

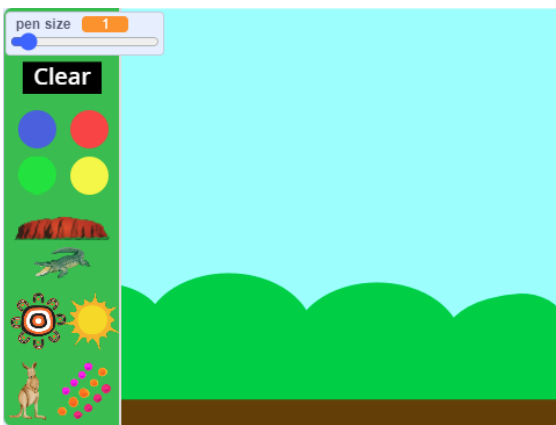
CODE OF ORIGIN



NT Create a scene

Create an art pad where the user can create their own scene that shows the Northern Territory.

INTRODUCTION



What you will need

HARDWARE

A computer capable of running Scratch 3

SOFTWARE

Scratch 3:
either online
[rpf.io/scratchon](https://scratch.mit.edu/projects/873312914/)
or offline
[rpf.io/scratchoff](https://scratch.mit.edu/projects/873312914/)

What you will learn

- How to use **clones**
- How to use **branching** to determine next actions
- How to use the **pen extension tool**

Starter Project

<https://scratch.mit.edu/projects/873313054/>

Additional notes for educators

Here is a link to the completed project
<https://scratch.mit.edu/projects/873312914/>

Read this [blog post](#) for information about the Code of Origin project by Code Club Australia.

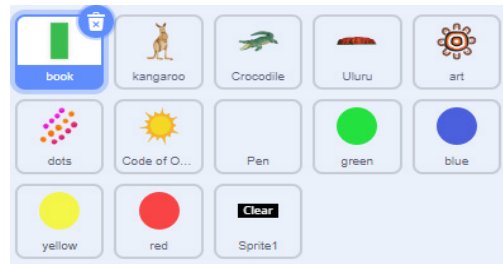


STEP 1 - SETTING THE BOOK

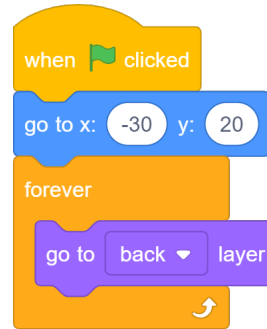
Open the starter project -<https://scratch.mit.edu/projects/873313054/>



Select the book sprite.



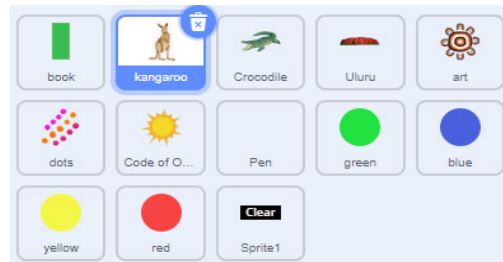
The book will help to control the sprites. Start with the event **when green flag clicked**. Set the **coordinates** to lock it into place. Add a **forever** block so it always sits at the **back**.



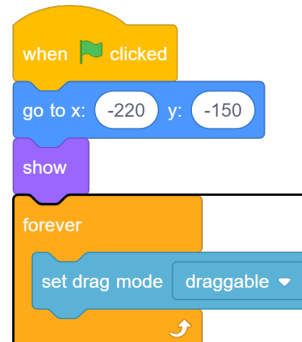
STEP 2 - CREATING THE STICKERS



Select the kangaroo sprite.



Set the conditions for the sprite **when the green flag is clicked**. Add a block for the sprites starting **location** and to **show**. Add a **forever** control and **set the drag mode to draggable**. This means that the user will be able to move the sprite on the screen.

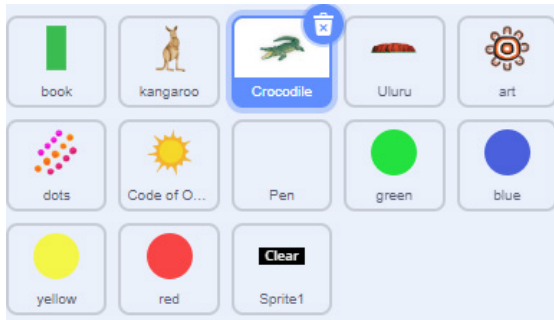


This algorithm needs to be copied to the other 'sticker' sprites - crocodile, Uluru, art, dots and sun. Drag the algorithm from the kangaroo sprite and 'drop' it onto the other sprites. For each sprite change the starting location as shown on the next page.





Crocodile sprite.

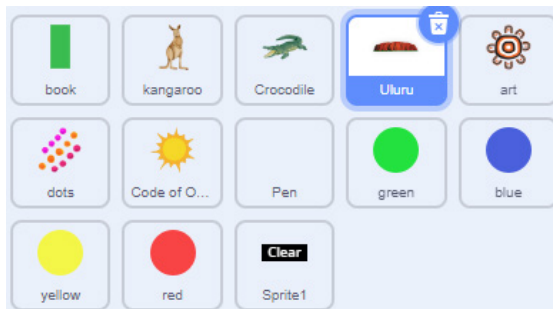


```

when green flag clicked
  go to x: -190 y: -40
  show
  forever loop
    set drag mode to draggable
  
```



Uluru sprite

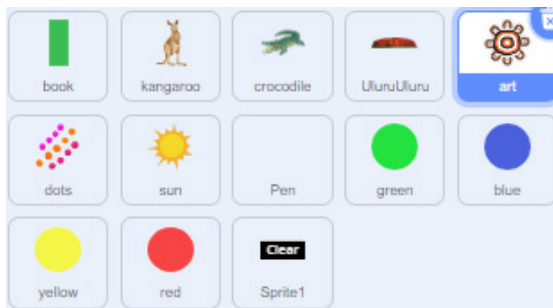


```

when green flag clicked
  go to x: -190 y: -10
  show
  forever loop
    set drag mode to draggable
  
```



Art sprite

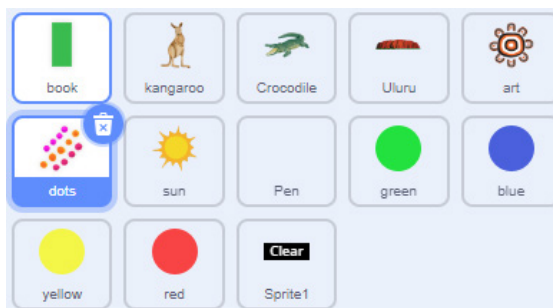


```

when green flag clicked
  go to x: -210 y: -90
  show
  forever loop
    set drag mode to draggable
  
```



Dots sprite



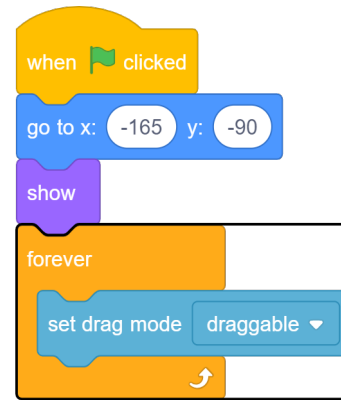
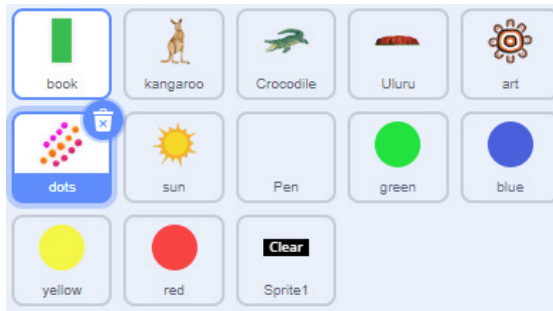
```

when green flag clicked
  go to x: -165 y: -90
  show
  forever loop
    set drag mode to draggable
  
```





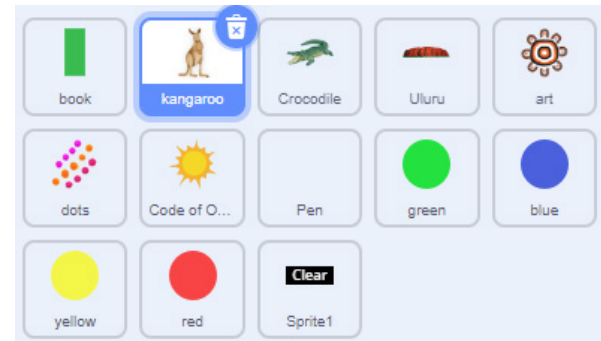
Sun sprite



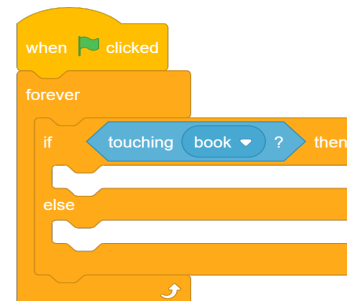
The next algorithm that we create will set conditions so that when the sprite is touching the book it is small, but when it is dragged onto the page it gets larger. The algorithm will be the same for all of the sprites, except for the location information. You can drag the completed algorithm to all the other sprites to copy it across.



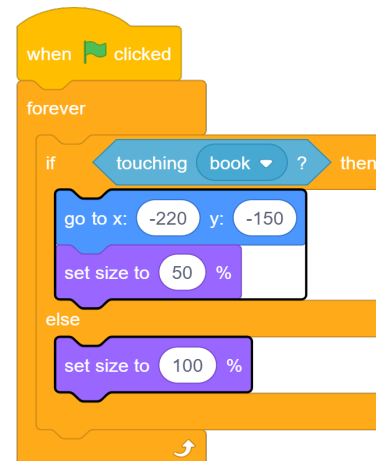
Click on the kangaroo sprite.



Start with the **green flag clicked** block. Add a **forever** control and inside an **if-then-else** block, that contains the sensing block of **touching book**.

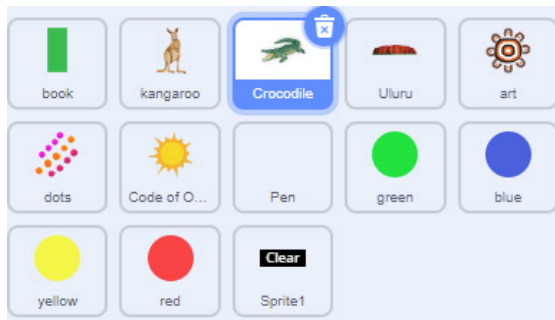


If the sprite is touching the book sprite we need to send it to it's **location** and **shrink it to 50%**. If it is not touching the book we want to **set the size to 100%**.





Crocodile sprite.

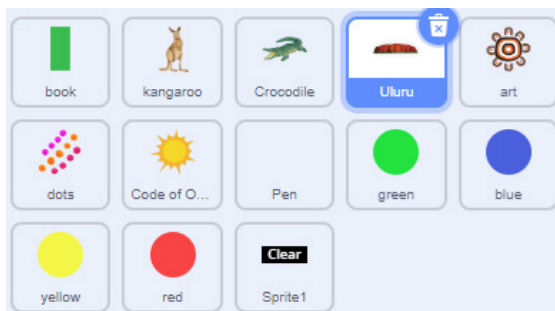


```

when green flag clicked
  forever loop
    if touching book then
      go to x: -190 y: -40
      set size to 50 %
    else
      set size to 100 %
  
```



Uluru sprite

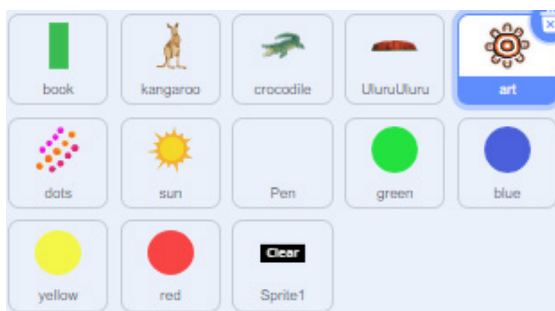


```

when green flag clicked
  forever loop
    if touching book then
      go to x: -190 y: -10
      set size to 50 %
    else
      set size to 100 %
  
```



Art sprite

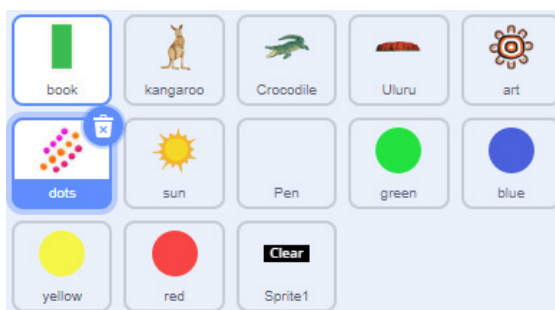


```

when green flag clicked
  forever loop
    if touching book then
      go to x: -210 y: -90
      set size to 50 %
    else
      set size to 100 %
  
```



Dots sprite



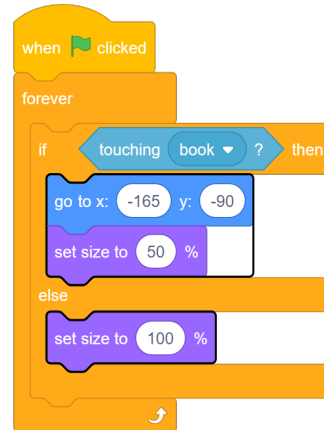
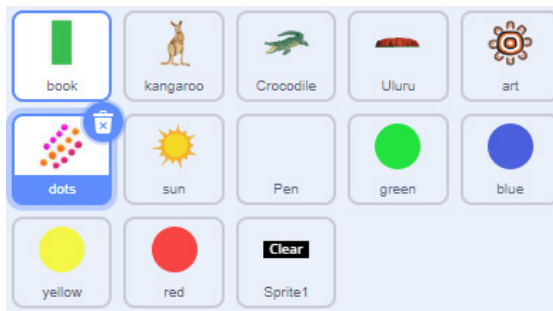
```

when green flag clicked
  forever loop
    if touching book then
      go to x: -170 y: -150
      set size to 50 %
    else
      set size to 100 %
  
```





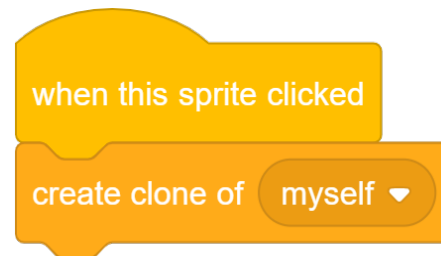
Sun sprite



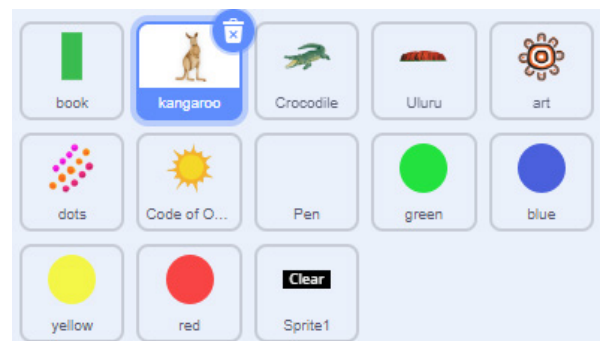
On **each of the sprites** add this code. Make sure you put it on the kangaroo, crocodile, Uluru, art, dots and sun.



Add an event **when this sprite clicked**.
Add a block to **create a clone of itself**.
For the user this means that when they click the sprite it will create a sticker they can drag onto the page.
Remember to put it on each sticker sprite.



Click on the kangaroo sprite.

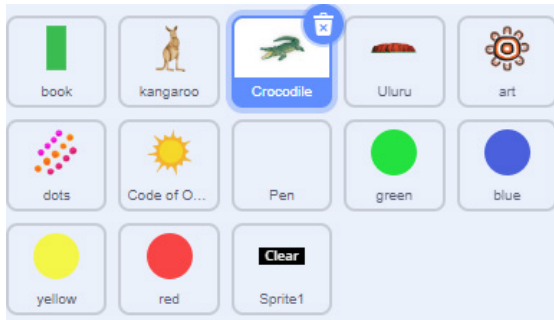


This next block of code is created by duplicating the green flag algorithm that is already on the page, and changing the event to **when I start as a clone**.





Crocodile sprite.

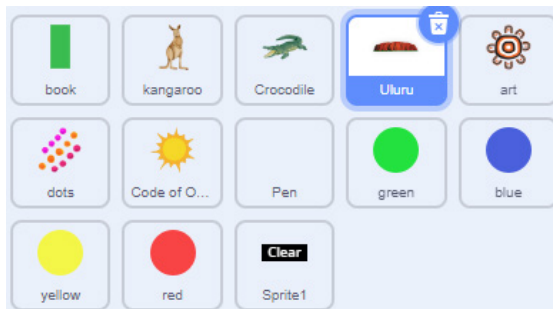


```

when I start as a clone
  forever
    if touching book ? then
      go to x: -190 y: -40
      set size to 50 %
    else
      set size to 100 %
  
```



Uluru sprite

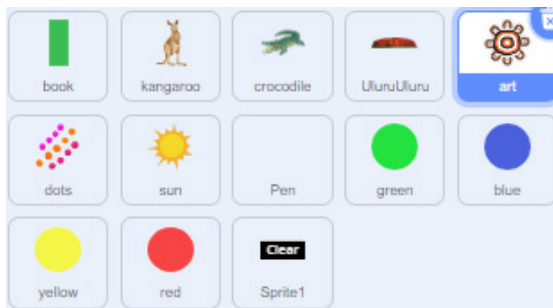


```

when I start as a clone
  forever
    if touching book ? then
      go to x: -190 y: -10
      set size to 50 %
    else
      set size to 100 %
  
```



Art sprite

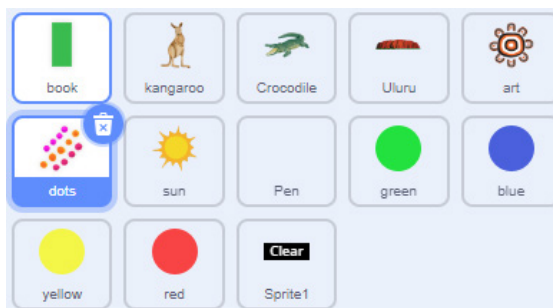


```

when I start as a clone
  forever
    if touching book ? then
      go to x: -210 y: -90
      set size to 50 %
    else
      set size to 100 %
  
```



Dots sprite



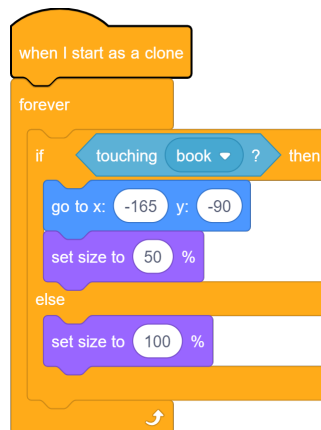
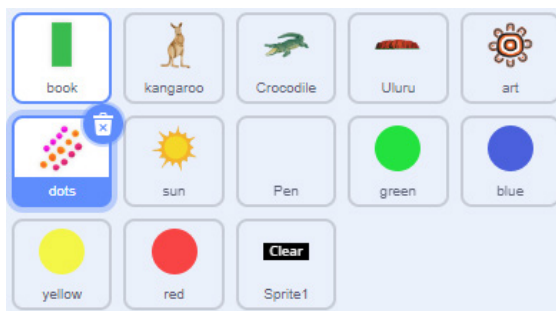
```

when I start as a clone
  forever
    if touching book ? then
      go to x: -170 y: -150
      set size to 50 %
    else
      set size to 100 %
  
```





Sun sprite

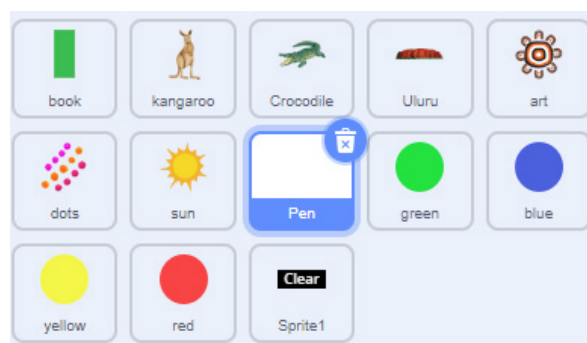


Test your code. Click a sprite to make a clone and then drag it onto the page. It should get larger. Drag it back to the book and it should go back to its starting spot.

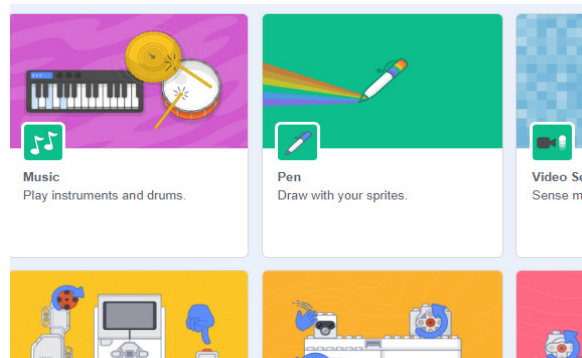
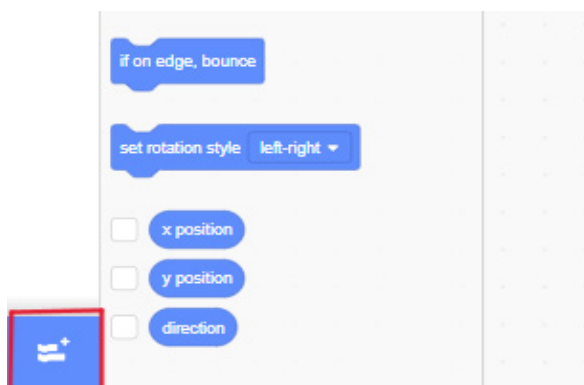
STEP 3 - DRAWING TOOLS



Click on the pen sprite.

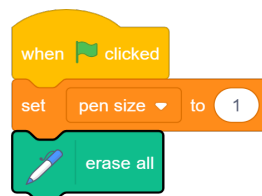


Add in the pen extension blocks. Click on the Add Extension button in the bottom left of your screen. When the selection comes up, choose the Pen tools.

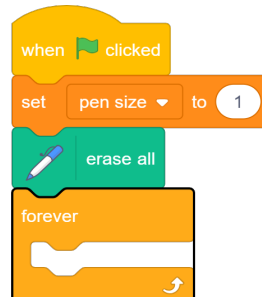




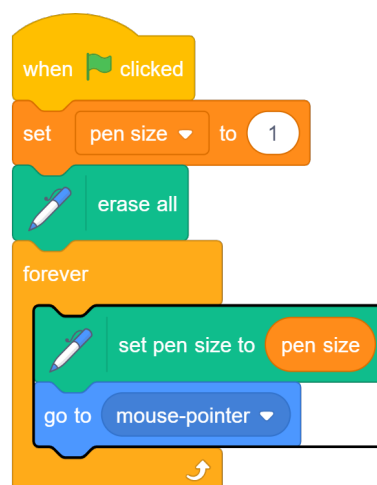
Start with the event **when green flag is clicked**. Make a new variable called **pen size** and set it to **1**. Add a block from the pen menu, **erase all**.



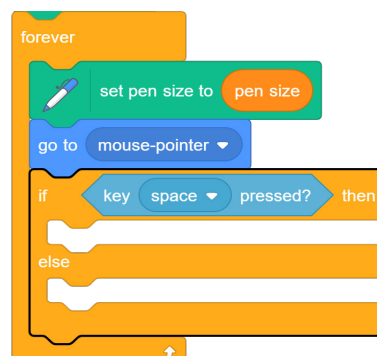
Add a **forever** control block.



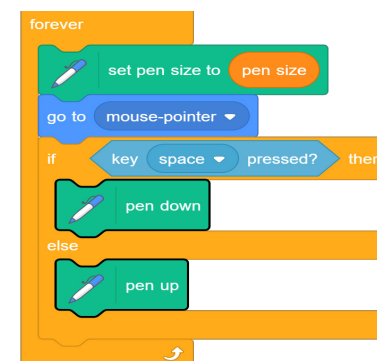
In the forever block add a block to **set the pen size**. From the variable menu add in the **pen size** bubble. Then add a motion block to **go to mouse pointer**. This means the pen will be wherever the user moves their mouse.



Add in an **if-then-else** block. Make sure you are still in the forever block. Add in the sensing block of **space key pressed**. This will control the pen.



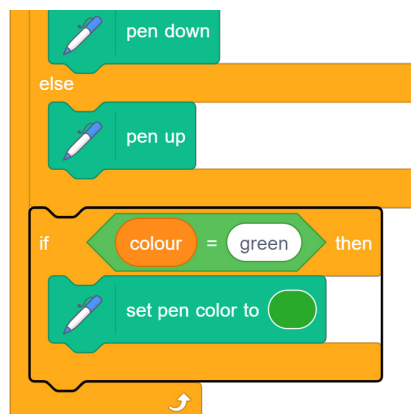
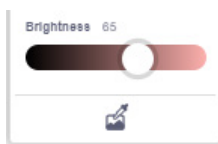
After then add in a **pen down** block. After else add in a **pen up** block. If the space key is pressed it will draw, and if it isn't it won't draw.



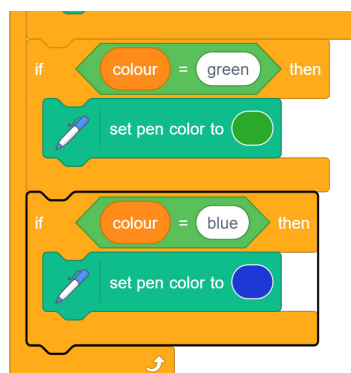


Underneath add an **if-then** block. Make a new variable called colour. Add an **=** **operator** to create the condition if **colour** (variable) = green then **set the colour of the pen** to green.

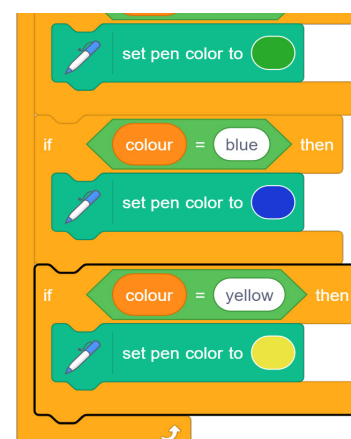
Note - the colour needs to be exact to work. Use the eyedropper on the green sprite to match the colour.



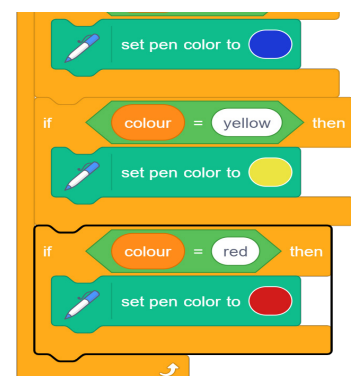
Duplicate the if-then block for the colour blue. Make sure it goes under the last if-then block.



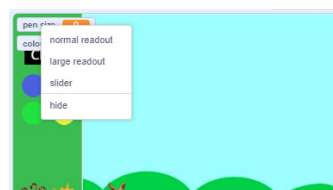
Duplicate it for yellow. Make sure it goes under the last if-then block.



Duplicate it for red. Make sure it goes under the last if-then block.



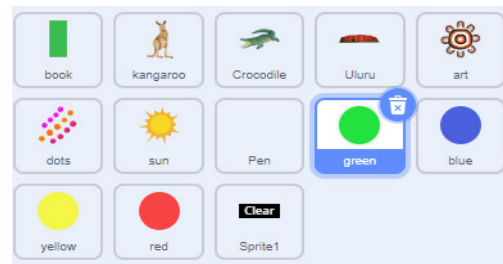
On the project page, right click the pen size variable and change it to a slider. Hide the colour variable.



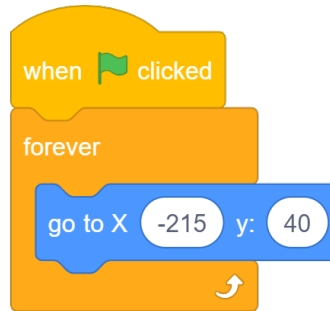
STEP 4 - THE COLOURS



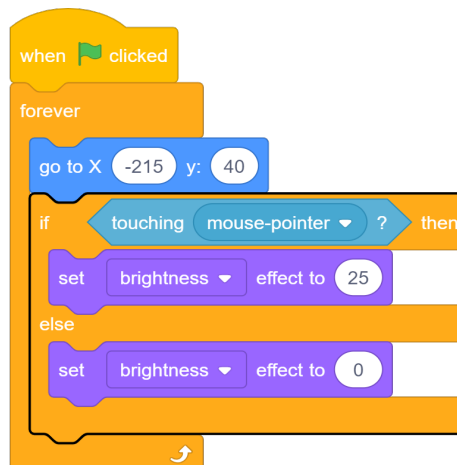
Click on the green sprite



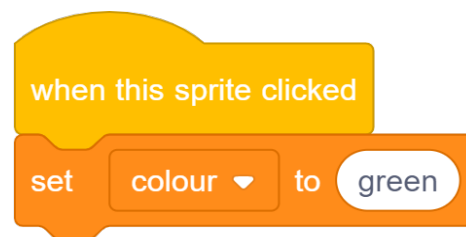
Create an algorithm that will be followed **when the green flag is clicked**. Add a **forever** block and its **location**.



Add in an **if-then-else** block with the sensing block of **touching mouse point**. Add a **set brightness effect to 25** after then, and a **set brightness effect to 0** after the else. These blocks will create an effect that when the user hovers over the button it will change colour, giving the illusion of the button being pressed.



Start a new algorithm **when this sprite is clicked**. From the variables menu **set colour to green**.



These algorithms will be copied to each of the coloured sprites (blue, yellow and red), change the location and the colour variable.





Drag and drop the code from the green sprite onto the blue sprite and make the necessary changes.

```

when this sprite clicked
  set colour to blue
  
```

```

when green flag clicked
  forever loop
    go to X: -215 y: 80
    if touching mouse-pointer then
      set brightness effect to 25
    else
      set brightness effect to 0
  
```



Drag and drop the code from the green sprite onto the yellow sprite and make the necessary changes.

```

when this sprite clicked
  set colour to yellow
  
```

```

when green flag clicked
  forever loop
    go to X: -170 y: 40
    if touching mouse-pointer then
      set brightness effect to 25
    else
      set brightness effect to 0
  
```



Drag and drop the code from the green sprite onto the red sprite and make the necessary changes.

```

when this sprite clicked
  set colour to red
  
```

```

when green flag clicked
  forever loop
    go to X: -170 y: 80
    if touching mouse-pointer then
      set brightness effect to 25
    else
      set brightness effect to 0
  
```



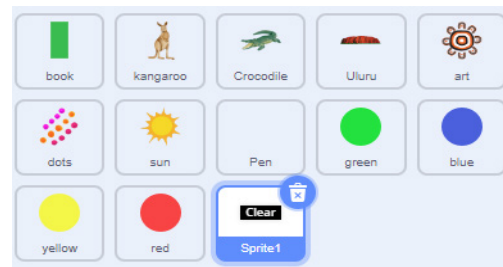
Test your code. Remember that you have coded the colour sprites to respond to clicking on the colour and then holding the space key down to make it draw.



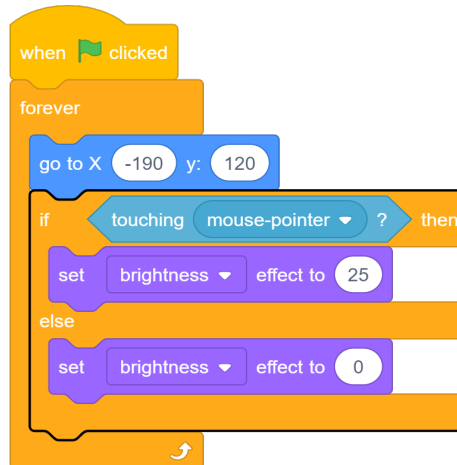
STEP 5 - THE CLEAR BUTTON



Click on the clear sprite



This sprite will clear all of the drawing that the user has done. Copy the code from the colour sprites that starts with the green flag.



Add a second algorithm that start with the event when the sprite clicked. Add an erase all button from the pen menu.



Your program is now complete. Test out all features to see that it is working correctly.



Challenges:

Improved user experience

There are some things that are a little glitchy in how this program works. What could you do to improve the code to make it better for the user?

Larger screen

Can you add code to the book so that it can be open and closed? The user can open it to access the stickers and pen, and then close it to have more background to work on.

Background

At the moment there is only one background for the user. Can you change it so that they can select from multiple backgrounds?

Congratulations!

You have created a game that will help others learn about the Northern Territory.

Which state or territory will win the Code of Origin?

