

There are lots of ways we can ask people to do things. If someone said, "Turn on the lights," "Put the lights on," or even "It's dark—turn on the thingy," you would know what to do. But to program a computer, we need to give exactly the right words—and in the right order. Words that tell computers and people what to do are called instructions.

Making breakfast

Imagine you are programming our friendly robot Ada to make breakfast. Can you put these instructions in the correct order?

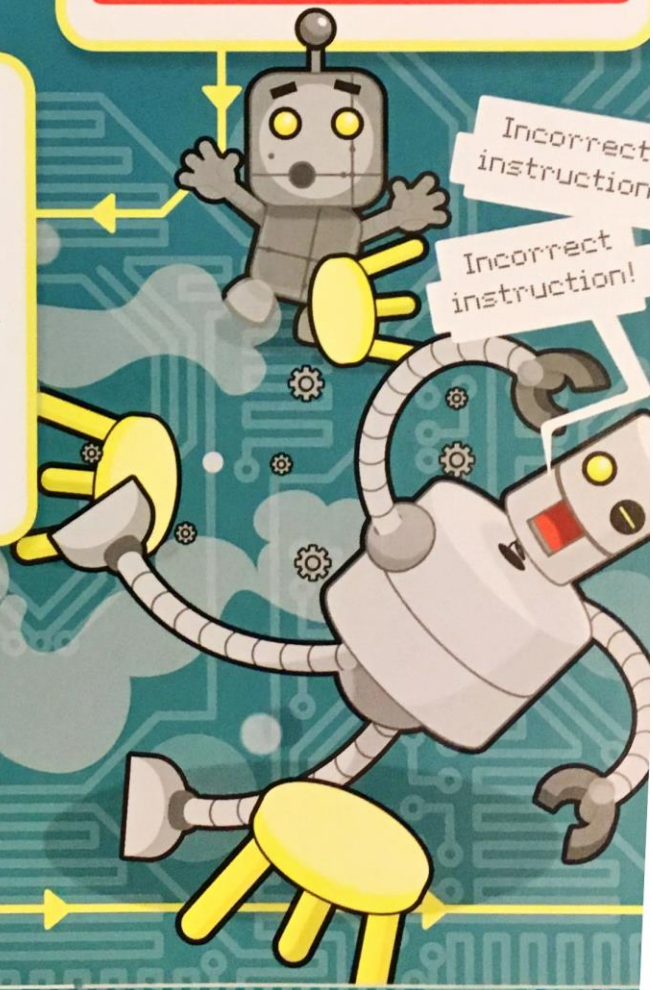
- A** Open the cereal box.
- B** Pour some milk onto the cereal.
- C** Take the lid off the milk carton.
- D** Tip some cereal into the bowl.
- E** Get a bowl from the cupboard.

Become a human robot

It's time to become a human robot! It will help you think about how to give precise instructions. You need a partner to play this game.

One of you needs to pretend to be a robot. The other one needs to be the programmer and give instructions to the robot. This player's task is to give the robot instructions to walk to the door. The only commands that can be given to the robot are:

- Walk forward.**
- Turn left.**
- Turn right.**
- Stop.**



Robot artist

Here's another game to help you practice giving instructions.

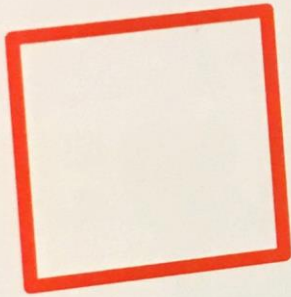
You need:

- 1 A partner
- 2 A piece of paper
- 3 A pencil

Sit at a table next to your partner. One of you needs to be the robot artist, while the other is the programmer. The programmer needs to give the robot instructions to draw one of the pictures below. This time the robot is just moving a pencil. The robot partner is only allowed to do what the programmer says. Here are the commands you can use:



1



2



Once you get good at this game, why not try getting the robot to play with their eyes closed. Be careful not to draw on the table!

PEN DOWN

Now let's learn how to draw using Scratch. We need to use the command "Pen down," then command the sprite to move around.

All square

Try this exercise to draw a square:

1

Click on the **Pen** group.



2

Drag **Pen down** to the scripts area.



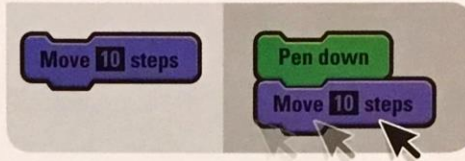
3

Click on the **Motion** group.



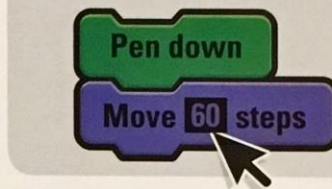
4

Drag a **Move 10 steps** code block to join on to your program.



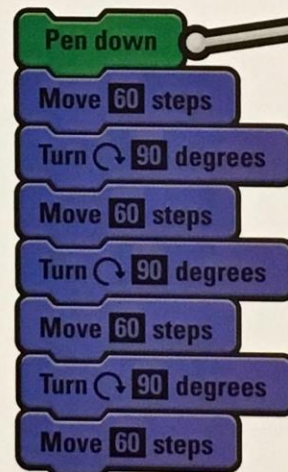
5

Change the 10 to 60.



6

Complete the program:



Click any block to run the program.

Drawing shapes

Now change your code so it looks like this:

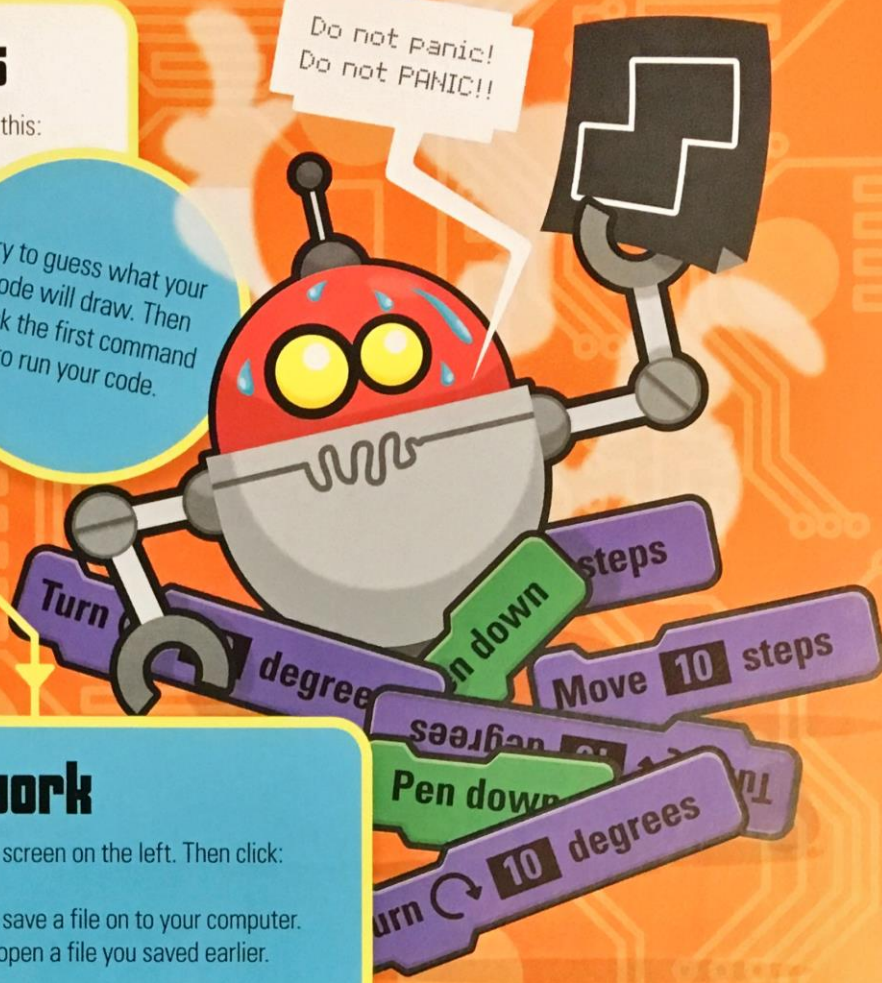
```

Pen down
Move 20 steps
Turn 90 degrees
Move 80 steps
Turn 90 degrees
Move 20 steps
Turn 90 degrees
Move 80 steps

```

Try to guess what your code will draw. Then click the first command to run your code.

Do not panic!
Do not PANIC!!



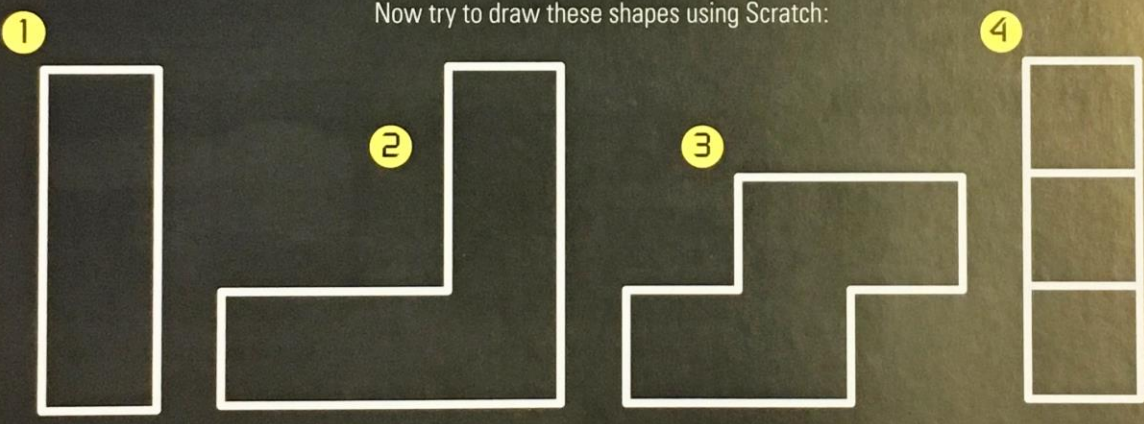
Storing your work

Click the **File** menu at the top of the screen on the left. Then click:

- New** – to start some new work.
- Download to your computer** – to save a file on to your computer.
- Upload from your computer** – to open a file you saved earlier.

More shapes!

Now try to draw these shapes using Scratch:



See page 33 for answers.

PRESS A KEY

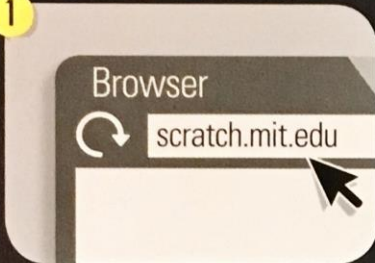
So far, all the code we have written runs when we tell it to start. We are now going to learn how to make our code change when different keys are pressed. A key press during a program is a type of input.

Right and left

When **R** is pressed, we want the sprite to move right. When **L** is pressed, we want it to move left.

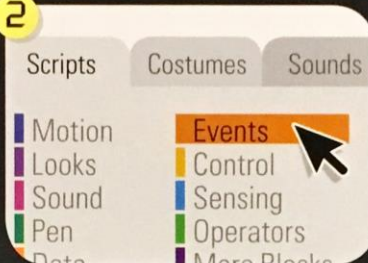
Press R to move right

1



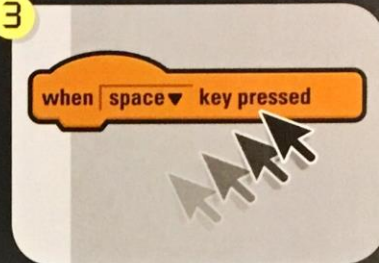
Start Scratch. Click **'Create'** or **'Try it out.'**

2



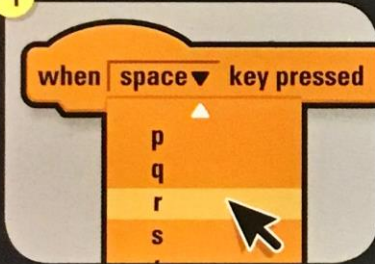
Click on the **Events** group.

3



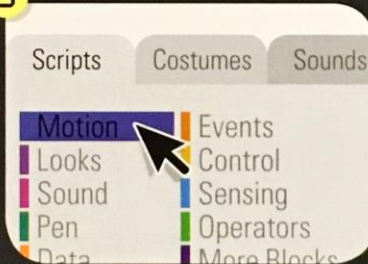
Drag **'When key pressed'** to the scripts area.

4



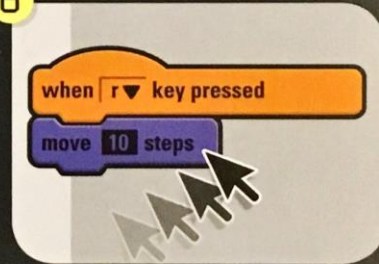
Select **r** as your key.

5



Click on the **Motion** group.

6



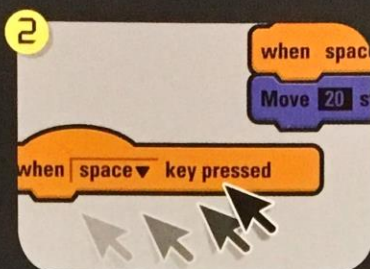
Drag **Move steps** to the scripts area.

Now try pressing the **R** key on the keyboard...

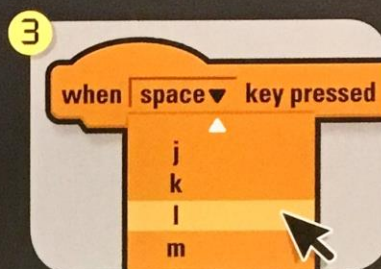
Press L to move left



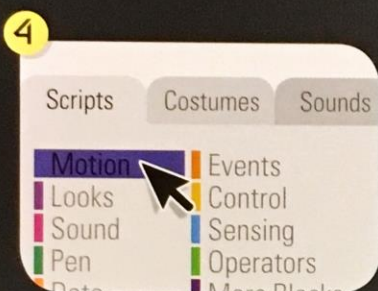
Click on the **Events** group.



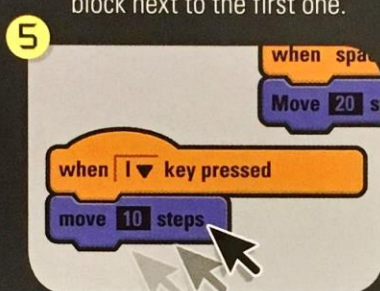
Drag a **When key pressed** block next to the first one.



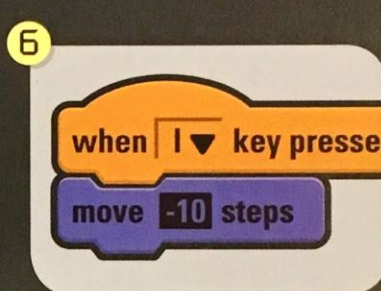
Select **l** as your key.



Click on the **Motion** group.



Drag over a **Move steps** block.



Change the move amount to **-10** steps.

Now press **L** to move left and **R** to move right!

How does the code work?

We have made two different bits of code. When we press the **R** key on the keyboard, this tells Scratch to move the sprite right by 10 steps.

When we press the **L** key, it tells Scratch to move the sprite by minus 10 steps—making it move in the opposite direction by 10 steps.

Each bit of code runs when we press a key. We are using two different inputs to make our program run two different bits of code.

Can you make your sprite move more quickly?

Change your code so the sprite will move when you press the arrow keys on the keyboard.

Key word

Input: An action (such as pressing a key) that tells a program to do something

We've learned how to move a sprite to the right and left. We will now find out how to move it up, down, and any other direction you want by pressing different keys.

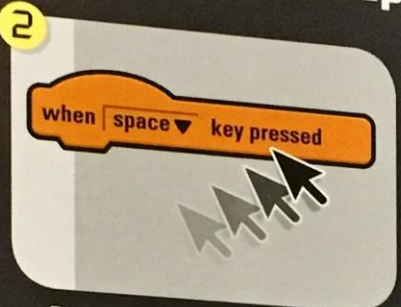
Using degrees to make turns

We are going to make a program that has four different input keys. Each input key will run code that makes the sprite point in a different direction.

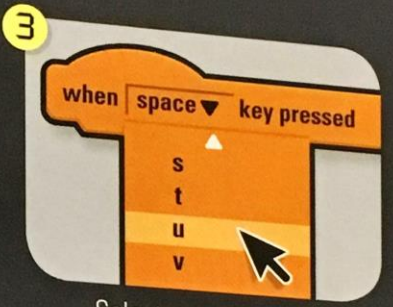
Press U to point up



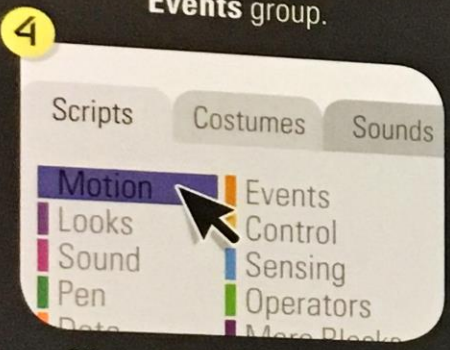
1 Start Scratch, then click on the **Events** group.



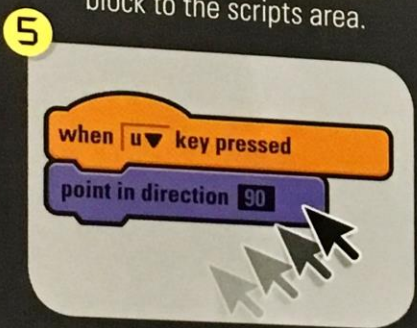
2 Drag a **When key pressed** block to the scripts area.



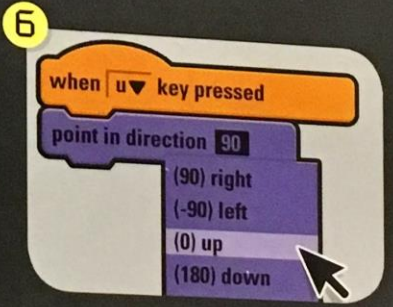
3 Select **u** as your key.



4 Click on the **Motion** group.



5 Drag a **Point in direction** block to the scripts area.



6 Select **(0) up**.

See what happens when you press **U**

The code says: when the **U** key is pressed, point in direction 0 degrees, which means point upward.



Press D to point down

We still need to make the sprite move. When we press **D**, we want it to point down.

1

```

when u key pressed
  point in direction 0
  move 10 steps
  
```

Drag in a **Move steps** block. Press **U** to test it!

2

```

when space key pressed
  Move 20 steps
  
```

Drag over another **When key pressed** block. Put it next to the first one.

3

```

when space key pressed
  b
  c
  d
  e
  
```

This time, select **d** as your key.

4

```

when d key pressed
  point in direction 90
  
```

Drag in a **Point in direction** block.

5

```

when d key pressed
  (90) right
  (-90) left
  (0) up
  (180) down
  
```

Select **(180) down**.

6

```

when d key pressed
  point in direction 180
  move 10 steps
  
```

Drag in another **Move steps** block.

Check your code works by pressing **U** and **D**

You need to finish this program by adding code to make the sprite move left and right. Add two more **When key pressed** blocks, so they change the sprite's direction when **L** and **R** are pressed.

Point **left** and move:

```

when l key pressed
  point in direction -90
  move 10 steps
  
```

Point **right** and move:

```

when r key pressed
  point in direction 90
  move 10 steps
  
```



SKETCHING WITH INPUTS

Now that we know how to move a sprite around with input commands, we are going to make a simple sketching program. Players will be able to draw what they want by pressing different keys to paint up, down, left, or right.

Create your own drawing game

1

Start Scratch, then make a program that will move the sprite up, down, left, and right. Turn back to page 26 for a reminder of how to do this. Test your program!

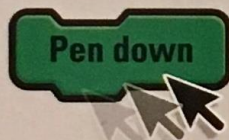
The code block for 'when u key pressed' contains 'point in direction 0' and 'move 10 steps'. The code block for 'when l key pressed' contains 'point in direction -90' and 'move 10 steps'. The code block for 'when r key pressed' contains 'point in direction 90' and 'move 10 steps'. The code block for 'when d key pressed' contains 'point in direction 180' and 'move 10 steps'.

Challenge

Try changing the keys used to move things around. You could use the cursor (arrow) keys.

We need to make the sprite draw a line when it moves. Click on the **Pen** group. Drag a **Pen down** block onto the scripts area. Click the **Pen down** block, then try pressing the **U**, **D**, **R**, and **L** keys.

Looks
Sound
Pen
Data



3

To clear the screen we need to use the **Clear** command block. Drag it over to the scripts area, then try clicking it.

Looks
Sound
Pen
Data

