

## Party piñata

Create a swinging piñata that releases treats when it is hit, and eventually breaks to reveal a birthday message



### Step 1 Introduction

In this project, you will create a swinging piñata that releases treats when it is hit, and eventually breaks to reveal a birthday message.

A **piñata** is often used to celebrate a birthday, new year, or festival. The shape and decorations of a piñata are designed to match the theme of the occasion. It's also full of treats. The aim is to hit the piñata with a stick to break the piñata and release the treats.



**Code Club** is celebrating its 10th birthday this year. Did you know that there are Code Clubs on **six out of seven continents** around the world? That's a lot of coders making things with code! Do you think any other clubs will be doing this project right now?

## Step 2 Start the party

In this step, you will choose a piñata costume and code the piñata to swing.



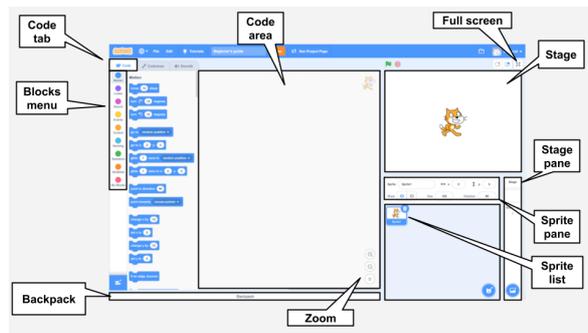
Open the **Party piñata starter project** (<https://scratch.mit.edu/projects/653082997/editor>).  
Scratch will open in another browser tab.



### Working offline

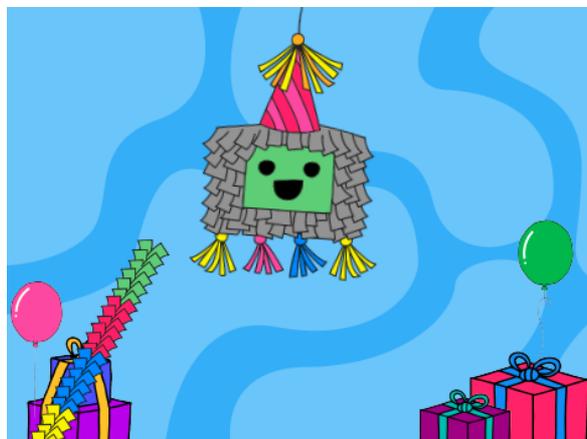
For information about how to set up Scratch for offline use, visit **our 'Getting started with Scratch' guide** (<https://projects.raspberrypi.org/en/projects/getting-started-scratch/1>).

The Scratch editor looks like this:



The **Stage** is where your project runs and a **backdrop** changes the way that the Stage looks. A Code Club party backdrop has been added for you.

In Scratch, characters and objects are called **sprites**, and they appear on the Stage. You can see the **Piñata** and **Stick** sprites on the Stage.

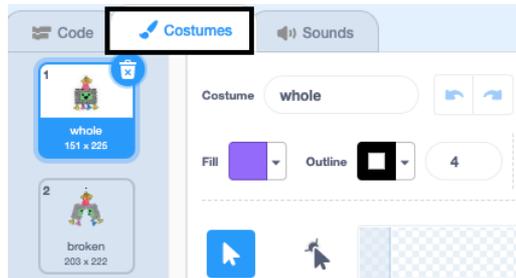
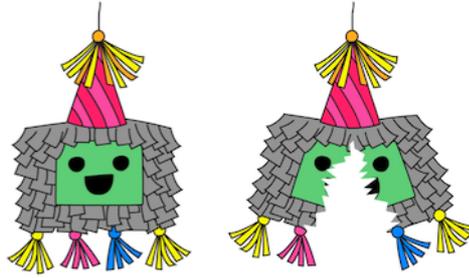


At the moment there is not much happening at this party. You can change that!

A sprite can have code, costumes, and sounds to change the way that it looks and what it does.



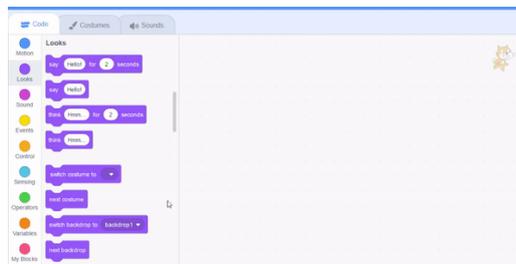
Click on the **Piñata** sprite in the Sprite list and select the **Costumes** tab. There are two piñata costumes, one named 'whole' and the other named 'broken'.



Click on the **Code** tab. Go to the **Looks** blocks menu then drag a **switch costume to** block to the Code area.

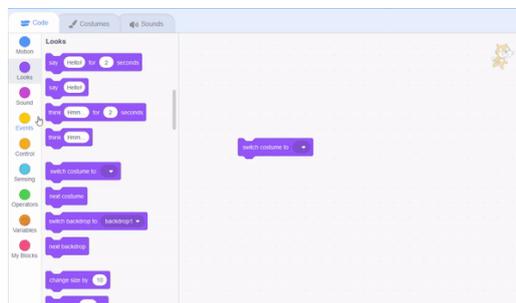
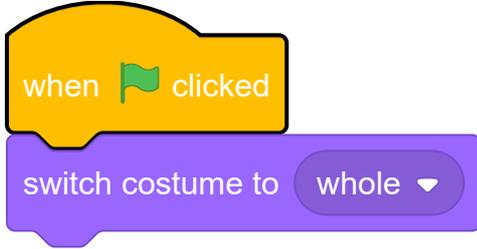


Click on the costume name to open a **drop-down menu** then select the **whole** costume:



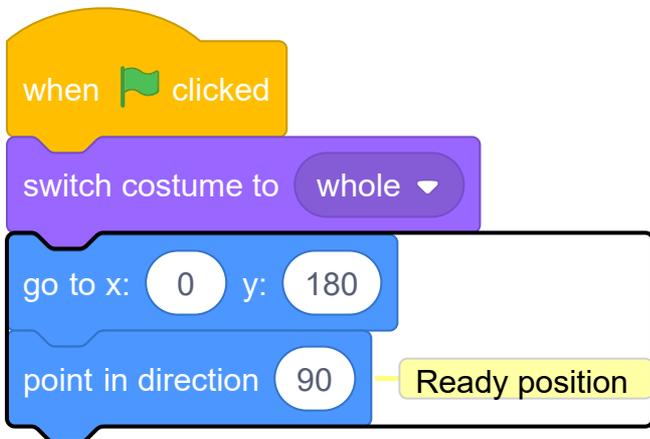
Blocks can be connected together in the Code area to run more than one at a time. Connected blocks will run in order from top to bottom.

Drag a **when flag clicked** block from the **Events** blocks menu and connect it to the top of your looks block in the Code area. The blocks will snap together:



The starting position of a piñata is always the same, it only starts moving when the piñata game is ready to play.

In the **Motion** blocks menu, find the **go to x: 0 y: 180** and **point in direction 90** blocks. Drag the blocks to the Code area and connect them to the bottom of your code:



A **forever** loop runs the code blocks inside it again and again. It is the perfect loop for a swinging piñata that is hard to hit.

Drag a **forever** block from the **Control** blocks menu and connect it to the bottom of your code:



```
when green flag clicked
  switch costume to whole
  go to x: 0 y: 180
  point in direction 90
  forever loop
```

A **repeat** loop can be used to make the **Piñata** sprite repeat a small movement many times. This will make the piñata appear animated.

Drag a **repeat 10** block into the Code area and attach it inside your **forever** loop.



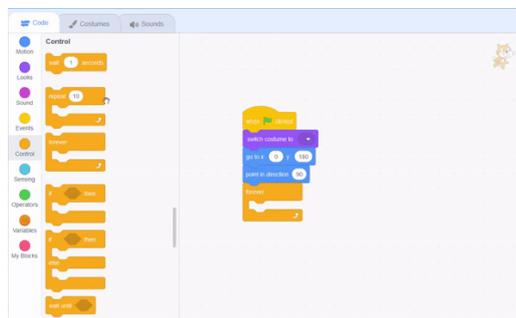
Go to the **Motion** blocks menu and drag a **turn clockwise 15 degrees** block into the **repeat** block.

Change the **15** degrees to **1** degree so that the piñata only swings a small amount each time:



```
when green flag clicked
  switch costume to whole
  go to x: 0 y: 180
  point in direction 90
  forever loop
    repeat 10
      turn 1 degrees
```

Change to 1

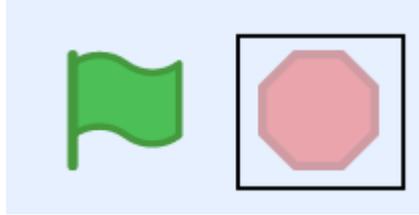


**Test:** Run your project, by clicking on the **green flag** above the Stage, to see the piñata swing.

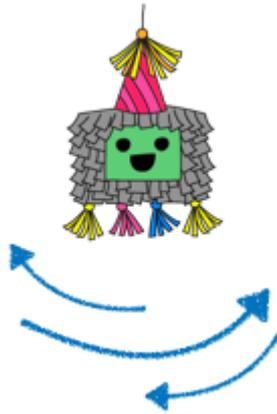


**Mmm, something is not quite right!** When an object is hung from the ceiling, it won't just rotate in one direction, it will swing back and forth.

Stop your project by clicking on the **red stop icon** above the Stage.



Add code to your **forever** loop so that the piñata swings from the centre back and forth continuously like a pendulum:



when  clicked

switch costume to whole ▾

go to x: 0 y: 180

point in direction 90

forever

repeat 10

Swings 10 degrees clockwise from the centre

turn  1 degrees

repeat 20

Swings 20 degrees anticlockwise through the centre

turn  1 degrees

Change to 1

repeat 10

Swings 10 degrees clockwise back to the centre

turn  1 degrees

Change to 1

**Test:** Run your project to see the piñata swing.



**Debug:** If the piñata does not swing correctly:

- Look at your code to make sure the **repeat** blocks are in the correct position
- Check that the **turn clockwise** and **turn anticlockwise** arrows are correct
- Make sure that you have used the numbers from the code above



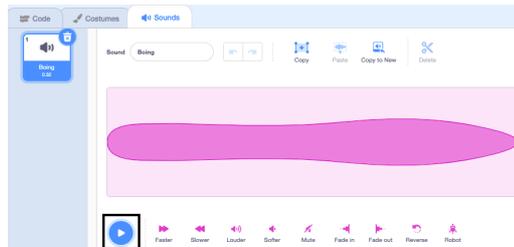
**Save your project**

### Step 3 Hit the piñata

In this step, you will code the piñata to play a sound and count one hit every time the piñata is clicked.



Click on the **Sounds** tab for the **Piñata** sprite and you will find a **Boing** sound. Click on the **Play** icon to hear the sound.

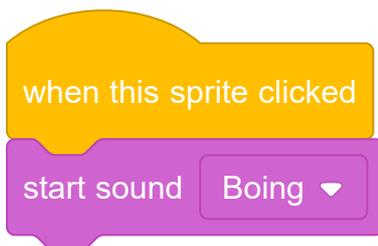


A group of connected blocks in Scratch is called a **script**. Sprites can have more than one script.

Click on the **Code** tab. From **Events**, drag a **when this sprite clicked** block into the Code area to start a new script.



In the **Sound** blocks menu, find the **start sound** block. Drag it underneath the **when this sprite clicked** block:



**Test:** Run your project by clicking on the **green flag** above the Stage. Click on the piñata as it swings to hear the boing sound.

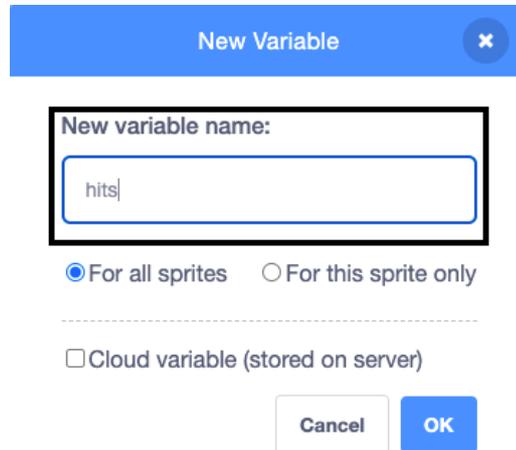


A **variable** is a way of storing numbers and/or text. The number of times the piñata is clicked will be stored in a variable called **hits** so it can be used at any time.

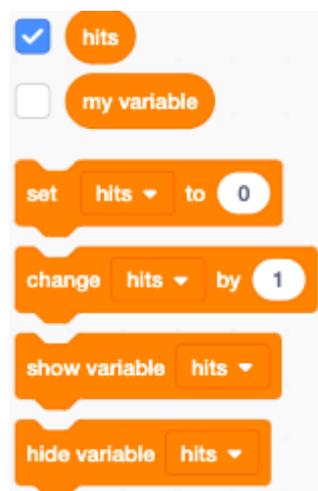
From the **variables** blocks menu, click the **Make a Variable** button.



Call your new variable **hits**:



**Notice:** The new 'hits' variable appears on the Stage and can now be used in the **variable** blocks.



Each time the project starts, the number of **hits** should be reset to **0**.



Drag the **set hits to 0** block into the first script in the Code area, between the **switch costume to** block and the **go to x: (0) y: (180)** block.

Your code should look like this:

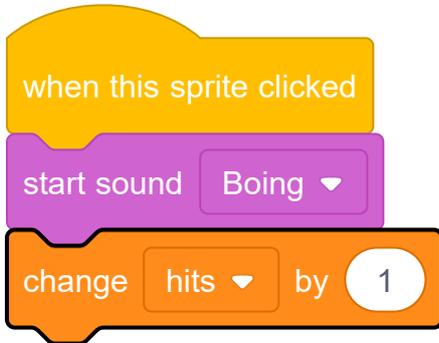


```
when green flag clicked
  switch costume to whole
  set hits to 0
  go to x: 0 y: 180
  point in direction 90
  forever
    repeat 10
      turn 1 degrees
    repeat 20
      turn 1 degrees
    repeat 10
      turn 1 degrees
```

Every time the **Piñata** sprite is clicked, the number of **hits** should increase.



Add a block to change **hits** by **1** when the **Piñata** sprite is clicked:



**Test:** Run your project a couple of times. Check that **hits** always starts at **0** and increases by **1** each time you click on the **Piñata** sprite.



A piñata is hard to break but it does not last forever. Your piñata will last for **10 hits** before breaking open.

An **if** block can be used to make a decision based on a **condition**.

We use **conditions** all the time to make decisions. We could say "if the pencil is blunt, then sharpen it". **If** blocks and conditions let us write code that does something different depending on whether a condition is true or false.

Go to the **Control** blocks menu. Drag an **if** block into the Code area and insert it around the blocks in your **when this sprite clicked** script:



```
when this sprite clicked
  if  then
    start sound Boing
    change hits by 1
```

The **if** block has a hexagon-shaped input where you can build a condition.

The **Piñata** sprite should play a sound and increase the count of **hits** **if** the number of **hits** is **less than 10**.



First add a **<** operator into the hexagon-shaped input:



```
when this sprite clicked
  if  <  then
    start sound Boing
    change hits by 1
```

Finish building the **if** condition by dragging in the **hits** variable to the left of the **<** operator and typing the value '10' on the right:



```
when this sprite clicked
if hits < 10 then
  start sound Boing
  change hits by 1
```

**Test:** Run your project again. Hit the piñata 10 times to hear the sound and see the **hits** variable increase.



Hit the piñata a few more times. The **hits** variable will not go above 10 because that condition is no longer 'true' so the code inside the **if** block won't run.

Add a second **if** block inside the first. This time the condition will check if **hits** = 10 and if 'true' the costume will change to **broken**:



```
when this sprite clicked
  if hits < 10 then
    start sound Boing
    change hits by 1
    if hits = 10 then
      switch costume to broken
```

**Test:** Run your project a couple of times. Check that the **Piñata** sprite starts with the 'whole' costume then changes to the 'broken' costume after **10 hits**.



When the **Piñata** sprite has broken, all the other sprites need to know that the party has started.

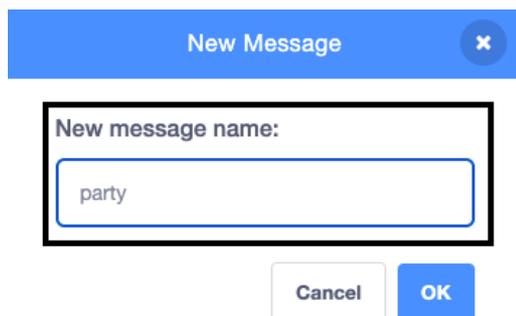
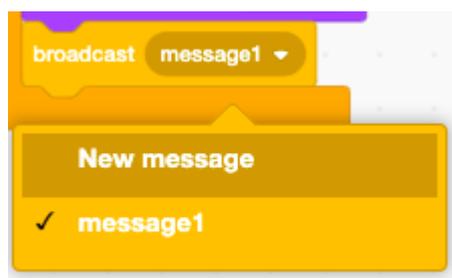
In Scratch, the **broadcast** block can be used to **send** a message that all sprites can **receive**.

Add a **broadcast message** block from the **Events** blocks menu:



```
when this sprite clicked
  if hits < 10 then
    start sound Boing
    change hits by 1
    if hits = 10 then
      switch costume to broken
      broadcast message1
```

Click on **message1** and choose **New message**. Name the new message **party**.



Your **broadcast** block will look like this:

broadcast

party ▼



**Save your project**

## Step 4 Use a stick

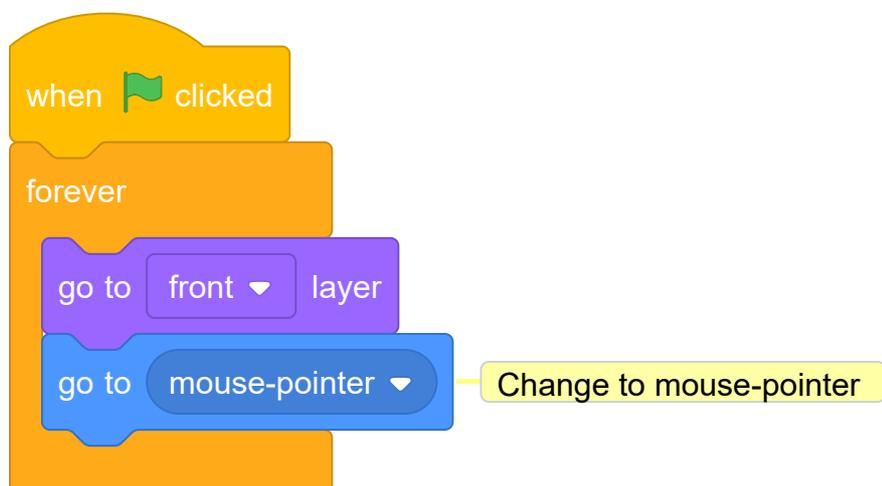
Piñatas are usually hit with a stick made from wood or thick card that is covered in colourful strips of paper. In this step, you will add code to control the piñata stick and play looping music when the piñata breaks.



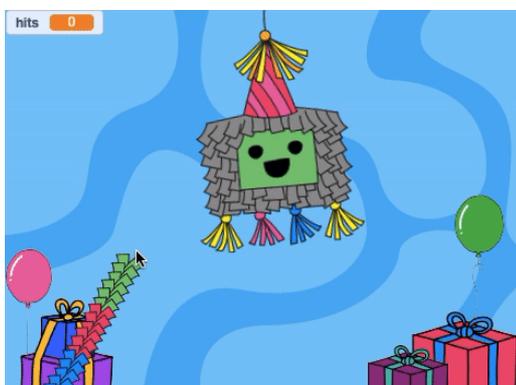
Click on the **Stick** sprite in the Sprite list. Add code so that the stick always stays in front of the other sprites and follows the mouse-pointer (or your finger on a tablet).



Use the **go to random position** block, but select **mouse-pointer** from the drop-down menu:



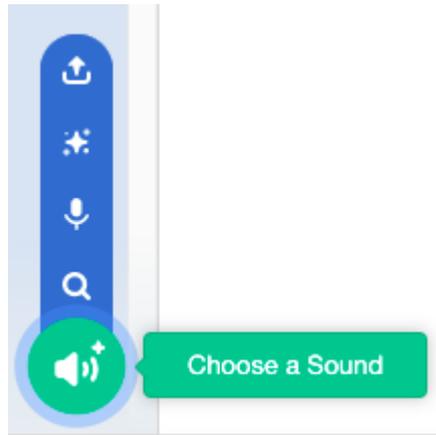
**Test:** Run your project and check the **Stick** sprite follows your cursor or finger around the Stage.



There are many different types of sounds in Scratch from voice and animal noises to over 100 other sound effects.

Scratch also has **looping sounds** that can be used in **forever** or **repeat** loops to sound like they are playing continuously.

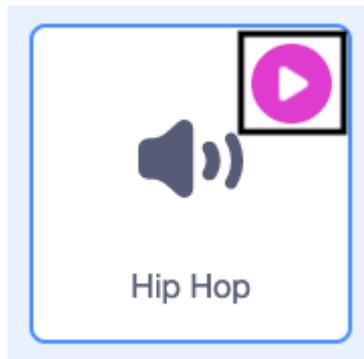
Go to the **Sounds** tab and click on the **Choose a sound** icon.



From the **Choose a sound** gallery, select the **Loops** category.



**Choose:** Hover over the **play** icons to hear the looping sounds. Add your favourite by clicking on it.



The sound will then appear in your Sounds list:



Click on the **Code** tab and create a new script to loop the sound **forever** when the **party** message has been received:



```
when I receive party
  forever
    play sound Hip Hop until done
```

Choose your sound

**Test:** Run your project, and click on the piñata ten times to hear the looping party music.



Save your project



Click on the **Piñata** sprite.

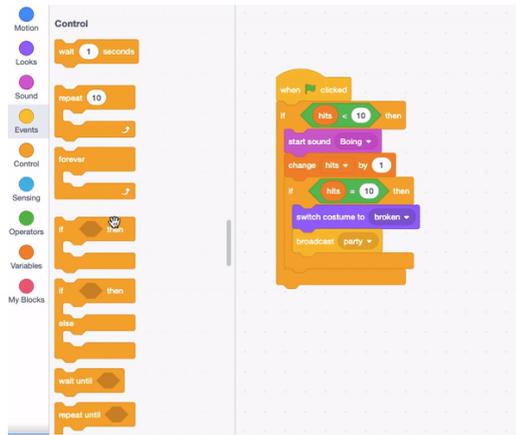


Insert a **repeat** loop into your existing code. Change the value to **4** then add a **create clone of myself** block. Use the drop-down arrow to select the **Treats** sprite:



```
when this sprite clicked
  if hits < 10 then
    start sound Boing
    change hits by 1
    repeat 4
      Change to 4
      create clone of Treats
      Select Treats
    if hits = 10 then
      switch costume to broken
      broadcast party
```

**Tip:** Use the spare space in the Code area to build your new code then drag it into the existing script:

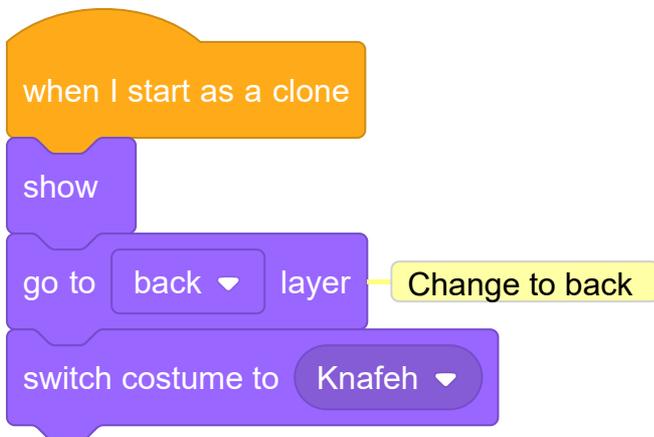


Click on the **Treats** sprite.



Create a new script using the **when I start as a clone** block.

Add blocks from the **Looks** blocks menu to control the appearance of each new clone:



You can pick random treats to be released when the piñata is hit. Use a **pick random** operator to select a random costume from 1 to 26 each time a clone is created:



```
when I start as a clone
  show
  go to back layer
  switch costume to pick random 1 to 26
```

Change to 26

At the moment, the **Treats** clones will appear behind the **Piñata** sprite, but treats should fall from the piñata to a random position.



Add code to make the cloned **Treats** sprites **glide** to a random position:



```
when I start as a clone
  show
  go to back layer
  switch costume to pick random 1 to 26
  glide 1 secs to random position
```

**Test:** Run your project and hit the piñata to see four clones of the **Treats** sprite after each hit. The costumes will be selected at random and the treats will each glide to a random position.



Add animation to make the **Treats** sprite clones **turn forever** when they reach their random position. Remember animations work best when small movements are used, so change the number of degrees to **1**:



```
when I start as a clone
  show
  go to back layer
  switch costume to pick random 1 to 26
  glide 1 secs to random position
  forever
    turn 1 degrees
```

**Test:** Run your project again to see the **Treats** sprite clones spin.



Save your project

## Step 6 Create a message

In this step, you will write a message and animate it using motion and colour effects.



What would you write in a birthday card to send to Code Club? It could be:

- Your favourite thing about Code Club
- A message about your fabulous Code Club leader
- Details of what you want to make next with your coding skills

The first Code Club projects were written in English, but within a year they had been translated into Brazilian Portuguese, Dutch, German, Norwegian, and Ukrainian. French, Greek, and Spanish translations quickly followed and now some of the Code Club projects have been translated into **28 native languages**. Thank you to our awesome translation community!



Click on the **Message** sprite in the Sprite list and select the **Costumes** tab.



The costume has some text saying 'Happy Birthday Code Club'. Double click (or tap and hold on a tablet) on the text to select the text editing tool.



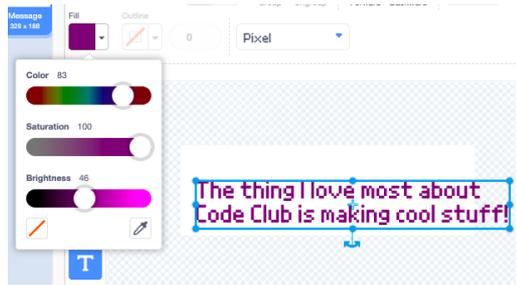
You can now type your new Code Club birthday message. Press **Enter** on your keyboard to start a new line.



**Tip:** Don't worry if your message is a bit too big for the box as you can resize it later.



**Choose:** Click on the **Fill** icon to open the colour drop-down menu. Move the fill sliders to the left or right to select your favourite colour.



**Choose:** Click on the **Font** tool and a drop-down list of fonts will appear. The 'Pixel' font is selected in the starter project, but you can use any of the fonts available.

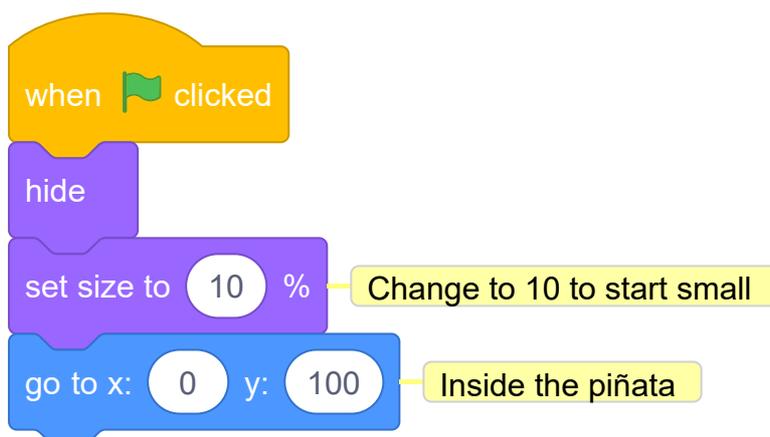


Click on the **Select** tool and eight circles will appear around your message. Use these circles to resize your message by clicking on them and dragging them within the white box.



Your message is ready, now you can add code to hide your message inside the piñata and make your message fall from the piñata after the tenth hit.

Click on the **Code** tab then create a script to **hide** the message in the piñata when your project starts:



Create a new script to start when the **party** message has been received.



Add a **repeat** loop to animate the message. The message will **change size** to grow and **change y** position to fall as it animates:

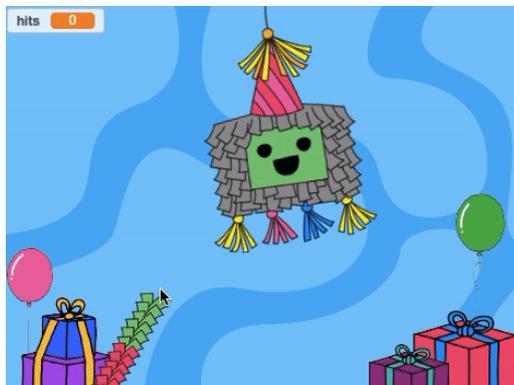
```
when I receive party
  show
  repeat 20
    change size by 5
    change y by -10
```

Change to 20

Change to 5

Change to -10

**Test:** Run your project. Hit the piñata ten times to see the message fall.



Save your project

## Step 7 Share

---

Now, it's time to share your project so other people can join the party.

If you have a Scratch account, you can share your project through Scratch. You can send a link to people you know, and the whole Scratch community will be able to find your project and try it out.



### Share your Scratch project

Please make sure that you **do not** share any personal information about yourself when sharing your Scratch projects.

- Give your Scratch project a name.
- Click the **Share** button to make the project public.
- If you like, you can add instructions in the **Instructions** box, to tell other people how to use your project.
- You can also fill in the **Notes and Credits** box: if you have made an original project, you can write some short comments, or if you have remixed a project, you can credit the original creator.
- Click the **Copy Link** button to get the link to your project. You can send this link to other people by email or text, or on social media.

Scratch provides the ability to comment on your own and other people's projects. If you do not want to allow people to comment on your project, you should turn off commenting. To turn off commenting, set the slider above the **Comments** box to **Commenting off**.



### Comments and feedback in Scratch

Scratch provides the ability to comment on your own and other people's projects. If you do not want to allow people to comment on your project, you should turn off commenting. To turn off commenting, go to the Project Page and set the slider above the **Comments** box to **Commenting off**:

If you are happy and feel safe to allow people to write comments on your project, you can leave the first comment:

If you think a comment or project is mean, insulting, too violent, or otherwise inappropriate, click the **Report** button to let the Scratch Team know about it. To report a comment, click the **Report** button above the comment. To report a project, click the **Report** button on the Project Page:

Read the **Scratch Community Guidelines** ([https://scratch.mit.edu/community\\_guidelines](https://scratch.mit.edu/community_guidelines)) so that you know how you and others can maintain a friendly and creative community.

### Submit your project



Inspire the global Code Club community with your project!

To submit your project to our **'Party piñata – Community' Scratch studio** (<https://scratch.mit.edu/studios/31111242>) for other people to see, please complete **this form** (<https://form.raspberrypi.org/f/community-project-submissions>).



## Step 8 Upgrade your project

---

If you have time, you can upgrade your project. You might have ideas about what to add already!

You could:

- Use additional looks, sounds, or motion blocks to animate your message and treats further
- Find images of your favourite treats and add them to the costumes for the **Treats** sprite
- Increase the number of treats that fall from the piñata on each hit
- Make your project easier or harder by changing the number of times you need to hit the piñata before it breaks

### Try it



Did you know you can add code to the backdrop too?

What happens to the backdrop when the piñata breaks? Which blocks would create this effect?

**See the code** (<https://scratch.mit.edu/projects/653771814/>)

You could add code to make the message animate **forever** when it is in position. Use **change size** and **change color effect** blocks so that it looks like the message is moving with the party beat:



```
when I receive party
  show
  repeat 20
    change size by 5
    change y by -10
  forever
    change size by 20
    change color effect by 25
    wait 0.5 seconds
    change size by -20
```

The code consists of the following blocks:

- when I receive** (party)
- show**
- repeat** (20)
  - change size by** (5)
  - change y by** (-10)
- forever**
  - change size by** (20) - Positive number to grow
  - change color effect by** (25) - Change colour
  - wait** (0.5) seconds - Try different numbers to match your music
  - change size by** (-20) - Negative number to shrink

See the code (<https://scratch.mit.edu/projects/656332454/>)



### Completed project

You can view the **completed project here** (<https://scratch.mit.edu/projects/649873783/>).

### Submit your project



If you have already shared your project with us just save your changes and we will see your awesome upgrades.

If you haven't yet shared your project but want to submit your project to our **'Party piñata – Community' Scratch studio** (<https://scratch.mit.edu/studios/31111242>) for other people to see, please complete **this form** (<https://form.raspberrypi.org/f/community-project-submissions>).

## Step 9 What next

---

We hope you have enjoyed this special Code Club birthday party piñata project.

If you are new to Scratch, continue exploring with our **Introduction to Scratch** (<https://projects.raspberrypi.org/en/pathways/scratch-intro>) path for beginners. 

If you already know about sprites, costumes, backdrops, and loops, try our **More Scratch** (<https://projects.raspberrypi.org/en/pathways/more-scratch>) path. 

To learn about lists, clones, and creating your own blocks, try our advanced **Further Scratch** (<https://projects.raspberrypi.org/en/pathways/further-scratch>) path. 



### Take part in Coolest Projects

Check out **Coolest Projects** (<https://coolestprojects.org/>), the world's leading technology showcase for young people! On the **Coolest Projects website** (<https://coolestprojects.org/>), you can find out when project registration is open, and get ready to register your project!

If you have made a Scratch project, you can register your project in the Scratch category when Coolest Projects project registration is open. Your project doesn't have to be finished – prototypes and works in progress are welcome too! When you have entered your project, your creation will be showcased in the Coolest Projects online gallery, for people all over the world to see! Join other young people in celebrating and recognising each other's achievements as a community.

You can use the Coolest Projects **How to make a project** (<https://coolestprojects.org/2020/03/31/how-to-make-a-project-workbook-and-additional-resources/>) resources to help you think of original project ideas.

If you need a reminder of what you have learnt, you can go to our **'Getting started with Scratch' guide** (<https://projects.raspberrypi.org/en/projects/getting-started-scratch>). 

---

Published by **Raspberry Pi Foundation** (<https://www.raspberrypi.org>) under a **Creative Commons license** (<https://creativecommons.org/licenses/by-sa/4.0/>).

View project & license on GitHub (<https://github.com/RaspberryPiLearning/party-pinata>)