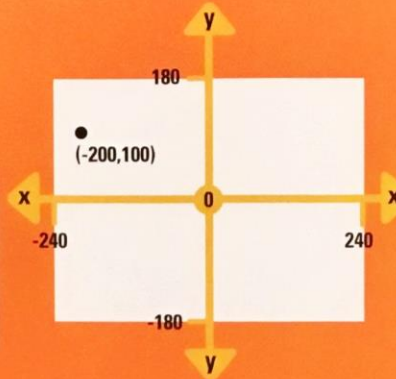# Set x and y coordinates

`set x to -200`

**Set x to** tells Scratch how far to place the sprite to the left or right of the screen.

`set y to 100`

**Set y to** tells Scratch how far to place the sprite up or down the screen.

y
180
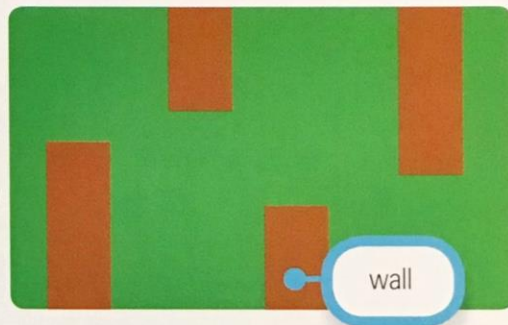● (-200,100)
x
-240
0
240
x
-180
y

## 4

Draw a simple background for the game. Look at step 5 on page 45 for help getting started.

Use the **Rectangle** tool to draw some walls. Make them all the same color.
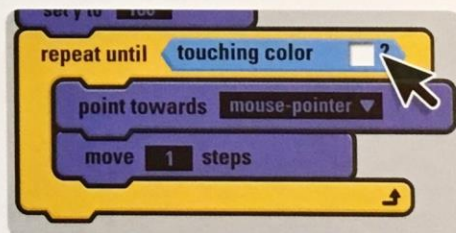
wall

The **Repeat until** block will loop forever because we haven't told it when to stop yet. It needs to repeat until the sprite touches a brown color—the wall color.
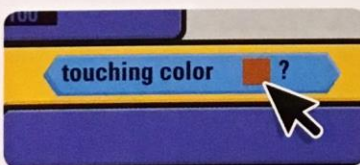
**Sprites**

Sprite1

Click on the Scratch sprite icon and then the **Scripts** tab to bring your code back.

Events
Control
Sensing
Operators
More Blocks

Click the **Sensing** group.

`repeat until  touching color    ?`
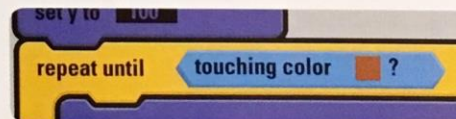`point towards  mouse-pointer ▼`
`move  1  steps`

Drag a **Touching color** block onto the top of the **Repeat until** block.

`touching color    ?`

Click the colored square then choose the color to check for...

...by clicking one of the walls.

`repeat until  touching color    ?`

Now your game will play until the sprite hits a wall. Test it by clicking the green flag at the top of the screen. To play again, drag your sprite away from the wall.

# REPEAT UNTIL CAUGHT

*Eeeeeek!*

We're going to practice using "repeat until" loops by creating another game. This will have two sprite objects, the Scratch cat sprite and a dog, which will chase after the cat. The player will control the cat, moving it around the screen until it is caught.

## 1

## How to code a chase game

Drag code to the scripts area to make the Scratch sprite move slowly across the screen, pointing towards the mouse pointer.

```
when [green flag] clicked
repeat until
    point towards   mouse-pointer ▼
    move   2   steps
```
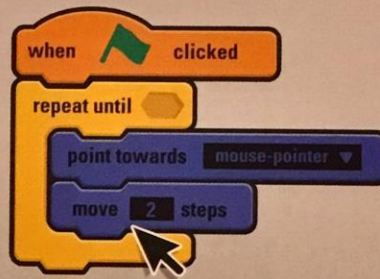
Look at steps 3 and 4 on pages 44–45 for help. Remember to use the **Repeat until** loop. Change the speed of the sprite to move 2 steps each loop.

## 2

Make the Scratch sprite smaller by clicking the **Shrink** icon at the top of the screen and then clicking the Scratch sprite several times.

*Click the green flag at the top of the screen to test your code.*

*Pick me!*

## 3

Now add the second sprite.

rite: 👦 ✔ 📤

Click this icon to choose a new sprite from the library.

Dog1     Dog2

Then scroll down to the **dog** and click on it.

**OK** Click **OK**.

Use the **Shrink** icon to make the dog smaller.

*Pick me!*

*Pick me!*

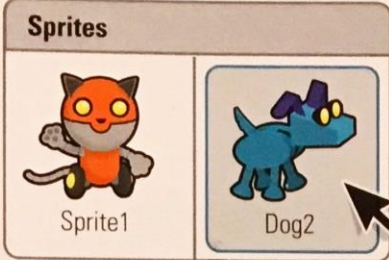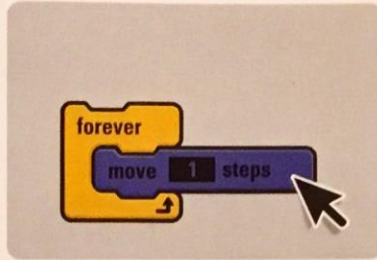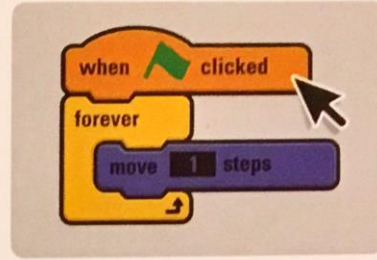## 4

Now we will make the dog sprite move.

**Sprites**



Click the **dog** so the code you are about to build will control the dog rather than the cat!

Drag in a **Forever** loop block from the **Control** group. Add a **Move** block. Set it to 1 step each loop.
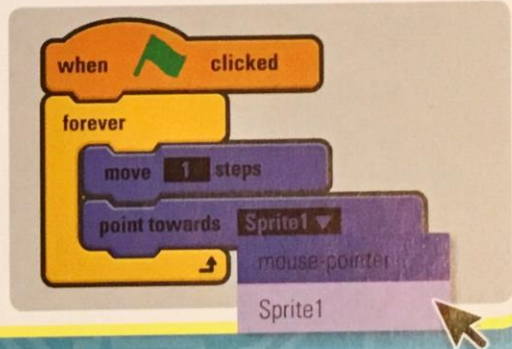
From the **Events** group, drag a **When green flag clicked** block over. Put it at the top of the code.

---

After step 4, test your code. The cat should follow the mouse pointer. The dog should move in a straight line forever—this means it might get stuck on the right side of the screen. Just drag it to the left!

## 5

To make the dog chase the cat, click the **dog** sprite.

**Sprites**



From the **Motion** group, drag on a **Point towards** code block. Set it to "Sprite1."

---

## 6

Now the most important part—making the "until" part of the "repeat until" loop! The aim is to keep the cat moving until the dog catches it.
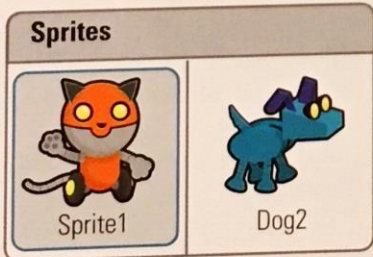
**Sprites**



Click the **cat** sprite.

Drag a **Touching** code block from the top of the **Sensing** group and set it to "Dog2."

Now click the green flag and start to play your game!

# ADDING SOUND

Our code so far has used two different inputs: pressing keys and moving the mouse. Our outputs have all been on screen. We are now going to learn how to control another sort of output—sound.
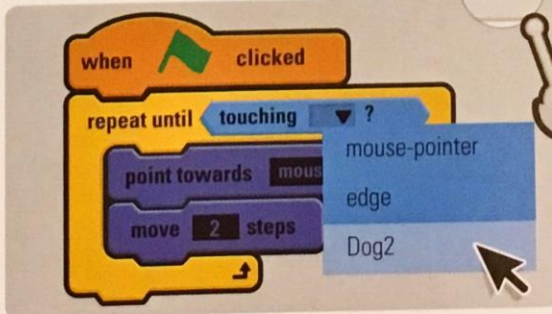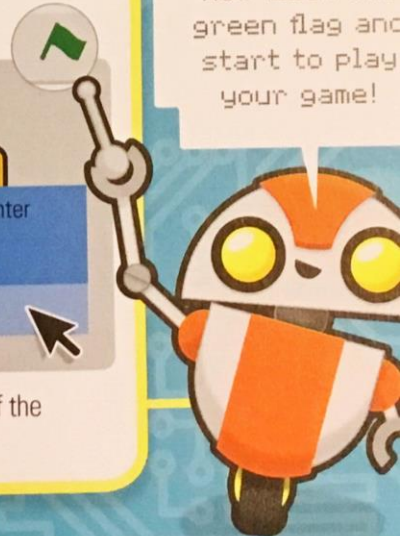
## Getting started with sound in Scratch

Click the **Sound** group.

| Looks | Co |
| Sound | Se |
| Pen | Op |
| Data | M |

play note 60 ▼ for 0.5 beats

Drag a **Play note** code block onto the scripts area and try clicking it.

## Key word

**Output:** The information produced by a computer, such as sound or movements of the sprite

## Changing beats

play note 60 ▼ for 2 beats

Try changing the value in the "beats" box to 2. Click the code block.

play note 60 ▼ for 0.25 beats

Now change it to a small number—0.25 (a quarter of a beat). Click it.

The bigger the number, the longer the note is played. Experiment!

## Changing the note

Change how high or low the note is by changing the value in the "note" box.

Either type a number in, or choose a note from the keyboard.

play note 64 ▼ for 0.5 beats

E(64)

The higher the number, the higher the note. The lower the number, the lower the note.

You may not be able to hear notes much below 20, or above 100—but your dog might!

# Creating a tune

Drag a few **Play note** code blocks onto the scripts area and change their note values.

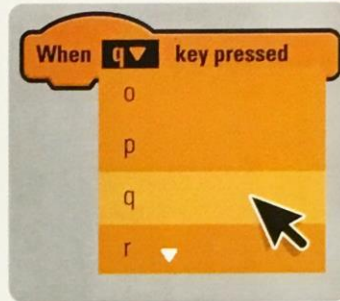Click the top one to play all the notes. Experiment to make your own tune!

```
play note  64 ▼  for  0.5  beats
play note  62 ▼  for  0.5  beats
play note  60 ▼  for  0.5  beats
play note  60 ▼  for  0.5  beats
```

# Make a piano program

**1** Click the **Events** group.

```
Events
Control
```

**2** Drag a **When key pressed** code block onto the scripts area and set it to "q."

```
When  q ▼  key pressed
         o
         p
         q
         r  ▼
```

**3** Click the **Sound** group.

```
Motion        Eve
Looks         Cor
Sound         Ser
Pen           Ope
Dot           Mo
```
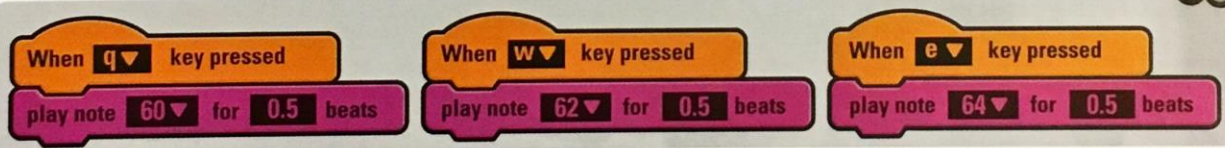
**4** Drag a **Play note** code block onto the **When key pressed** code block, so the note will be played when the "q" key is pressed.

```
When  q ▼  key pressed
play note  60 ▼  for  0.5  beats
```
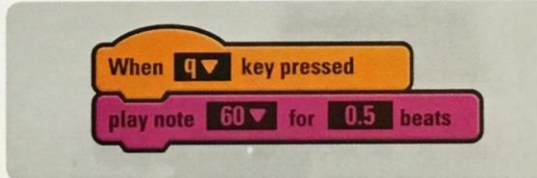
**5** Repeat steps 1 to 4 to create more blocks of code. Then change the value of the keys that they will respond to and the notes they will play, so they look like this:

```
When  q ▼  key pressed
play note  60 ▼  for  0.5  beats
```

```
When  w ▼  key pressed
play note  62 ▼  for  0.5  beats
```

```
When  e ▼  key pressed
play note  64 ▼  for  0.5  beats
```

Add more **When key pressed** blocks to complete your piano. You can experiment with the **Set instrument** block to change the sound.

# SOUND EFFECTS

The previous pages looked at how we could make sound using code. We are now going to look at how we can use sound within loops, and how to add sound effects to games.
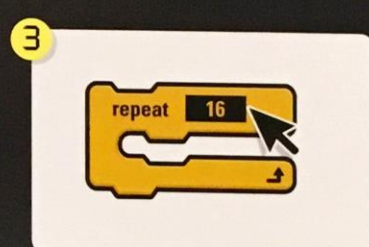
## Create a drum machine

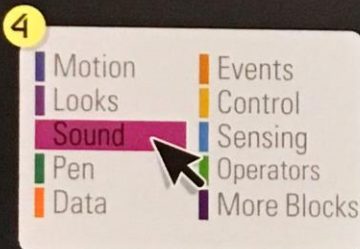We can combine sound effects with loops to make a simple drum machine.
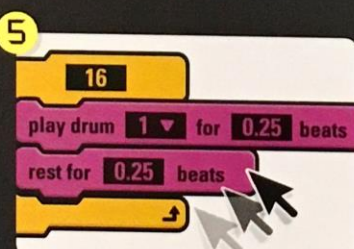
**1**

- Events
- **Control**
- Sensing
- Operators
- More Blocks

Click on the **Control** group.

**2**

`repeat 10`

Drag a **Repeat** loop block onto the scripts area.

**3**

`repeat 16`

Change the number of loops to repeat to 16.

**4**

| | |
|---|---|
| Motion | Events |
| Looks | Control |
| **Sound** | Sensing |
| Pen | Operators |
| Data | More Blocks |

Click on the **Sound** group.

**5**

`16`
`play drum 1 ▼ for 0.25 beats`
`rest for 0.25 beats`

Drag in a **Play drum** and a **Rest** block.

**6**

`play drum 1 ▼ for 0.25 beats`
`rest for 0.25 beats`
`play drum 2 ▼ for 0.25 beats`
`rest for`
(1) Snare Drum
(2) Bass Drum
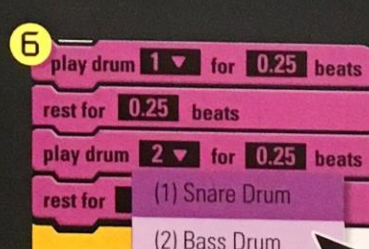
Add another **Play drum** and **Rest** block. Choose "Bass drum" this time.

Click the **Repeat** block to play your drum machine. Experiment by changing the "drum" number and how many beats it plays for. Change the number of times the loop repeats to make the drum machine play for longer.

Add this code so the drum machine speeds up when you press **f**.

`When f ▼ key pressed`
`change tempo by 20`

Or slow it down again by changing the tempo by −20.

`When s ▼ key pressed`
`change tempo by -20`

# Add sound to your games

Games can be a lot more fun if they have sound effects. We are going to learn how to add sound to the games we made in the previous pages.

```
when [flag] clicked
set x to -200
set y to 100
repeat until < touching color [ ] ? >
    point towards  mouse-pointer ▼
    move  1  steps
```

1. Make the **Maze Game** on page 46.

   We are going to add a sound effect that is played when the player crashes into a wall.

   We need to figure out where to put a **Play drum** code block.

   It needs to be put at the end of the **Repeat until** loop, outside the loop.

2. Click on the **Sound** group.

| Motion | Events |
|--------|--------|
| Looks | Control |
| Sound | Sensing |
| Pen | Operators |
| Data | More Block |

3. Drag a **Play drum** code block to the end of the loop and join it on.

   Choose "Drum 10" then test your game!

```
play drum  10 ▼  for  1  beats
```

```
when [flag] clicked
repeat until < touching  Dog2  ? >
    point towards  mouse-pointer ▼
    move  2  steps
play sound  meow ▼
```

1. Make the **Chasing Game** on page 48.

   Start by remaking the code for the cat and the dog. Check to make sure the game works properly first.

2. First, click on your **cat** sprite. From the **Sound** group, drag a **Play sound meow** code block to the end of the **Repeat until** loop for your cat sprite.

   Now when the dog catches the cat, the cat will say "meow!"

You can insert a start-up tune for your game here. Copy the one shown on the right—or compose your own!

```
when [flag] clicked
[         ]
repeat until < touching  Dog2  ? >
    point towards  mouse-pointer ▼
    move  2  steps
```

```
set instrument to  16 ▼
set tempo to  480  bpm
play note  48 ▼  for  0.5  beats
play note  60 ▼  for  0.5  beats
play note  72 ▼  for  0.5  beats
```

# KEEPING SCORE

## How to code a deadly shark game

We are going to make another chasing game, this time with the player avoiding a shark for as long as possible. We will use a variable to count how long the player avoids the shark and use this variable as the score.

**1** Start Scratch. Delete the main sprite and create your own fish sprite.

Shrink the fish.

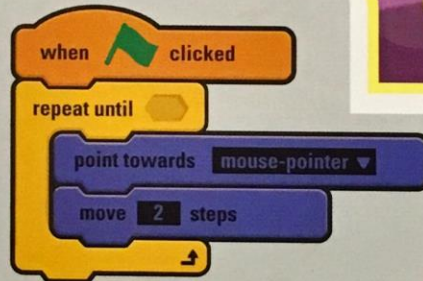For help, see steps 1 and 2 on page 44.

*i*

Sprite1

**2** We will create code to make the fish swim toward the mouse pointer.

See pages 44–45 for help—but remember to use a **Repeat until** loop instead of a **Forever** loop.

Change the amount moved to 2 steps.
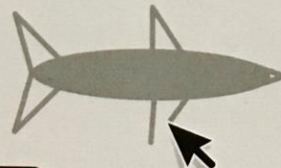
Click the green flag to test your code.

```
when [green flag] clicked
repeat until
  point towards  mouse-pointer ▼
  move  2  steps
```

**3**

w sprite: 👤 / 📤

Start to make a shark to chase the fish by clicking **Paint new sprite**.

Choose a color and select the **Ellipse** tool. Draw a narrow ellipse.

Use lines to draw fins, nose and tail.

Then fill them in.

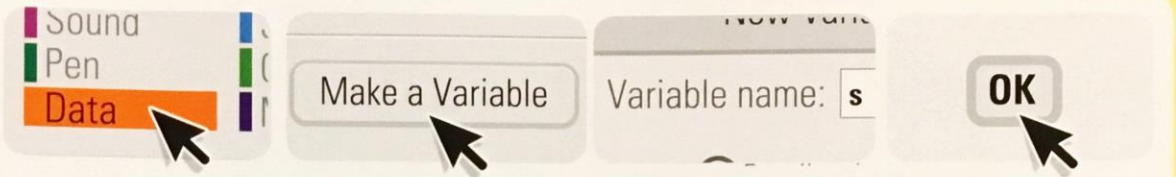Add an eye.

If you make a mistake, use the **Undo** button.

Use the **Shrink** button to make the shark sprite smaller.

**4**

Now we will make a variable to keep the score.

| Sound | |
|-------|---|
| Pen | |
| **Data** | |

Make a Variable
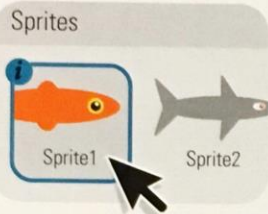
Variable name: s

**OK**

Click on the **Data** group.  Click **Make a Variable**.  Call it "s" (for score).  Then click **OK**.

**5**

Click your **fish** sprite so we can add more code to it.
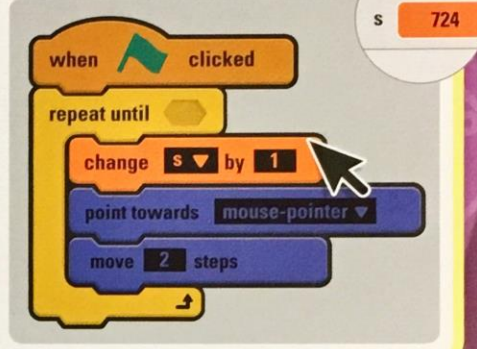
Sprites

Sprite1    Sprite2

**6**

The score needs to go up as the fish swims around.

Drag a **Change s by 1** code block from the **Data** group into the loop here.

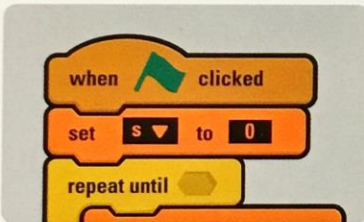Click the green flag to test your code.

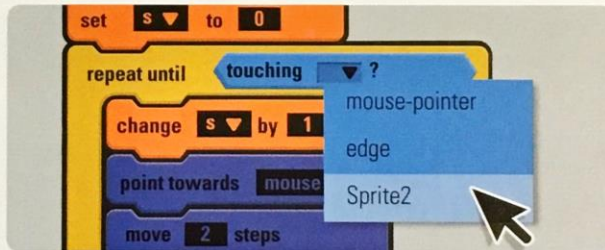The score in the top left of the stage should keep going up.

s   724

when ⚑ clicked
repeat until
change s by 1
point towards mouse-pointer
move 2 steps

**7**

The score must be reset each time the game is played. Drag a **Set s to 0** code block into your program before the start of the loop.
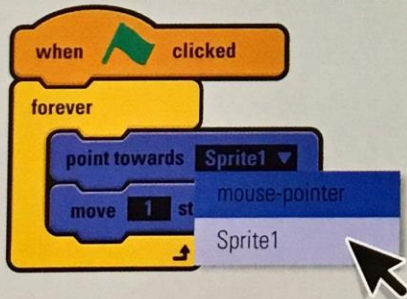
when ⚑ clicked
set s to 0
repeat until

**8**

Next we will make the game stop when the shark catches the fish. From the **Sensing** group, drag a **Touching** code block onto the **Repeat until** loop. Set it to "Sprite2."

set s to 0
repeat until touching ▼ ?
change s by 1
point towards mouse
move 2 steps

mouse-pointer
edge
Sprite2

**9**

when ⚑ clicked
forever
point towards Sprite1 ▼
move 1 st

mouse-pointer
Sprite1

Double-click the **shark** sprite so you can add code to control it. Copy the code on the left, which will make the shark chase your fish. See step 5 on page 49 for extra hints.

Now test your game!

Move the sprites apart to start a new game.