



SCRATCH

Maths



SCRATCH

Maths



Scope & Sequence

SCRATCH

Maths

Module 3: Interacting Sprites

Investigation 1
Animating Sprites



Year 5 – Computing focus (20+ hours of teaching materials)

Module 1: Tiling Patterns



Module 2: Beetle Geometry



Module 3: Animating Sprites



Year 6 – Mathematics focus (20+ hours of teaching materials)

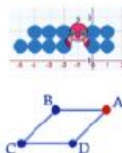
Module 4: Building with Numbers



Module 5: Exploring Mathematical Relationships



Module 6: Coordinates and Geometry



Module 3: Animating Sprites

Investigation 1 Animating Sprites



Investigation 3 Broadcasting Messages



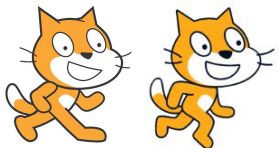
Investigation 2 Encountering Conditions



Investigation 4 Interactive Stories



SCRATCH Vocab



costume: a version of a sprite's appearance

Animation

is an illusion created by showing different costumes in sequence



when this sprite clicked

that block that makes the script conditional on the sprite being clicked



repeat 10

this block **repeats** the blocks inside it for the specified number of times



show



hide


show: this block will show the sprite on the stage if it is currently shown

hide: this block will hide the sprite on the stage if it is currently shown



change x by 10

this block changes the **y position** of the sprite by the specified amount



if on edge, bounce

checks if the sprite is **touching the edge** of the stage and if true, reverses the direction the sprite is pointing



forever

this block **repeats** the blocks inside it forever (until the stop sign is clicked)

Code

Costumes

Sounds



Motion

Looks

Sound

Events

Control

Sensing

Operators

Variables

My Blocks

Pen

Motion

move 10 steps

turn 15 degrees

turn 15 degrees

go to random position

go to x: 0 y: 0

glide 1 secs to random position

glide 1 secs to x: 0 y: 0

point in direction 90

point towards mouse-pointer

change x by 10

set x to 0

Blocks

when clicked

go to x: 0 y: 0

erase all

clears everything and starts again from the beginning - DO NOT EDIT

stamp

Scripts area



Stage

Sprite

Title



x

0



y

0

Stage

Show



Size

100

Direction

0



Sprites

Backdrops

1

SCRATCH

Maths



Investigation 1
Animating Sprites





MODULE 3: INVESTIGATION 1

Activity 3.1.1 – Multiple Sprites

Learning Intention: Explore projects with multiple sprites and scripts initiated by different events (sequential and parallel) and bridging to knowledge of coordinates.

Success Criteria:

- Use multiple costumes to animate a sprite;
- Build parallel scripts for different sprites
- Use knowledge of coordinates to position sprites

This investigation introduces the concept of multiple sprites and parallel scripts, events, hide and show plus other graphic effects as well as using repeat to control the speed of the sprite's behaviour. It also introduces sprites jumping and walking in the stage, using different costumes to animate their shapes.

Investigation 1

Animating Sprites



Multiple
Sprites

Starter project:
**31-Multiple
Sprites**

Teleporting
Nano

Continue with
**31-Multiple
Sprites**

Jumping
Tera

Continue with
**31-Multiple
Sprites**

Walking Pico

Continue with
**31-Multiple
Sprites**

change x by 10

forever

show

hide

ACTIVITY INSTRUCTIONS

- 1 Open project [31-Multiple Sprites](#), remix it, add your name to the title, save the project, and share it to our class studio, [Investigation 1](#).
- 2 Explore the project and discuss what you can see (see the top three discussion points below). Click each of the sprites and discuss what happens and why.
- 3 Click on the green flag, see what happens and think why (note – there is no script to define any behaviour, so nothing happens whatever is clicked).

The [31-Multiple Sprites](#) project will be used throughout the whole of Module 3, building behaviours for the different sprites. Therefore a *setup script* needs to be built to return each sprite to his/her **initial position** when the green flag is clicked, i.e. the positions they started in when the project was opened (more blocks will be added to these scripts later in the module).

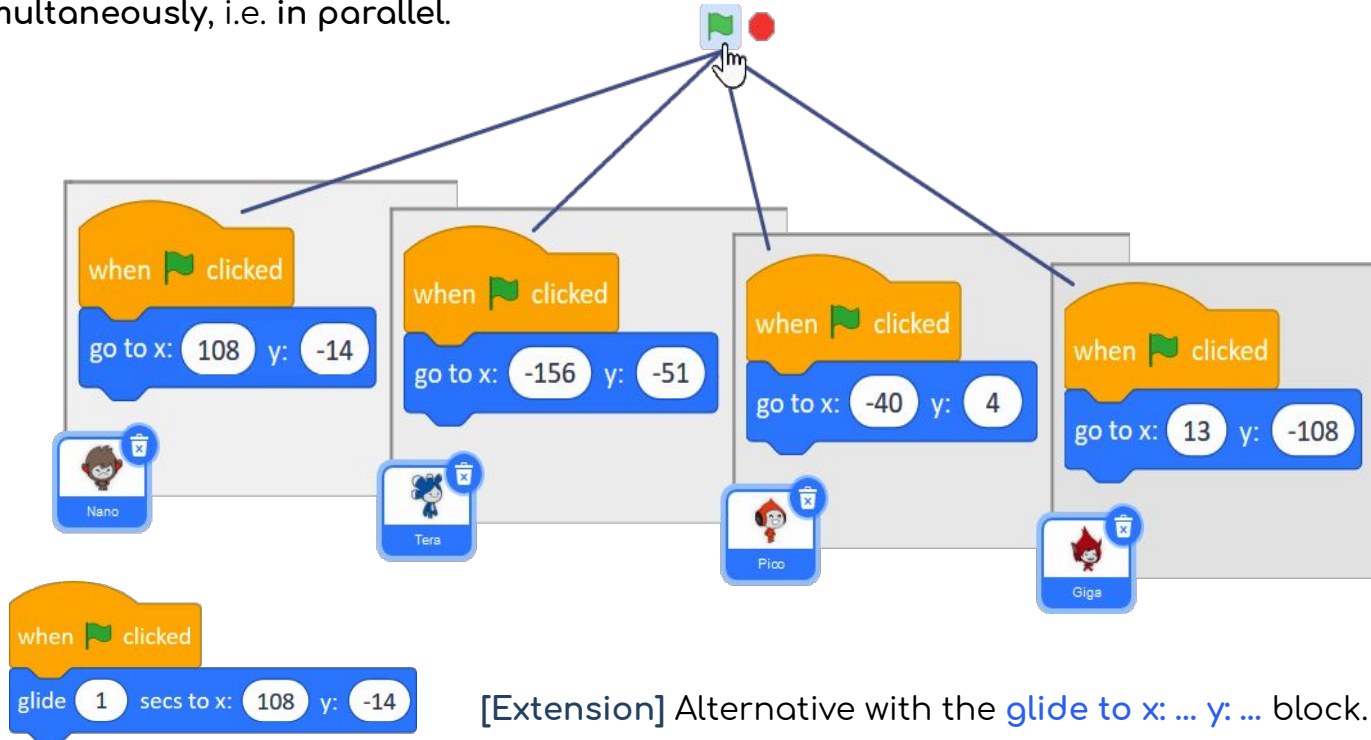
- 4 Add the block **when green flag clicked** to each of the sprites and snap the **go to x: ... y: ...** block to this hat block, ensuring the x and y values are set to the correct initial position.
- 5 [Extension] Alternatively, pupils may replace the **go to x: ... y: ...** block with the **glide ... secs to x: ... y: ...** block with the same coordinates. They can experiment with different times (e.g. 0.5, 1 or 2 secs), dragging the sprites from their initial positions then clicking the green flag.



EXTRA HELP CONT.

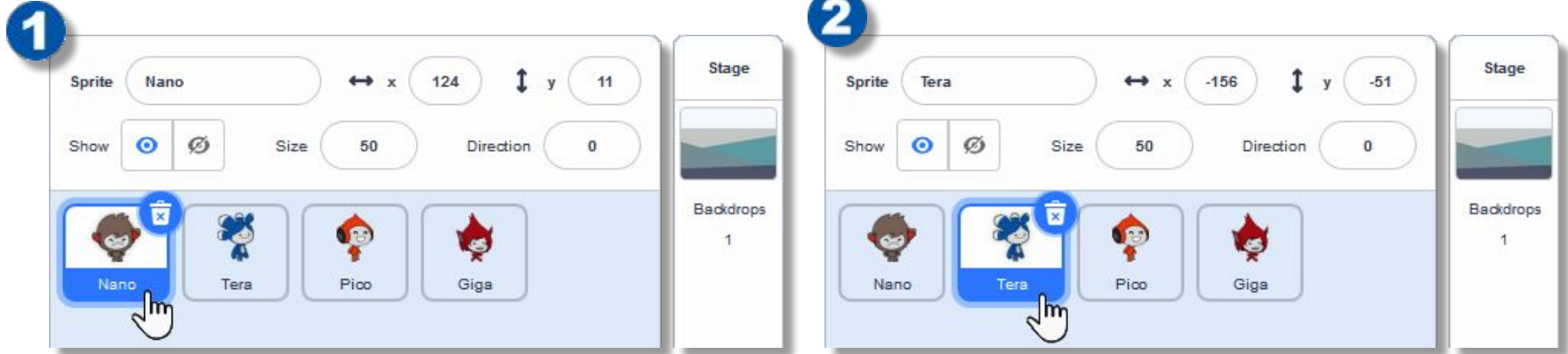
Below are the example scripts to return each sprite to its initial position. Note that:

- ▶ If you click one of these scripts directly in the scripts area, only that script is run.
- ▶ However, if you click the green flag, all **when green flag clicked** scripts (four in this case) are run simultaneously, i.e. in parallel.



Note that each sprite has a name, its own costume, and its own scripts area. To build the script(s) or explore them, we first have to click that particular sprite's icon (see 1 and 2 below). When we click the icon, the scripts of that sprite are displayed in the scripts area.

Each script belongs only to one sprite.



Finding the coordinates of a sprite

Within the Motion group, the [go to x: ... y: ...](#) block is automatically set with the current coordinates of that sprite. Drag the sprite within the stage and note that the coordinates in the [go to x: ... y: ...](#) block (and also in the [glide to ...](#) block) are updated whenever you release the mouse. To create a *setup script* to return the sprite to its initial position, ensure the sprite is in the position you want it to return to, drag the [go to x: ... y: ...](#) block to the scripts area and snap it onto the *setup script*.



MODULE 3 • INVESTIGATION 1 • ACTIVITY 3.1.2

Teleporting Nano

Learning Intentions:

Explore how to make a sprite jump to a random position.

Explain different strategies for how to hide and show a sprite.

BridgE to mathematical fluency using number fact knowledge and choosing appropriate strategies.

Success Criteria:

- Control a sprite jumping and moving with script.

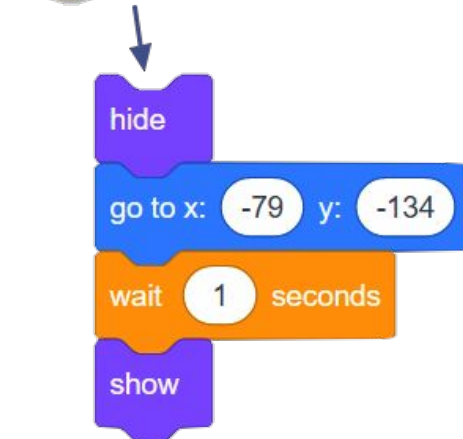
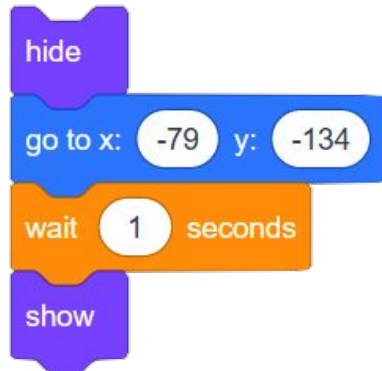
Investigation 1 Animating Sprites



ACTIVITY INSTRUCTIONS

- 1 Continue in your project **31-Multiple Sprites**. The following activities develop different behaviours for each sprite. Start with Nano who has a special ability: he can vanish and reappear somewhere else on the stage whenever he is clicked.
- 2 Select Nano, drag the **hide** and **show** blocks from the Looks group in the scripts area and explore them in isolation.

Hint: start by dragging the **hide** and **show** blocks into the scripts area and explore them first as isolated blocks, before using them in the script.



Improved teleporting with jumping to random x and y positions.



ACTIVITY INSTRUCTIONS

- 3 Pupils build a script to initiate Nano's special ability, starting with the **hide** block. They add the **go to x: ... y: ...** block and the **wait 1 secs** block (otherwise Nano would reappear instantly). They set the x and y positions to the location on the stage they want Nano to teleport to. Finally they add the **show** block to the bottom of the script.
- 4 Run the script by clicking it. If it works correctly, drag the **when this sprite clicked** hat block into the scripts area and snap it to the top of the script (for basic teleporting script see 1 in additional support). However, this teleporting script will work only when Nano is clicked for the first time. Each additional click will then make Nano reappear in the same place.



ACTIVITY INSTRUCTIONS

- 5 To make Nano always reappear in a random position, replace the values of both x and y in the **go to x: ... y: ...** block with the **pick random ... to ...** blocks. In each of them they set the furthest points of the stage (from left to right for x and from bottom to top for y) where they would want Nano to reappear (see 2 in additional support).

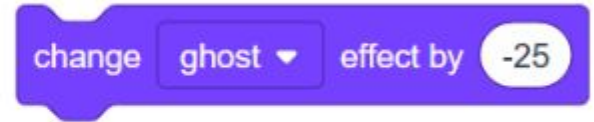
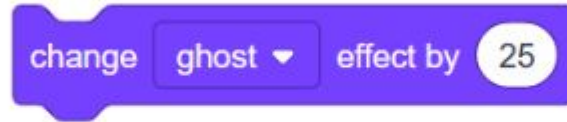


THINGS TO NOTE

- ◆ The x axis ranges from -240 to 240 and the y axis from -180 to 180.
- ◆ Clicking the red stop sign will reset all graphic effects (including the ghost effect).

ACTIVITY INSTRUCTIONS

- 6 [Extension] Make Nano disappear and reappear slowly by using the **change ghost effect by ...** block. Build your own block **disappear** and use it instead of the **hide** block. Then build your own block **reappear** for the reverse process and use it instead of the **show** block (see 3 and 4 in additional support).



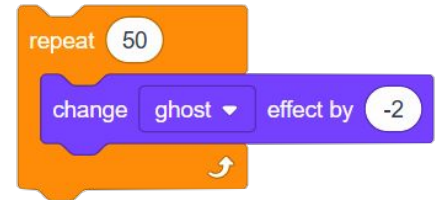
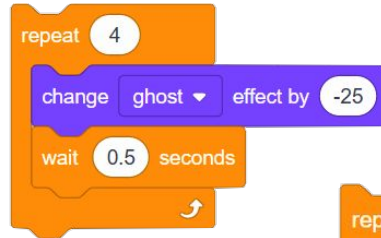
Click the **change ghost effect by ...** block again and again. Then explore the **reverse process** by changing 25 into -25.

Put the **repeat** block around the **change ghost effect by ...** block to enable these processes to happen in one click.



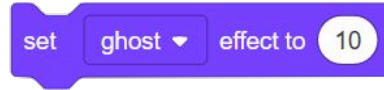
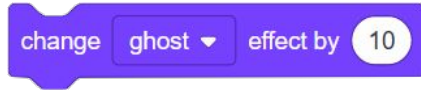
To slow down the process of disappearing and reappearing, add the **wait ... secs** block with a small number inside the **repeat**. Note, this process is not smooth.

The speed of Nano disappearing and reappearing can be made slowed down and be smooth (or even very slow) by modifying the number in the **repeat** block and the **change ghost effect by ...** value.



EXTENSION TO ACTIVITY 3.1.2

Explore the difference between the **change ghost effect by ...** and **set ghost effect to ...** blocks. Explore. Drag them as isolated blocks in the scripts area, set both values to 10 and click each block repeatedly several times. Think about what happens.



standing



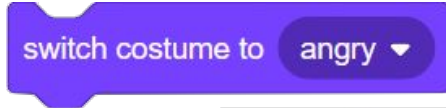
talking



laughing



angry



DISCUSSION POINTS

- ◆ What exactly does the **hide** block do? What would happen if the **show** block is removed from your script – where would Nano be?
- ◆ What other way could you 'hide' the sprite? (*setting its ghost effect to 100*)
- ◆ What values did you select for x and y? How did you select these values?
- ◆ What would happen if you selected only negative numbers for x and y?





MODULE 3 • INVESTIGATION 1 • ACTIVITY 3.1.3

Jumping Tera

Learning Intention:

Explore how to change the y position to move the sprite up and down.
Explain how the sprite can be moved at different “speeds”.
bridge to mathematical fluency (using number fact knowledge) and reasoning (doing/undoing).

Success Criteria:

- Move the sprite’s position without dragging it;
- Move the sprite at different speeds using script

Investigation 1 Animating Sprites

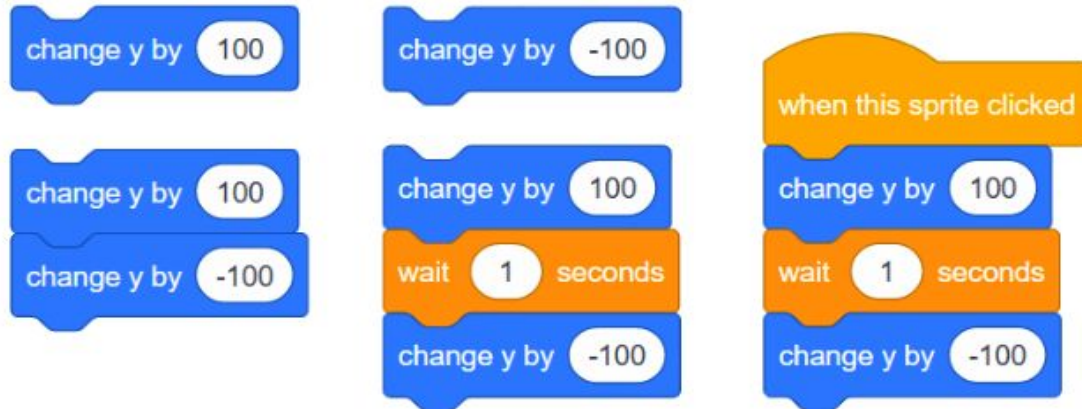


ACTIVITY INSTRUCTIONS

- 1 Continue in your project **31-Multiple Sprites**.

Tera also has a special ability: when she is clicked, she can jump really high, And then return back down to the same position.

- 2 Select Tera and explore one isolated **change y by ...** block, changing the values in it and clicking it. You may try negative inputs to see what happens.
- 3 Start to build a script to give Tera her special ability. In the **change y by ...** block set the value you want Tera to jump (e.g. 100). Next, drag a second **change y by ...** block into the scripts area, keeping it isolated from the first one, type in the value to allow Tera to **return to her starting position**, then explore both blocks by clicking them alternately.



ACTIVITY INSTRUCTIONS

- 1 Snap together both **change y by ...** blocks and click this short script to see what happens. It appears that *nothing* happens as Tera jumps up and down almost instantly. So add in the **wait 1 secs** block between two **change y by ...** blocks to see Tera jump (see additional support). Add the hat block **when this sprite clicked** on top of your script and test it by clicking Tera.

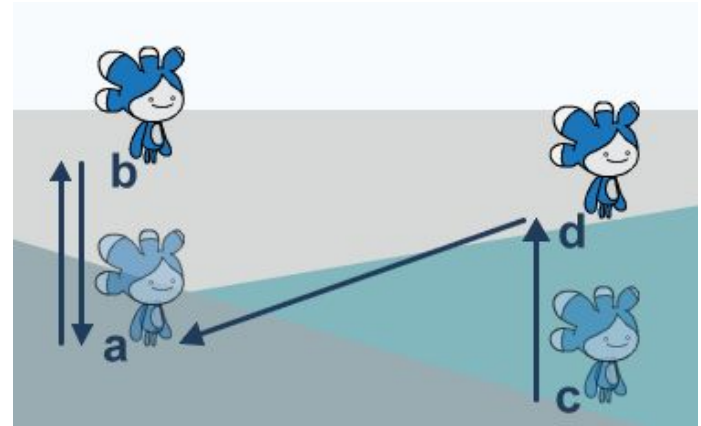
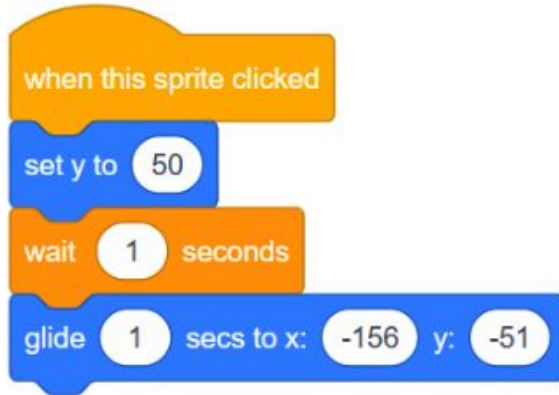
Now make Tera jump back more smoothly, as if she *floats in the air*.

- 5 Add a **repeat** block around the second **change y by ...**. Edit your script to move Tera down more slowly but still over the same total distance, for example by changing y by -5 each time and repeating this 20 times, or changing y by -2, repeating it 50 times (see additional support).



ACTIVITY INSTRUCTIONS

- [Extension] Make Tera jump up smoothly as well, but not as slowly as when floating down. Do this by adding one more **repeat**, this time around the first **change y by ...** block and set the correct values both in **repeat** and **change y by ...**.
- [Extension] Use different costumes within her jump (see additional support).

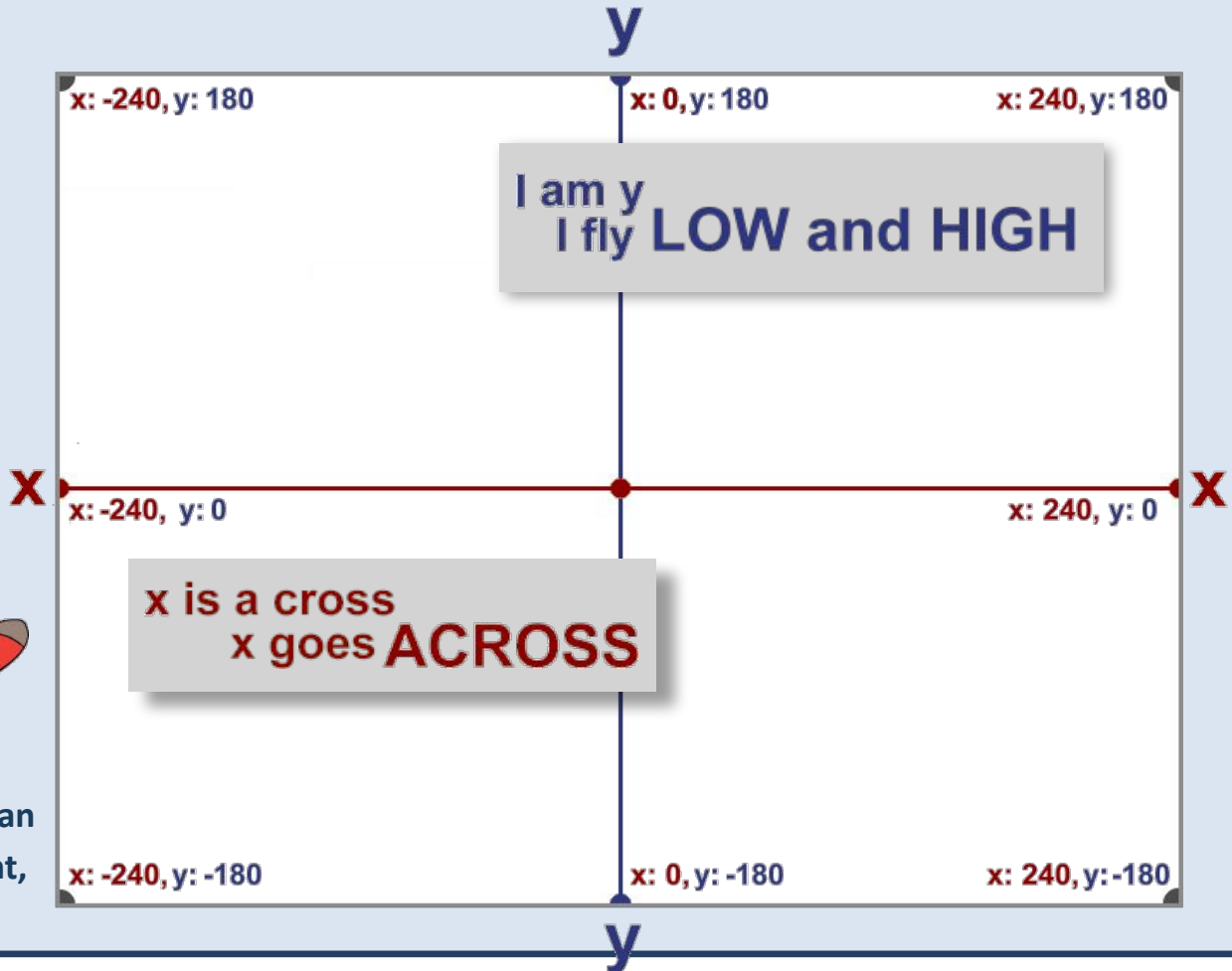




Hello I am Tera.
I can jump to
any y , low or high!



Hello I am Nano. I can
teleport to any point,
to any x and y !





MODULE 3 • INVESTIGATION 1 • ACTIVITY 3.1.4

Walking Pico

Learning Intentions:

Explore how to switch between multiple costumes to animate a sprite.

Explore how to run a script forever.

Explain how to ensure a sprite stays within the stage area.

bridgE to mathematical reasoning (always, sometimes, never).

Success Criteria:

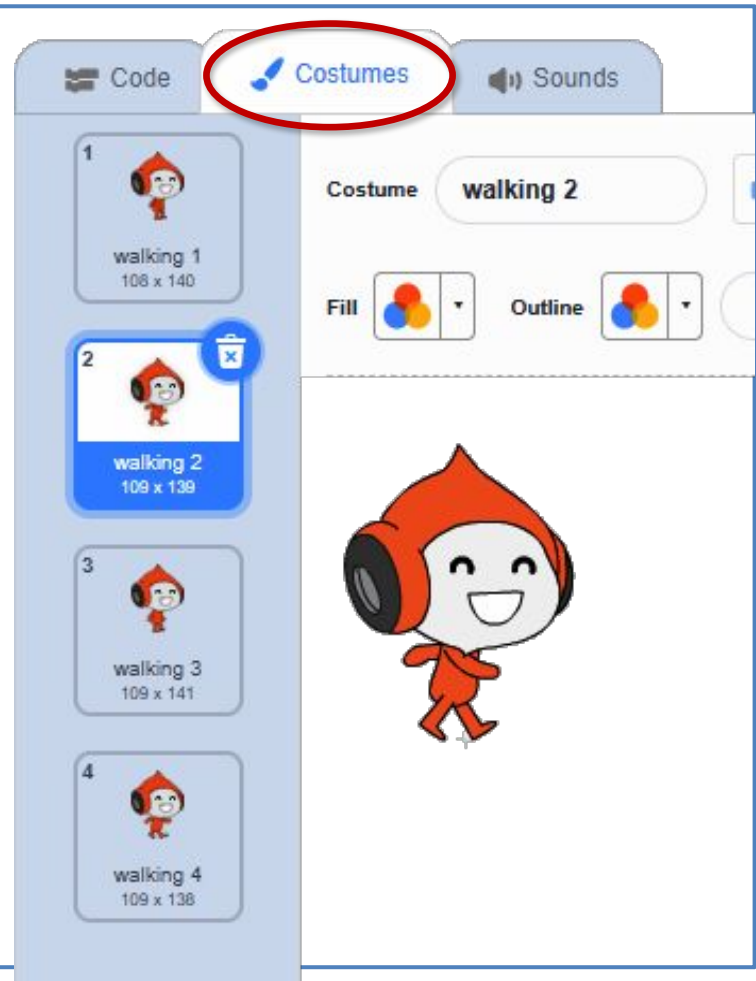
- Walk a sprite around the stage.

Investigation 1 Animating Sprites



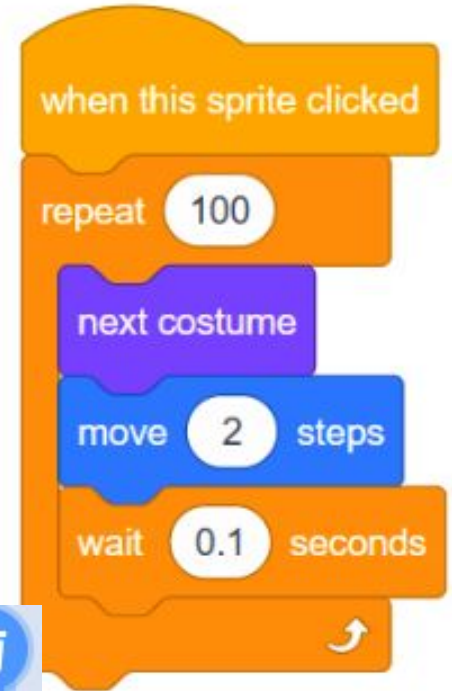
ACTIVITY INSTRUCTIONS

- 1 Continue in your project **31-Multiple Sprites**.
Pico has a special ability: he can walk around the stage.
- 2 Select Pico and explore his costumes in the Costumes tab by clicking on them one by one to see how he can be animated i.e. how he can be made to walk.
- 3 Drag the **next costume** block into the scripts area (keeping it isolated), then click it repeatedly and observe Pico.



ACTIVITY INSTRUCTIONS

- 4 Add a **repeat** block around this block and set its value to e.g. 100.
- 5 Make Pico walk by adding a **move** block inside **repeat** and set its value to a small number, e.g. 1 or 2. Click the script and observe the behaviour. You may decide to add a **wait** block with very short wait time, e.g. 0.1 or 0.2 seconds, then add the **when this sprite clicked** hat block on top of your script.



ACTIVITY INSTRUCTIONS

- 6 Click on Pico and observe him walking. Click Pico again to see what happens when he touches the edge of the stage. Add the **if on edge, bounce** block within the **repeat** block and then change the **repeat** block to the **forever** block.



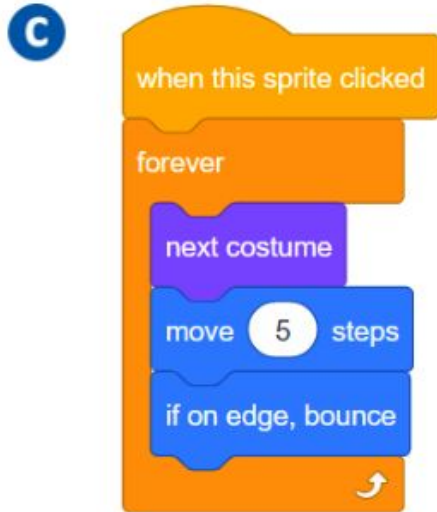
The **if on edge, bounce** block would be run only once if we add it on top or below the **repeat** block. If we want Pico to bounce whenever he touches the edge, this special block must be added inside **repeat**. Think of the **forever** block as a version of **repeat** with a “very big” value of repetition.



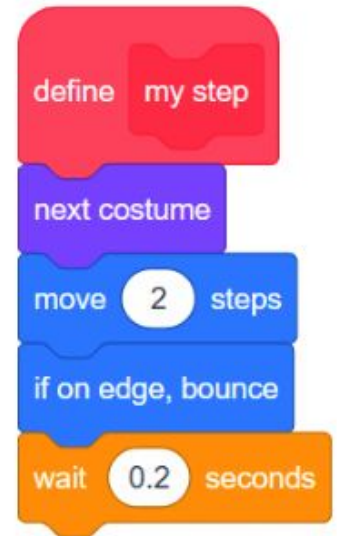
ACTIVITY INSTRUCTIONS

6 [Extension] Explore different values in the **move** and **wait** blocks to make Pico walk faster or slower (walk, stride, march, run).

c – Running and marching scripts



d – Solution with a new block



DISCUSSION POINTS

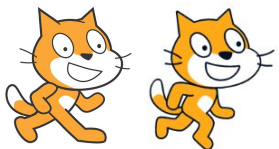
- ◆ How is the illusion of walking created?
- ◆ What happens when the sprite touches the edge? What does bounce mean?
- ◆ What happens if the **if on edge, bounce** block is placed outside of the **repeat** block, before or after?
- ◆ What is the difference between the **repeat** and **forever** blocks?
- ◆ Why can you not connect any blocks to the bottom of the **forever** block?
- ◆ How did you make Pico walk faster or slower?
- ◆ What direction does Pico walk in when he bounces off the edge of the stage? How does the bouncing work?

My Module 3: **Investigation 1** check list:

- I built a setup script for each of the sprites.
- I made Nano hide and then show somewhere else in the stage.
- I made Nano reappear in a random position.
- [Extension] I used ghost effect to make Nano disappear gradually.
- I made Tera jump high and back by changing her y position.
- I made Tera float down by repeating several smaller jumps.
- [Extension] I used the next costume and/or switch costume blocks to change the costumes of my sprites.
- I made Pico walk and bounce off the edge.
- I made Pico walk and bounce forever.



SCRATCH Vocab



costume: a version of a sprite's appearance

Animation

is an illusion created by showing different costumes in sequence



when this sprite clicked

that block that makes the script conditional on the sprite being clicked



repeat 10

this block **repeats** the blocks inside it for the specified number of times



show



hide

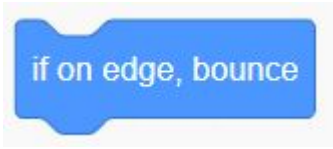
show: this block will show the sprite on the stage if it is currently shown

hide: this block will hide the sprite on the stage if it is currently shown



change x by 10

this block changes the **y position** of the sprite by the specified amount



if on edge, bounce

checks if the sprite is **touching the edge** of the stage and if true, reverses the direction the sprite is pointing



forever

this block **repeats** the blocks inside it forever (until the stop sign is clicked)



End of Investigation 1



INTERACTING SPRITES

MODULE 3: INVESTIGATION 2

Learning Intention: Explore conditions and the ways they are used to control the flow of the sprites' behaviour. Discover how to sense the current situation of a particular sprite.

Success Criteria:

- Explore how to make a sprite react to specific conditions.
- Explain how to execute a script until a condition is true.

This investigation introduces conditions and two new control structures: the first control structure runs a block or several blocks only if a certain condition is true, and the second repeats several blocks only until certain condition is true. These structures allow the building of scripts which sense the current situation of the sprites and then react accordingly.

Investigation 2

Encountering Conditions



Repeat until

Continue with
**31-Multiple
Sprites**
or start with
**32-Multiple
Sprites**

Touching Colour?

Continue with
**31-Multiple
Sprites**

Walking in the air

Continue with
**31-Multiple
Sprites**

Unplugged: True or False?

SCRATCH Vocab

Condition

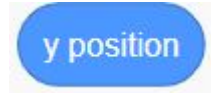
is a block reporting whether something is true or false



makes the sprite set its direction towards another sprite (selected from the drop down menu)



repeats the blocks inside it until its condition is true



reports the current y position of a sprite



reports true if the first value is greater than the second



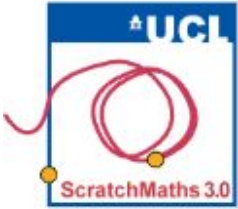
reports true if the sprite is touching another sprite (selected from the drop down menu)



reports true if the sprite is touching a specified colour



if the condition is true, runs the blocks inside



MODULE 3 • INVESTIGATION 2 • ACTIVITY 3.2.1



Repeat Until...

Learning Intentions:

Explain how to point one sprite towards another sprite.

Explore how to execute a script until a specified condition is true.

bridge to mathematical reasoning (always, sometimes, never).

Success Criteria:

- Make a sprite meet another sprite.
- Make sprites react conditionally to each other.

Investigation 2

Meeting conditions

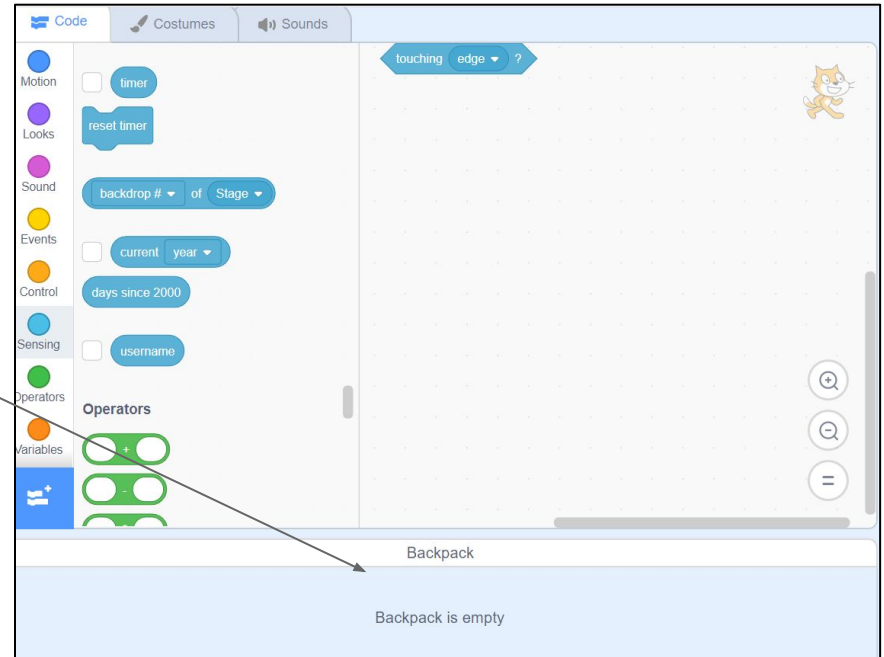


ACTIVITY INSTRUCTIONS

- 1 Continue in your project **31-Multiple Sprites**.
First teach Giga to walk in the same way as Pico.
- 2 Select Giga and build the same behaviour for her as for Pico in activity 3.1.4.
(Alternatively, you may select Pico and directly **copy his walking script** to Giga by dragging his script and dropping it on the picture of Giga within the sprites area.)



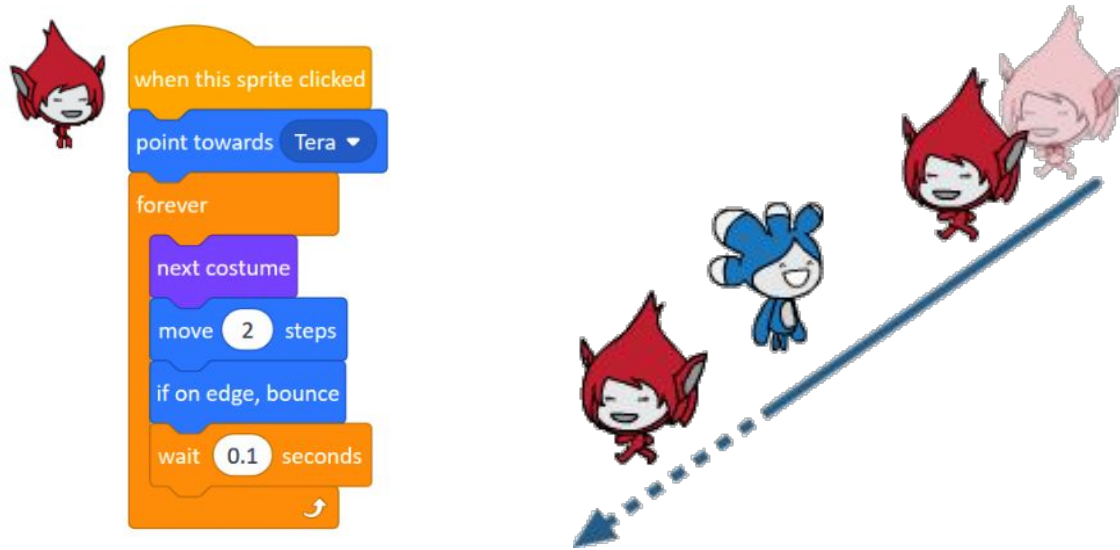
Hint: you can also use the backpack feature at the bottom of the Scratch stage to copy and paste your code sequences between projects!



ACTIVITY INSTRUCTIONS

Now adapt Giga's walking script to walk **towards Tera**, so she can deliver a message.

- 3 Click Giga and keep the walking script running. Drag the **point towards ...** block from the Motion group into the scripts area. Keep it isolated from other scripts and explore how it works.
- 4 Open the drop down menu of the block, select **Tera** and click the block (see additional support). While Giga is walking, explore the isolated **point towards ...** block by opening the drop down menu, selecting **Nano** or **Pico** as a new destination for Giga to walk towards and then clicking the block.



- 5 Snap the **point towards ...** block into the walking script (ensuring it is now set back to **Tera**), just in front of **forever**, so that whenever Giga starts walking, she first sets her direction towards Tera. Observe what happens when Giga reaches Tera (see additional support).

ACTIVITY INSTRUCTIONS

The screenshot shows the Scratch Sprites area. At the top, the selected sprite is 'Giga'. Below it, the 'Show' button is active (indicated by a blue eye icon). The 'Size' is set to 50 and the 'Direction' is -83. The coordinates are x: 13 and y: -108. Below the settings, there are four sprite thumbnails: Nano (a monkey), Tera (a blue character), Pico (a red character), and Giga (a red character with a blue trash icon). The Giga sprite is highlighted with a blue border.

Remember that all scripts and blocks in the scripts area belong to **the sprite which is currently selected** in the sprites area. That is, each sprite has its own scripts area.

The diagram illustrates the 'touching' block behavior. It shows three examples of the 'touching' block with a dropdown menu:

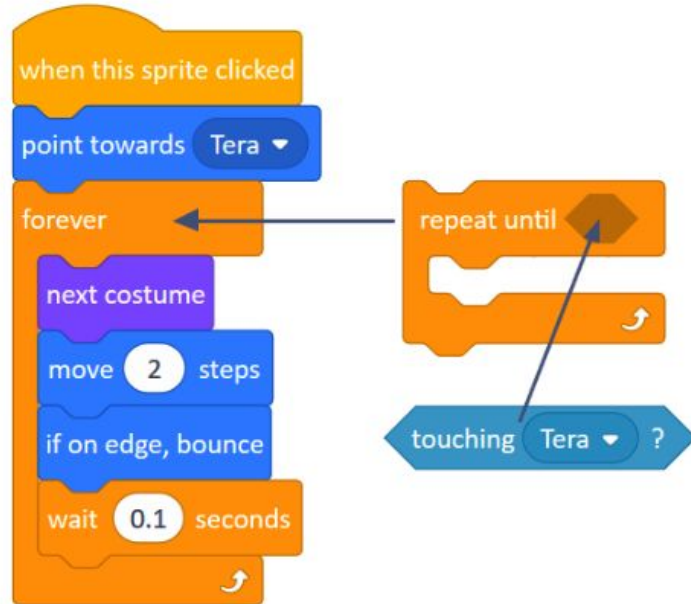
- Block 1: 'touching mouse-pointer' dropdown. A menu is open showing options: 'mouse-pointer' (checked), 'edge', 'Nano', 'Tera', and 'Pico'.
- Block 2: 'touching Tera' dropdown. A mouse cursor is hovering over the block, and a callout box shows 'false'. Below the block are the Tera and Giga sprites.
- Block 3: 'touching Tera' dropdown. A mouse cursor is hovering over the block, and a callout box shows 'true'. Below the block are the Tera and Giga sprites, with the Giga sprite overlapping the Tera sprite.

ACTIVITY INSTRUCTIONS

However, we want Giga to stop when she reaches Tera and not to continue walking through her!

- 6 Replace **forever** in the walking script with the **repeat until ...** block. Now you need a block that checks whether Giga is touching Tera or not: **the condition**.
- 7 Drag the **touching ... ?** block from Sensing, keeping it isolated, select Tera from its drop down menu and click the **touching Tera ?** Block. It will say *true* or *false* (see additional support).
- 8 Add the **touching Tera?** block into **repeat until ...** and test the script. Giga should now stop when she touches Tera (see additional support).

You can explore this block by keeping Giga walking and repeatedly re-selecting the destination sprite and clicking the **point towards ...** block several times.



ACTIVITY INSTRUCTIONS

Below are the three different ways for *repeating several blocks*. Think about the differences between the following three scripts.





MODULE 3 • INVESTIGATION 2 • ACTIVITY 3.2.2

Touching Colour?



Learning Intentions:

Explore how a sprite can react to touching a specific colour.
Explore how to run a script only if a certain condition is true.
bridgE to mathematical problem-solving and generalising.

Success Criteria:

- Make a sprite react when touching a given colour..

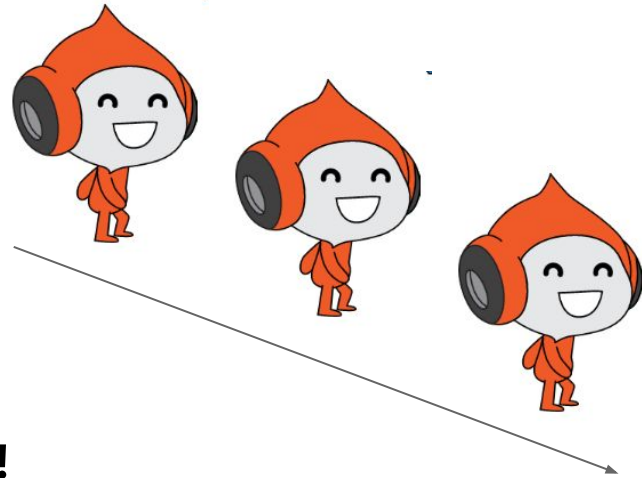
Investigation 2

Meeting conditions



ACTIVITY INSTRUCTIONS

- 1 Continue in your project **31-Multiple Sprites** and select Pico.
- 2 Add the block **point in direction ...** under the **when this sprite clicked** hat block and set this to a random value (e.g. between 45 and 135) using the **pick random ... to ...** block. This means that Pico will start walking in a random direction, not only horizontally, bouncing from the edges.



But!

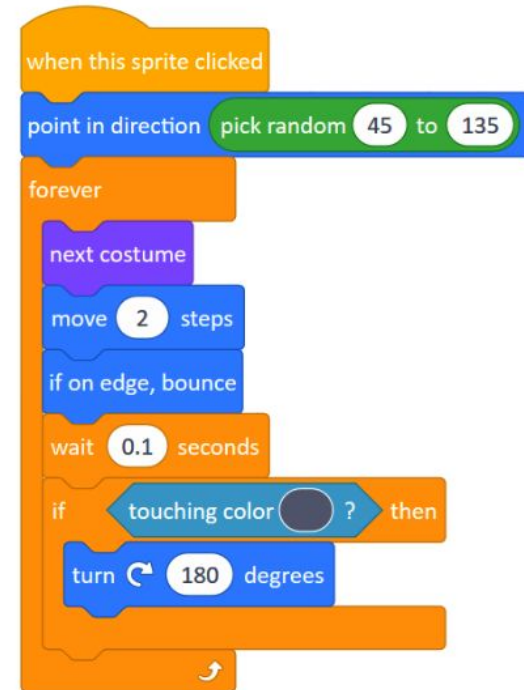
We want Pico to walk forever, but **turn back** when he reaches certain colour.

ACTIVITY INSTRUCTIONS

- 3 Drag the new **Sensing** block **touching color ... ?** into the scripts area, keeping it isolated and click on the square of colour in the block. The mouse pointer should turn to a hand – a 'colour picker'.
- 4 Choose a colour of the backdrop at which Pico should turn back (e.g. dark grey). The block will now act as a condition checking whether Pico is touching that particular colour.



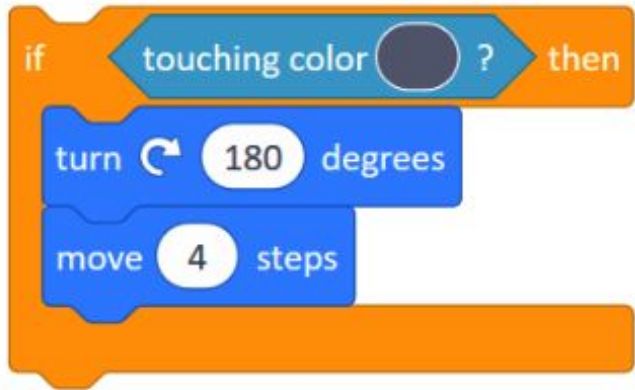
- 5 Check the block by dragging Pico onto different colours on the backdrop and clicking the **touching color ... ?** block to see if it says *true* or *false*. Try changing the colour in your block and then dragging Pico around and clicking the block again.



ACTIVITY INSTRUCTIONS

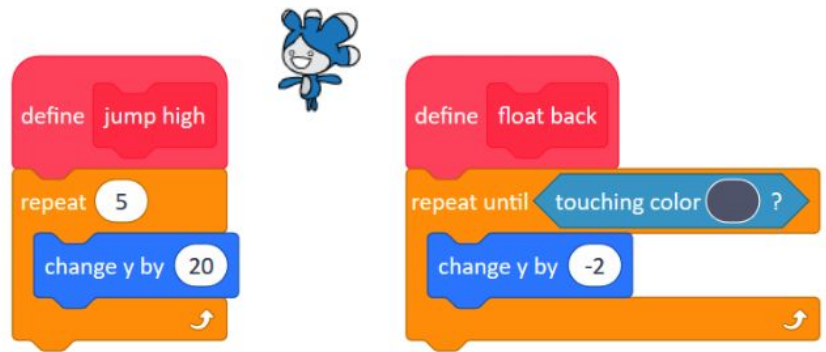
You now have a condition to recognize a particular colour. Next add an action to react in the case the condition is true.

- 6 Drag the **if <condition> then ...** block from the Control group, keeping it isolated and adding the **touching color ... ?** condition . Add the **turn right 180 degrees** inside it.
- 7 While Pico is still walking, click the **if <condition> then ...** block and observe what happens. Clicking it once means checking the condition once whether or not Pico is touching that particular colour. If you insert it inside the **forever** block of the walking script the condition will be checked again and again.



Extension

Below is one possible solution of Tera jumping high then floating down until she touches a particular colour of the backdrop





MODULE 3 • INVESTIGATION 2 • ACTIVITY 3.2.3

Walking in the Air



Learning Intentions:

Explore how to make a sprite react to reaching a specific vertical position on the stage.

Explain how to check whether the position of a sprite is more than a specified y value.

Success Criteria:

- Make a sprite move to specific y axis coordinates

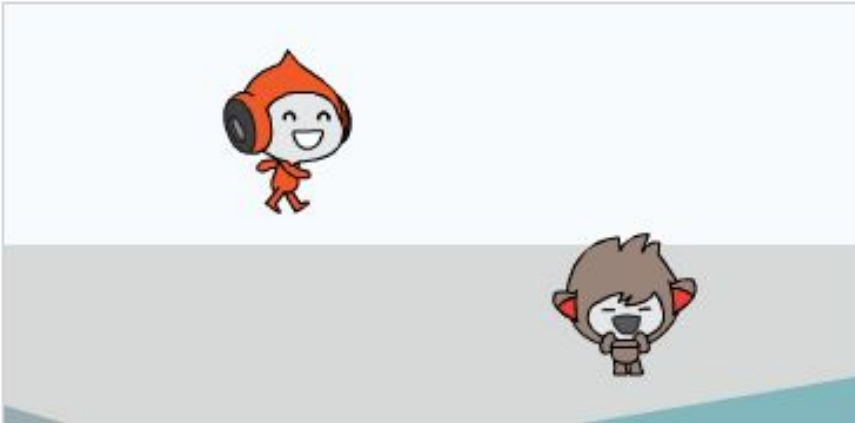
Investigation 2

Meeting conditions



ACTIVITY INSTRUCTIONS

- 1 Continue in your project **31-Multiple Sprites**, still working with Pico. We want to stop Pico from walking 'in the air', i.e. in the upper white part of the stage, using a similar approach to Activity 3.2.2.
- 2 Duplicate the whole **if touching color ... ? then ...** block and insert it in the existing **forever** walking script of Pico, replacing the dark grey colour by white colour of the sky in the stage (see additional support).
- 3 Run the script, observe Pico and think about his behaviour, noticing that he turns back as soon as he touches the sky colour by the top of his head instead of when his feet reach the same point.



While Pico walks forever around the stage, we do not want him to walk 'in the air' or in the sky.. Applying the strategy of checking whether he touches the colour of the sky does not work very well since Pico turns back as soon as he touches the sky by the tip of his cap, so it looks odd,

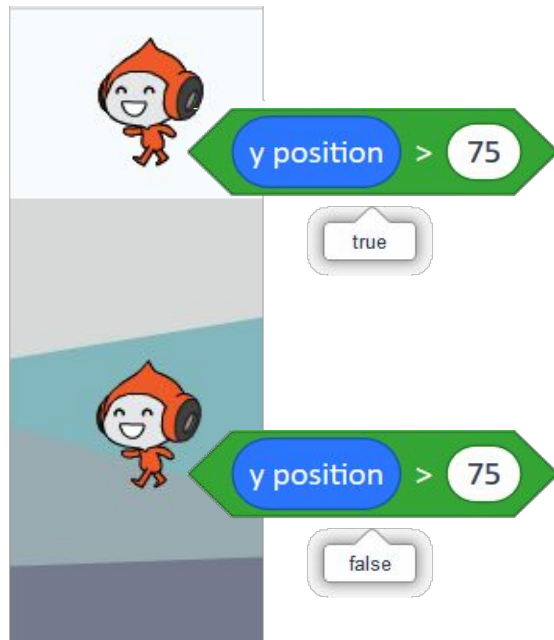
To modify this behaviour it is possible to check Pico's y position instead of checking the colour .

ACTIVITY INSTRUCTIONS

- Find out the highest y position Pico should walk to.. You do this by using the **y position** block from the Motion group, which is a reporter block.

NOTE: Always explore new blocks in isolation first. When you understand the **y position** reporter block, insert it into another new block

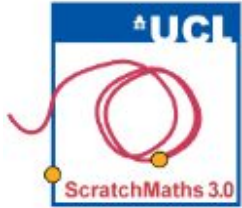
Also explore the compound block **... > ...** in isolation before using it inside **if ... then ...** block.



ACTIVITY INSTRUCTIONS

- 5 Drag the **y position** block into the scripts area and keep it isolated. Drag Pico vertically then click the **y position** block to find out his current y position. Repeat this to confirm his y position should never get bigger than 75.
- 6 To check whether Pico's y position is larger than 75, drag the **... > ...** condition block from the **Operators** group in the scripts area, keeping it isolated, insert the **y position** block in its left hole and type 75 in its right hole. Drag Pico up and down repeatedly, clicking this new condition **y position > 75** to find out whether it is *true* or *false*.
- 7 Replace the **touching color ... ?** block from the second **if ... then ...** block with your new condition to check Pico's **y position** and test it out (see additional support).





MODULE 3 • INVESTIGATION 2 • ACTIVITY 3.2.4



Unplugged: True or False?

Learning Intentions:

Envisage if a condition is true or false based on the current state of the sprites on the stage.
bridge to knowledge of coordinates.

Success Criteria:

- Identify whether conditions are true or false based on visuals. No technology!

Investigation 2

Meeting conditions



Use the picture to decide whether a condition is **True** or **False**.



Is Giga touching Pico?
True or **false**?
Explain why.

Solution!

Use the picture to decide whether a condition is **True** or **False**.

2



Is Pico touching Giga?
True or **false**?
Explain why.

Solution!

Use the picture to decide whether a condition is **True** or **False**.



Is Nano touching Pico?
True or **false**?
Explain why.

Solution!

Use the picture to decide whether a condition is **True** or **False**.



Is Tera touching the white colour?

True or **false**?

Explain why.

Solution!

Use the picture to decide whether a condition is **True** or **False**.

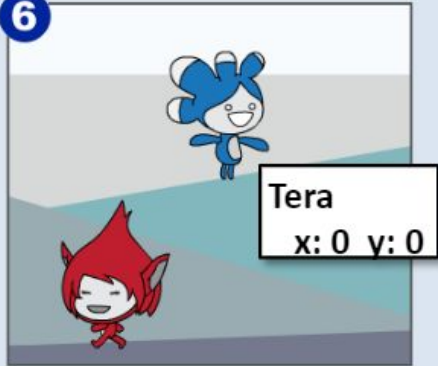


Tera stands in the centre.
Is Nano's **x position** smaller than 0?
True or **false**?
Explain why.

Solution!

Use the picture to decide whether a condition is **True** or **False**.

6

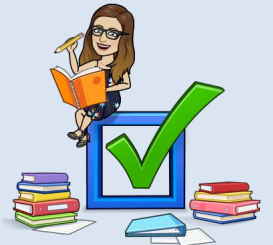


Tera stands in the centre.
Is Giga's **y position** bigger than 0?
True or **false**?
Explain why.

Solution!

My Module 3: Investigation 2 check list:

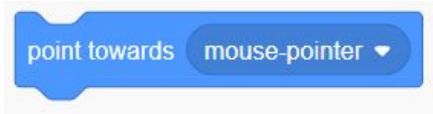
- I made Giga point towards a sprite before starting walking.
- I used the condition block to check if touching another sprite.
- I used the repeat until block with a condition to make Giga stop walking.
- I used the condition block to check if the sprite was touching a specific colour.
- I used the if block to make Giga turn back if she touches a selected colour.
- I made Pico turn back whenever his y position becomes bigger than certain value.
- I envisaged whether a condition block would say true or false using the current position of the sprites on the stage.



SCRATCH Vocab

Condition

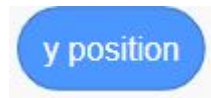
is a block reporting whether something is true or false



makes the sprite set its direction towards another sprite (selected from the drop down menu)



repeats the blocks inside it until its condition is true



reports the current y position of a sprite



reports true if the first value is greater than the second



reports true if the sprite is touching another sprite (selected from the drop down menu)



reports true if the sprite is touching a specified colour



if the condition is true, runs the blocks inside



End of Investigation 2



MODULE 3: INVESTIGATION 3

Broadcasting Messages

Learning Intention: Explore the concept of broadcasting messages between sprites, i.e., how to broadcast a message, how to receive it, and how to react to it. Build one or several scripts for different sprites to react to the same message in parallel.

Success Criteria:

- Explore how to make sprites interact.
- Explain the difference between a broadcast and a message.

This investigation introduces the concept of broadcasting as a means for sprites to communicate and collaborate. It also develops the concept of events and parallel reactions: whenever a message is broadcast, several sprites may react to it in parallel.

Investigation 3

Broadcasting Messages



**Unplugged:
Broadcast &
Receive**

**Introduction:
One to One**
Continue with
**31-Multiple
Sprites**
or start with
**33-Multiple
Sprites**

**Come to Tera:
One to Many**
Continue with
**31-Multiple
Sprites**

SCRATCH Vocab

broadcasting

is how Sprites communicate and collaborate. A sprite broadcasts a message and one or several sprites may react by running their scripts

event

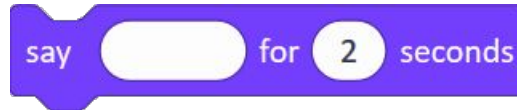
is a special situation like when I receive message or when green flag clicked etc. All scripts with that hat block will be run in parallel



this hat block will run the script attached to it whenever that message has been broadcast



this block is used to broadcast a message, which is text you type in



this block will say the text in it in a speech bubble for a specified number of seconds



this block will say the text in a speech bubble. To remove the speech bubble, click the red Stop sign or run a say block with different text



MODULE 3 • INVESTIGATION 3 • ACTIVITY 3.3.1

Unplugged: Broadcast & Receive



Learning Intentions:

Explore the concept of broadcasting and receiving messages.

Explain who reacts to a broadcast message and when.

Envisage how to broadcast messages between sprites in Scratch.

Success Criteria:

- Interact with others to model how sprites interact.

Investigation 3

Broadcasting Messages



ACTIVITY INSTRUCTIONS

This unplugged activity introduces the concept of broadcasting in a playful way and you need several people to play .

Print out copies of the cards from the additional support.

The **purple cards** should only be printed out once and each card should be given to **one child only** (i.e. 5 children should receive a purple card). You can print as many of the **orange cards** as you like

The **blue card** is for one of you **and should be read out to start the rhyme.**

STARTER CARD

Teacher reads the following:

The Grand old Duke of York he had ten thousand men

When I hear the line

The Grand old Duke of York he had ten thousand men

Stand up and say:

He marched them up to the top of the hill

And he marched them down again

When I hear the line

When I hear the line

When they were up

Stand up and say:

They were up

ACTIVITY INSTRUCTIONS

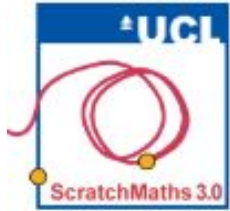
- 1 Think of yourselves as sprites and interpret the instructions on their cards as scripts.
- 2 You should only react when you hear the specific line (message) stated at the top of their card. In such case they follow the instructions on their card – to stand up, read aloud their line when they hear the previous line, then sit down.
- 3 Repeat this several times to ensure you all understand the process.
- 4 Discuss as a class what happened.

NOTE:

It is important to understanding the concept of broadcasting, that different messages can be sent to all the sprites but the sprites only react to a specific message otherwise the rhyme does not make sense.

DISCUSSION POINTS

- ◆ How did you know when to say your line (i.e. react to an event)? What was it important for you to do in order to do this? (*Listen*)
- ◆ Who could hear the lines of the poem (i.e. the messages)? (*Everyone that was listening*)
- ◆ What happened when multiple children had the same card (i.e. same script)? What do you think would happen in Scratch if multiple sprites had the same script?



MODULE 3 • INVESTIGATION 3 • ACTIVITY 3.3.2

Introductions: One to One



Learning Intentions:

Explore how sprites can collaborate by using broadcasts.

Explain how to broadcast and receive a message between two sprites.

Success Criteria:

- Code two sprites to interact by broadcasting and receiving a message.

Investigation 3

Broadcasting Messages



ACTIVITY INSTRUCTIONS

Continue in your project **31-Multiple Sprites** and select Nano. Build the following behaviour for him: when Nano is clicked, Tera will react by jumping high and floating slowly back.

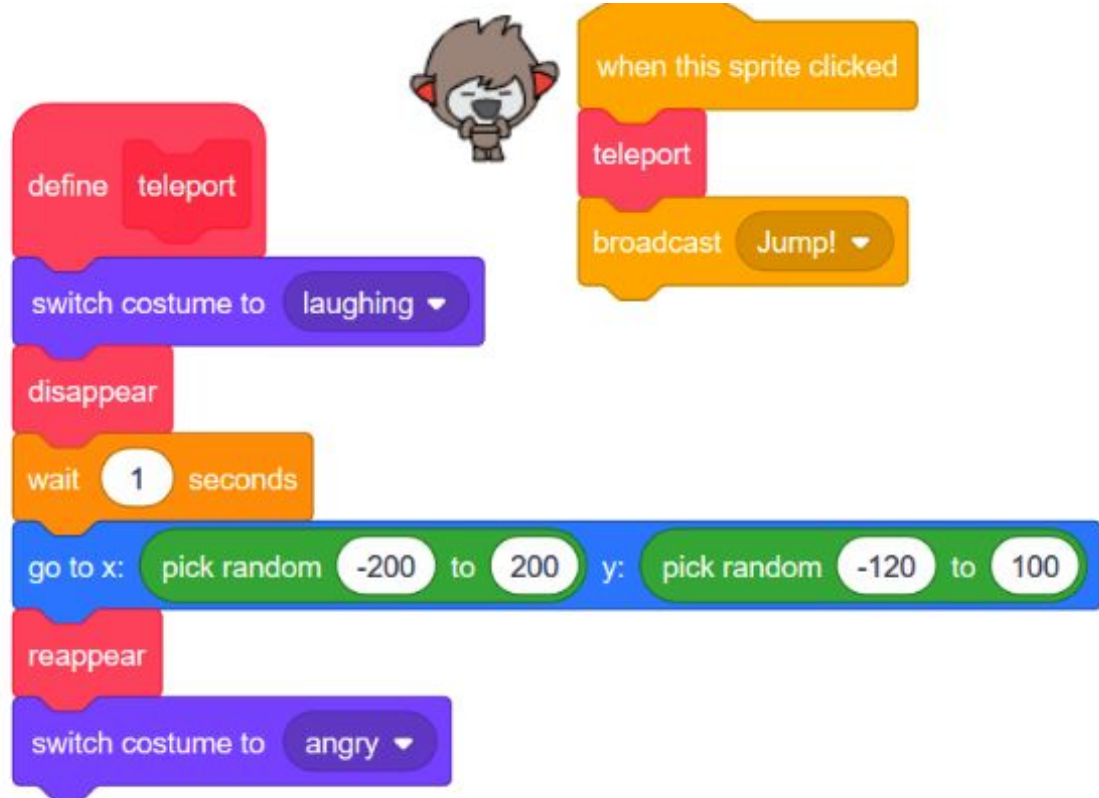
- 1 Drag a new block **broadcast message** from the **Events** group into the scripts area and keep it isolated from other scripts. In the drop down menu, choose **new message...** and type in **Jump!** – the message is to be broadcast. Click the block, making Nano broadcast his message – but nothing happens. Think about why this is.



- 2 Select Tera and make her “listen” for that message: from the **Events** group drag in the **when I receive Jump!** hat block and add Tera’s jumping reaction to it (see additional support).

ACTIVITY INSTRUCTIONS

- 3 Extend Nano's behaviour **when this sprite clicked**: he will first **teleport** himself and only then broadcast the *Jump!* message.



The image shows a Scratch script for a character named Nano. The script is triggered by the event "when this sprite clicked". The sequence of actions is as follows:

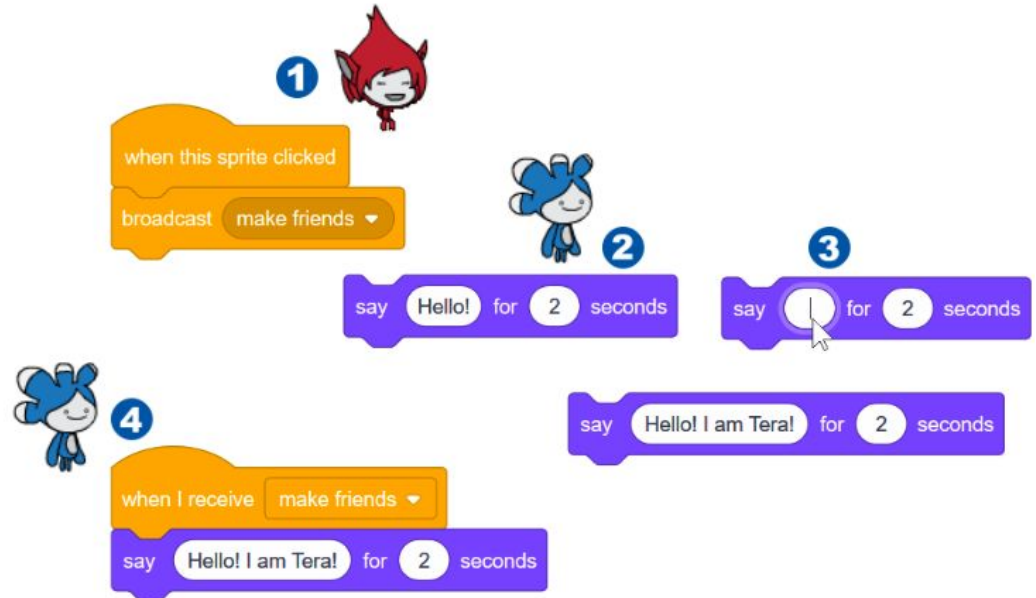
- Define a function named "teleport".
- Inside the "teleport" function:
 - Switch costume to "laughing".
 - Disappear.
 - Wait 1 seconds.
 - Go to x: pick random -200 to 200, y: pick random -120 to 100.
 - Reappear.
 - Switch costume to "angry".
- After the "teleport" function, broadcast the "Jump!" message.

The character Nano is shown in the top right corner of the script area.

ACTIVITY INSTRUCTIONS

Now select Giga. She wants to make friends with other sprites: when clicked, she will broadcast a message *make friends*. Currently only Tera will react.

- 4 Drag the **broadcast message** block from the Events group, change the *message* to *make friends*, and add the **when this sprite clicked** hat block.
- 5 Select Tera and make her “listen” for that message: from the Events group drag in the **when I receive *make friends*** hat block to build a script. Build a reaction e.g. the **say *Hello, I am Tera!* for 2 secs** block from the Looks group (see additional support).



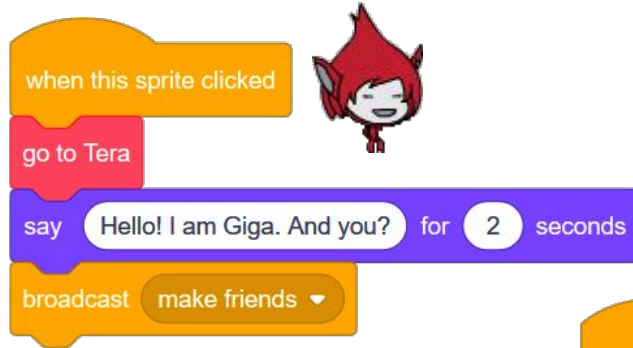
ACTIVITY INSTRUCTIONS

Extend to build **more complex behaviours** for Giga and Tera.

- 6 When Giga is clicked, she will walk over to Tera, *say Hello! I am Giga. And you? for 2 secs*, then broadcast her message.

Change Tera's reaction so she will also jump high and slowly float back, then say *Hello! I am Tera!*

- 7 [Extension] If pupils completed the extension from activity 3.2.1 then Giga will also walk to Nano. Repeat the same process and introduce Giga and Nano as well.





MODULE 3 • INVESTIGATION 3 • ACTIVITY 3.3.3

Come to Tera: One to Many



Learning Intentions:

Explore how to use broadcasts to initiate actions in multiple sprites at the same time.

Explain the difference between saying a message and broadcasting a message.

Success Criteria:

- Broadcast messages to multiple sprites.

Investigation 3

Broadcasting Messages



ACTIVITY INSTRUCTIONS

- 1 Continue in your **31-Multiple Sprites** project and select Tera.

Extend Tera's reaction: when she is clicked, Tera will jump high and float back, then *say Come to me, my friends! for 2 secs*, then broadcast the invitation message *come to Tera*, thus inviting **whoever is listening to her message**.

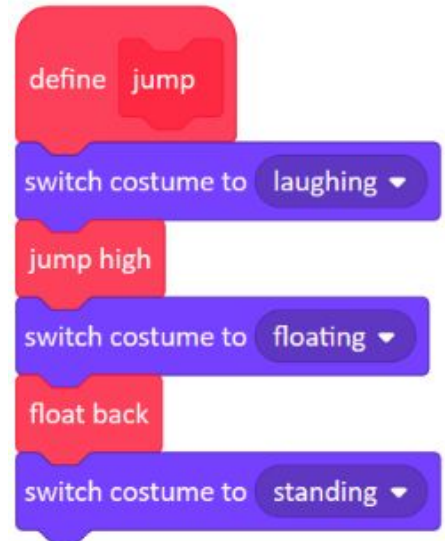
Note that Tera already jumps and floats back in more than one situation: when reacting to Giga's message *make friends* and also when she is clicked. This requires her *jumping and floating* reaction to be defined as a new block to avoid repeating the same script.

- 2 Build new block **jump** to combine Tera's jumping/floating behaviours.

1



2



ACTIVITY INSTRUCTIONS

- 3 Add the **say *Come to me, my friends!* for 2 secs** block to the **when this sprite clicked** script of Tera, and also the **broadcast *Come to Tera*** block.

Note: the difference between **say** and **broadcast** blocks (the **say** block has only visual effect on the stage: other sprites will not “see” that Tera said anything. The only way to let them know is to **broadcast** a message).

- 4 Select Pico and build his **when I receive *Come to Tera*** reaction: he will come to Tera and stay there, applying a similar strategy, with the **point towards *Tera*** and **repeat until touching *Tera*** blocks, as we did in Activity 3.2.1.



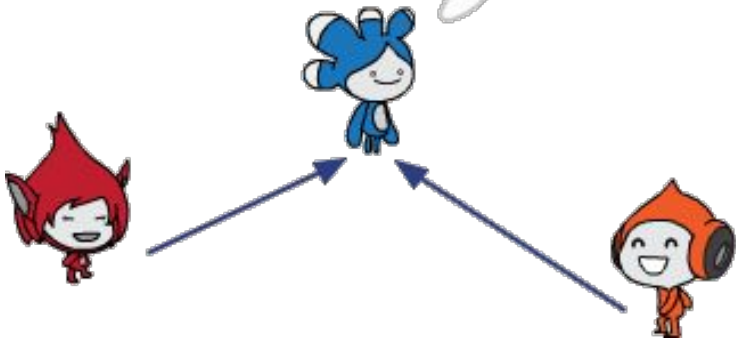
ACTIVITY INSTRUCTIONS

- 5 Build or duplicate identical *when I receive Come to Tera* reaction for Giga and test all the scripts by clicking on Tera.

Giga will have exactly the same reaction. Note that we have already built the **go to Tera** script for her in the previous Activity.



Come to me, my friends!



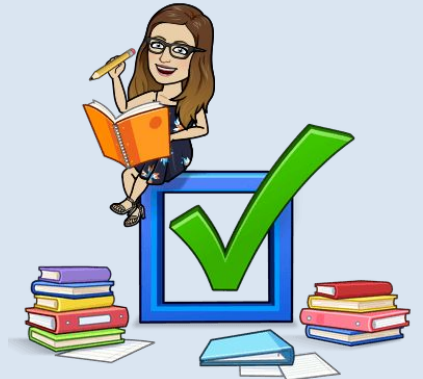
Nano will react by disappearing, jumping to the actual position of Tera then reappearing again. In the second alternative solution (on the right) Nano jumps to Tera but slightly changes his y and x positions before reappearing back.



The image shows two alternative Scratch scripts for Nano's reaction to the 'Come to Tera' event. The first script (left) has the following blocks: 'when I receive come to Tera', 'switch costume to laughing', 'disappear', 'wait 1 seconds', 'go to Tera', 'reappear', and 'switch costume to angry'. The second script (right) has the following blocks: 'when I receive come to Tera', 'switch costume to laughing', 'disappear', 'wait 1 seconds', 'go to Tera', 'change y by pick random -40 to -20', 'change x by pick random -30 to 30', 'reappear', and 'switch costume to angry'.

My Module 3: **Investigation 3** check list:

- I made Nano broadcast a message. I made Tera react to that message by jumping high and floating back.
- I made Nano first teleport then broadcast a message.
- I made Giga broadcast a message and I made Tera react to that message.
- I made a sprite say something e.g. a question, an invitation or a greeting.
- I made Tera broadcast a message. I made two sprites, Giga and Pico react to it in parallel by walking to Tera.
- I made Nano react to the same message by teleporting to Tera.





End of Investigation 3



MODULE 3: INVESTIGATION 4

Interactive Stories

Learning Intentions:

bridge to knowledge of coordinates in all four quadrants, factors as well as positive and negative numbers.

Envisage the outcome of different scripts, behaviours, reactions and interactions between sprites.

Explain why a script would implement an expected behaviour and how to modify it to achieve an expected interactions and outcomes.

Success Criteria:

- Broadcast messages to multiple sprites.

Investigation 4
Interactive Stories

- 1 Circle the script that will make Nano disappear, teleport to a random position and then reappear after 1 second.



when this sprite clicked

hide

go to x: 50 y: -60

wait 1 seconds

show

when this sprite clicked

hide

jump to random position

show

wait 1 seconds

when this sprite clicked

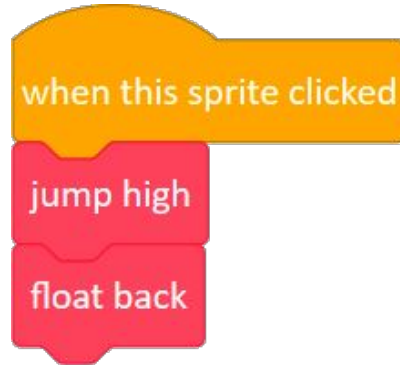
hide

jump to random position

wait 1 seconds

show

- 2 What number should go in the **repeat** block so that Tera floats back to the same position when she is clicked? Explain why.



Repeat number =

Explain why =

- 3 Will Tera move up or down when the script on the right is clicked? Explain your thinking.



when this sprite clicked

change y by pick random -50 to -10

Tera will move _____ because:

- 4 Circle the purple block that should be added in the script below to make Pico walk like the picture below.



when this sprite clicked

forever

- move 3 steps
- wait 0.2 seconds
- if on edge, bounce



change size by 10

next costume

show

switch costume to walking 2

- 5 Giga is walking towards Nano (see picture). For each of the blocks below fill in the value that will be shown when it is clicked.



touching Nano ▾ ?



?

touching color ● ?



?

touching color ● ?



?

- 6 Next to each of the blocks fill in the direction that Tera would move when that block is clicked.



Will Tera move **Up**, **Down**, **Left** or **Right**?

change y by 50

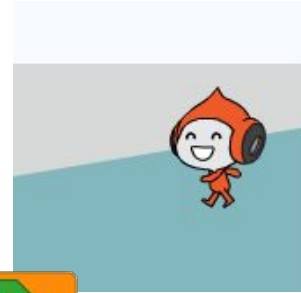
change x by 70

change x by -20

change y by -80

ASSESSMENT TASKS

- 7 Pico is near the right edge of the stage. Mark the script that will make him walk and stop when he reaches the middle of the stage.



```
when this sprite clicked
repeat 20
  next costume
  move 2 steps
  if on edge, bounce
  wait 0.1 seconds
```

```
when this sprite clicked
repeat until x position < 0
  next costume
  move 2 steps
  if on edge, bounce
  wait 0.1 seconds
```

```
when this sprite clicked
forever
  next costume
  move 2 steps
  if on edge, bounce
  wait 0.1 seconds
```

ASSESSMENT TASKS

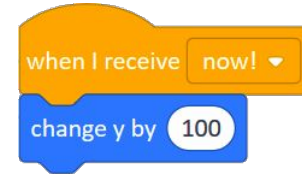
- 8 Look at Pico's script below. What will Nano and Tera do after Pico is clicked? Explain what will happen on the screen in the box below.

When Pico is clicked...

Pico



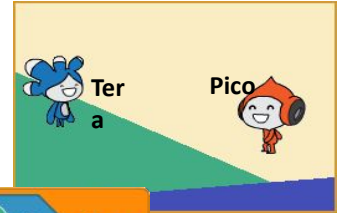
Tera



Nano



9 Look at the current position of Pico on the stage. The scripts on the right are in Pico's scripts area – explain what will happen to Pico if each of these scripts is clicked.



```

hide
if touching color [blue] ? then
  go to Tera
  
```

When this script is clicked... what will happen?

```

if touching color [yellow] ? then
  go to Tera
  hide
  
```

When this script is clicked... what will happen?

```

if touching color [green] ? then
  go to Tera
  say Hello! for 2 seconds
  
```

When this script is clicked... what will happen?

[Extension]

10 When Giga is clicked what will Nano do? Write your answer in the box on the right.

Giga



when this sprite clicked

say Hello! for 1 seconds

broadcast Bye!

Nano



when I receive Hello!

go to x: -100 y: 0

when I receive Bye!

hide

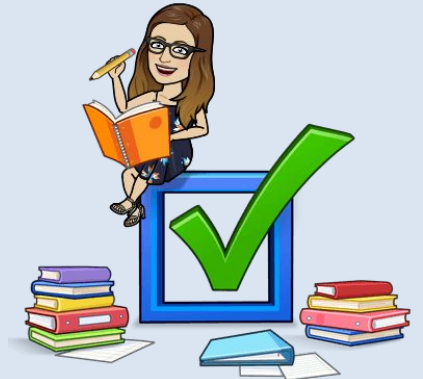
wait 1 seconds

show

When Giga is clicked Nano will...

My Module 3: **Investigation 4** check list:

- My Investigation 4 check list:
- I built an interactive story between two or more sprites.
- I prepared a plan for my story.
- I built several Scratch scripts to implement new behaviours.
- I built several interactions between our characters, using the broadcast message and when I receive message blocks.
- I used a script from another in our story project.
- We modified a script shared by another within our story project.



SCRATCH Vocab

broadcasting

is how Sprites communicate and collaborate. A sprite broadcasts a message and one or several sprites may react by running their scripts

event

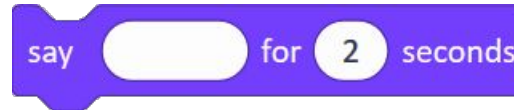
is a special situation like when I receive message or when green flag clicked etc. All scripts with that hat block will be run in parallel



this hat block will run the script attached to it whenever that message has been broadcast



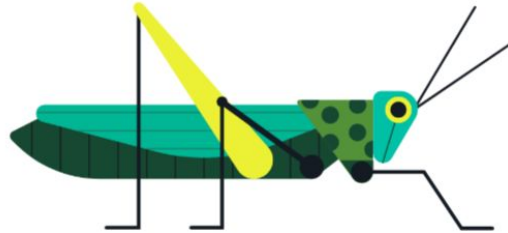
this block is used to broadcast a message, which is text you type in



this block will say the text in it in a speech bubble for a specified number of seconds



this block will say the text in a speech bubble. To remove the speech bubble, click the red Stop sign or run a say block with different text



End of Investigation 4