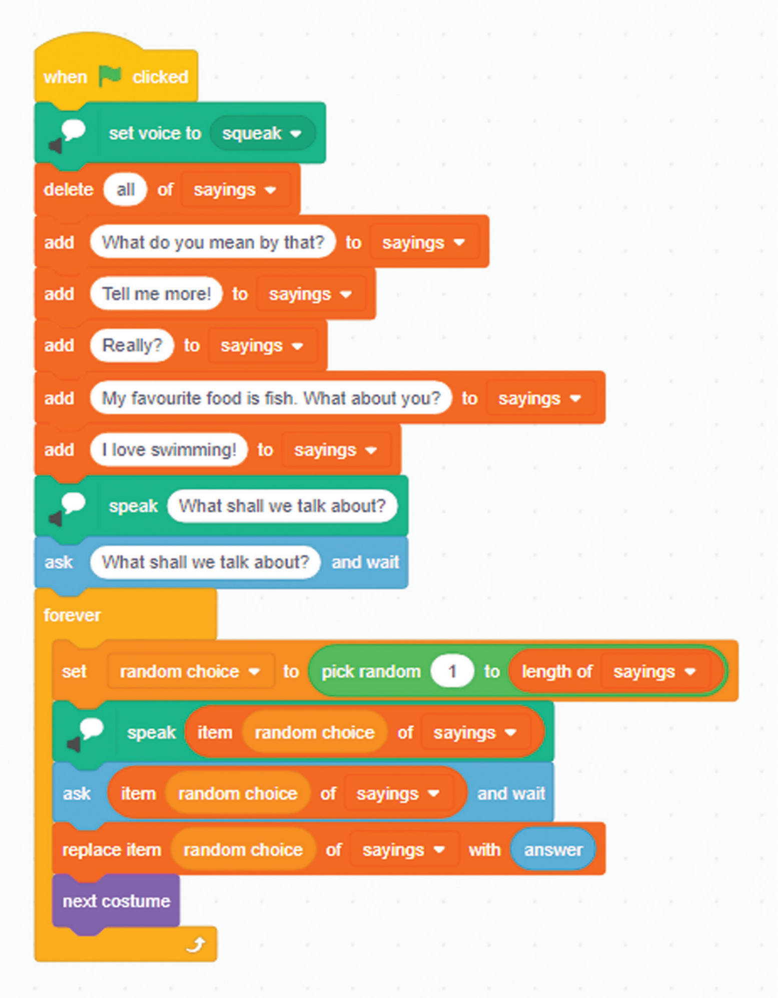
3. Add this script to your sprite. You'll need to add the Text to Speech extension for the **speak** block



What shall we talk about?

Let's talk about the weather!

Above: You type in what you want to say. The penguin's replies appear on screen and are also spoken out loud.



```
when [flag] clicked
set voice to squeak
delete all of sayings
add What do you mean by that? to sayings
add Tell me more! to sayings
add Really? to sayings
add My favourite food is fish. What about you? to sayings
add I love swimming! to sayings
speak What shall we talk about?
ask What shall we talk about? and wait
forever
    set random choice to pick random 1 to length of sayings
    speak item random choice of sayings
    ask item random choice of sayings and wait
    replace item random choice of sayings with answer
    next costume
```

Hot tip

I used a variable to remember which saying was randomly chosen, so the program can replace it with what the player enters. As a result, the penguin learns what you say and copies you.

Hot tip

Customize this game with your own personality. You can add lots more sayings at the start, and the program works better the more you add.

11 Making and sharing projects

Now it's time to make your own projects! Discover tips and resources to help here.

Making your own programs

Now you've seen how programs are built in Scratch, you can start to design and share your own creations. Here are some tips:

1. It's rare to program something that works first time unless it's a really simple program. Programming is all about trying something out, testing it, and then changing your scripts. Keep testing your program as you build it. You can click a script in the Code Area to run it straight away without running your whole program

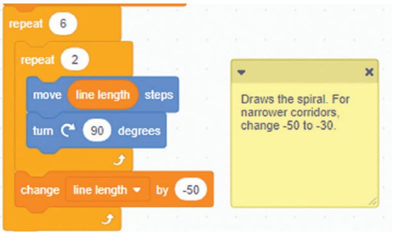
2. If you need to reset the position or look of a sprite for testing, remember you can click blocks in the Blocks Palette to use them on your sprite straight away

3. If you're having problems with a variable, click the checkbox beside it in the Variables part of the Blocks Palette. That will show the variable's value on the Stage so you can see what's going on. You can also show the x and y coordinates and direction of a sprite by checking the boxes beside them in the Motion blocks

4. If you can't see all the blocks, check whether you need to add an extension. Extension blocks have an icon on them and can be added using the **Add Extension** button in the bottom left of the Blocks Palette

5. To tidy up the Code Area, right-click on it and then choose Clean up Blocks from the menu. This lines up all your scripts and stops them overlapping

6. Add comments to your script to help you to remember what it's doing, and what you need to do with it next. Right-click on the Code Area to add a comment box. Click the bottom-right corner to resize it, and drag it into the Blocks Palette to delete it



```
repeat 6
    repeat 2
        move (line length) steps
        turn ⟳ 90 degrees
    change line length by -50
```

Draws the spiral. For narrower corridors, change -50 to -30.

Scripts glow in the Code Area as they run.

A great way to start making your own projects is by modifying existing projects. How can you build on the projects in this book?

Resources & Thank yous

The Scratch website
https://scratch.mit.edu/

The author's website, including examples, links to find the projects in this book online, and bonus content
https://www.sean.co.uk/scratch

Scratch Wiki: a reference guide to Scratch
https://en.scratch-wiki.info/wiki/Scratch_Wiki

Raspberry Pi forums, including one dedicated to Scratch
http://www.raspberrypi.org/forums/

To download the examples in this book, visit **www.ineasysteps.com**
Select **Free Resources** and then choose **Downloads**.

Acknowledgements

Scratch is developed by the Lifelong Kindergarten Group at the MIT Media Lab. See **https://scratch.mit.edu**

Special thanks to Mai T. Vu, J.D. and the students in her programming class at Casey Middle School in Boulder, Colorado; Darrell Little; and Laurence Molloy, who all helped with testing projects for this updated second edition.

Thank you to everyone who read the first edition and sent such lovely feedback! It's been amazing to see your projects!

Sense HAT photos are courtesy of Raspberry Pi. Image of micro:bit courtesy of micro:bit Educational Foundation at **microbit.org**. Thank you to those who helped with permissions and research queries, including Mitchel Resnick; Liz Upton, Eben Upton, Lucy Hattersley and Helen Lynn of Raspberry Pi; Jan Boström; Tim Benson; and Mike Cook.

Extra special thanks to Karen McManus, who was a fantastic help throughout this project, in particular with indexing and layout into the In Easy Steps style; and to star playtester Leo.

About the author

Sean McManus is an expert technology writer and Scratch enthusiast. He writes for The MagPi magazine and his other books include Cool Scratch Projects in easy steps, Coder Academy, Raspberry Pi For Dummies, and Web Design in easy steps.
Visit his website at **www.sean.co.uk** for additional resources.

Also available in this series by the same author: Cool Scratch Projects in easy steps. Build a drum machine, make projects with 3D glasses (included in the print edition), animate cartoon frogs, code maze games, and more!

Don't forget

If you enjoyed this book, please blog about it or write a review of it on your favorite online store! Thank you!

Index

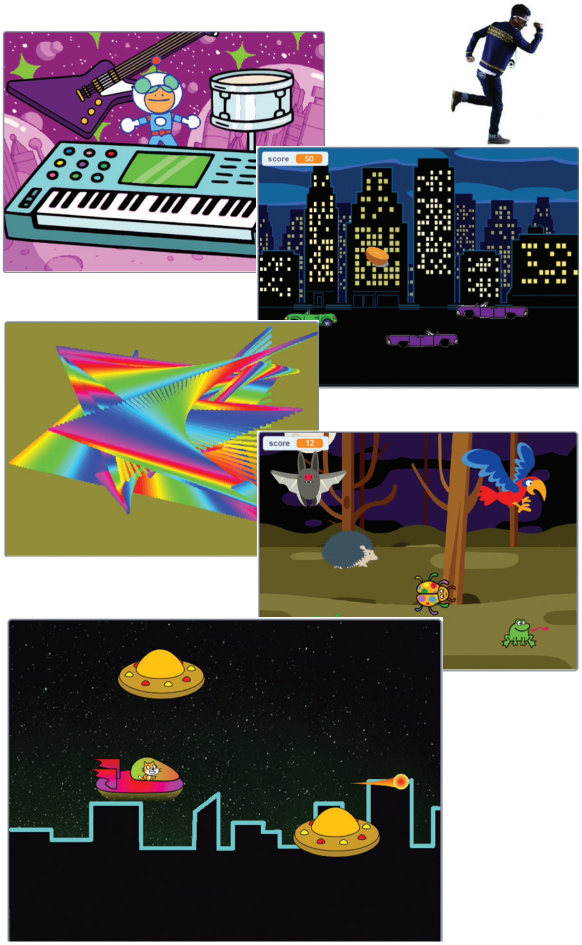
Learn to program with Scratch

Scratch makes it easy to create your own games, animations, music, art or applications. It's the perfect way to learn programming because it takes away a lot of the complexity. That means you can focus on having great ideas and bringing them to life. With this book as your companion, you'll learn how to:

- Build and share exciting Scratch projects.
- Create addictive arcade games, quizzes and word games.
- Make computer-generated art.
- Play music and compose your own tunes.
- Use variables, lists, loops, broadcasts, and operators to create sophisticated software.
- Avoid common programming pitfalls and bugs.
- Interact with webcam video, the Raspberry Pi Sense HAT, and the micro:bit.

This popular book has now been updated to cover Scratch 3 so you can use all the latest features.

With no prior knowledge needed, we'll take you from scratch to Scratcher!



Why choose this book?

It's written in plain English
Put off by the jargon? Don't worry, we keep things straightforward.

It's easy to follow
Clear, step-by-step instructions make learning simple.

It's fully illustrated
We don't just tell you how to do it, we also **show** you how.

It's in full color
This book's printed in color to make it simpler to use and easier on the eye.

And it's fantastic value
Do we need to say any more?

The Scratch programming language is widely used in schools and on the Raspberry Pi. Its drag-and-drop commands make it an ideal language for all ages to learn to program. And this popular book, **Scratch Programming in easy steps**, now fully updated for Scratch 3, is packed with ideas and games that illustrate what's possible with Scratch.

This book introduces you to Scratch essentials and then walks you through the commands to create games and animations that'll impress. Learn to:

- Build games that require skill, knowledge or quick fingers
- Add music
- Create eye-catching visual effects
- Keep score
- Avoid common pitfalls and learn how to fix bugs

Use Scratch Programming in easy steps to get creative and become a super Scratcher!

Let these icons make it even easier

Hot tip — to spice up your learning

Don't forget — highlights something worth remembering

Beware — wards you off potential danger!

NEW — indicates a new or changed feature

£10.99 UK / $15.99 US

ISBN 978-1-84078-859-4
US$15.99
51599>
9 781840 788594

Categories: Children's>Hobbies>Computers
Children'>Education>Technology
Computers> Programming

www.ineasysteps.com