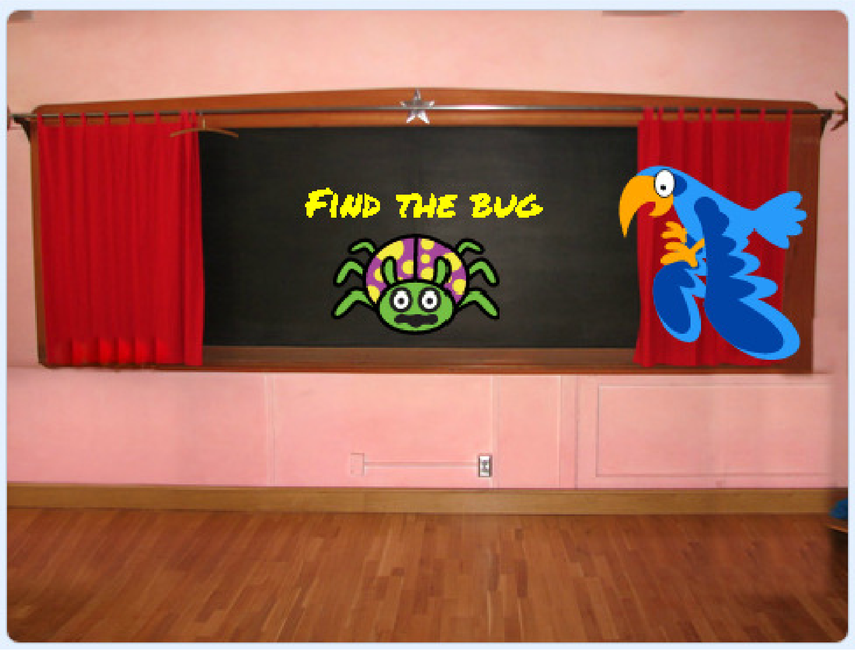
Find the Bug!

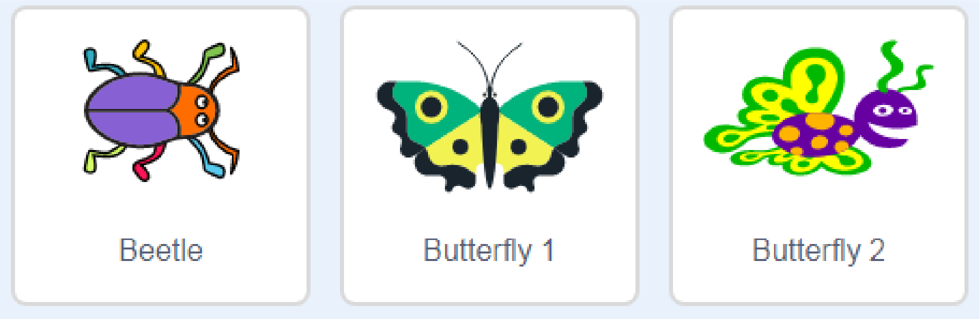
These instructions are a shorter version of https://projects.raspberrypi.org/en/projects/find-the-bug/.



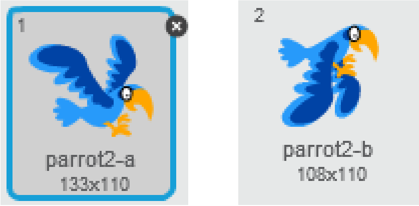
In this project, we will create a game to find a hidden bug. ‘Bug’ also means a mistake in your code, so it has two meanings here!

Step 1: Choose your sprites

- We won’t need Scratch cat, so that sprite can be deleted.
- There are several bugs in Scratch to choose from, or if you have time, you could draw your own:

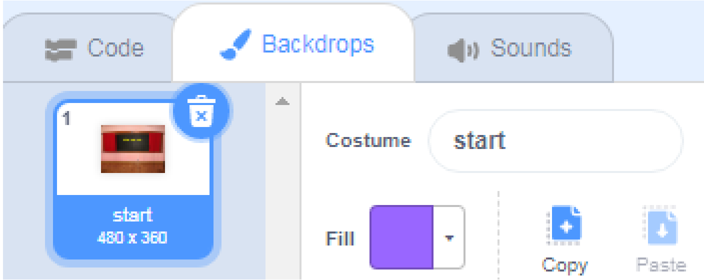


- We also need another sprite, to be an annoying distraction. I chose a parrot, but any flying sprite with two costumes will do:



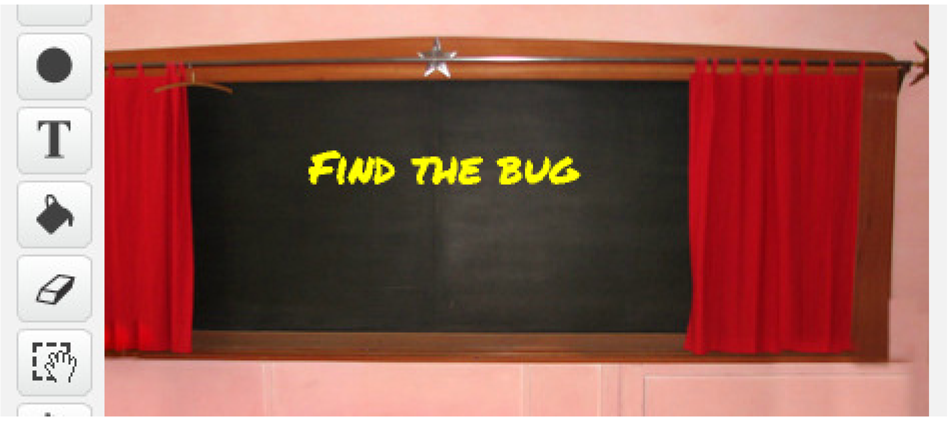
Step 2: Choose your backdrops

- We can add lots of backdrops to this game, to make it more challenging, but we will start with just two.
- First, choose the ‘chalkboard’ backdrop, as shown above. We will use this as our start and end screens, so we need to rename this first one to ‘start’:



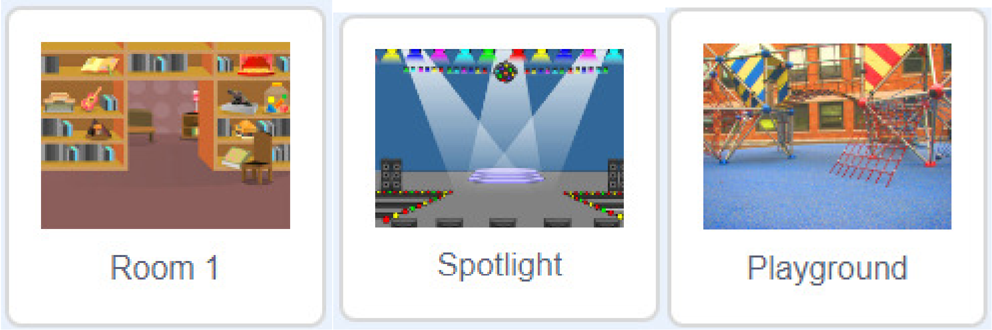
Costume: start

- Select the Text tool T on this backdrop and choose the font and colour you will use to add the text ‘Find the Bug’:



- Don’t forget the ‘undo’ button if you want to try again!

- Choose a second backdrop with lots of detail, so the bug can find a place to hide! For example, any of these will be good:

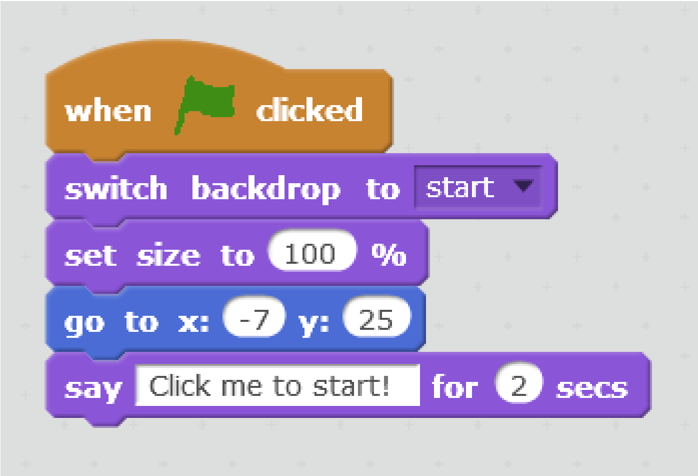


- There is no need to rename this backdrop.

Save your project

Step 3: Now some code!

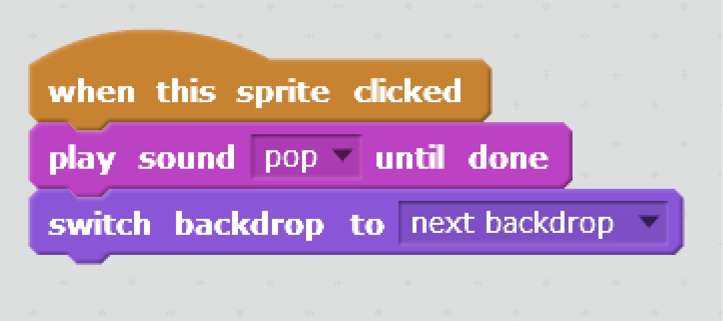
- Drag your bug sprite onto the chalkboard, where you want it to be when the game starts. Then add this code to it:



```
when ⚑ clicked
switch backdrop to start ▾
set size to 100 %
go to x: -7 y: 25
say Click me to start! for 2 secs
```

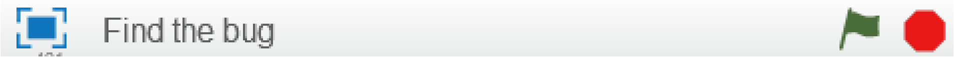
- This will make sure your game starts on the ‘start’ backdrop, the bug is full-size and in the right position. (You don’t need to change the numbers in the ‘go to x: y:’ block if you have already put your bug where you want it.)
- And… the bug tells us how to start the game!
- Test your code.

- This is a good start, but we want something to happen when we click the bug! Add this code below to the bug sprite:



```
when this sprite clicked
play sound pop until done
switch backdrop to next backdrop
```

- You can search the sound library for a different sound instead of 'pop' – it will work best if you choose a short one.
- Test your code. You should see the backdrop change when you click on your bug, and see it go back to the chalkboard when you click the green flag. If your code sometimes doesn't work, test it on the fullscreen setting by clicking this blue icon:



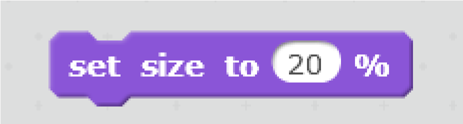
- It is easy to slightly drag a sprite on the small screen, so the code doesn't run, but this won't happen on the fullscreen setting.

Save your project

Step 4: Hide the bug

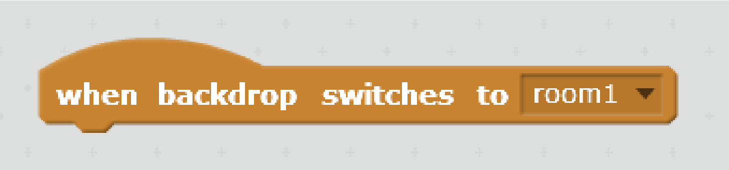
- Our backdrops should be working, but the bug needs to shrink, and find a place to hide!

- Make sure your project is stopped (by clicking the red button ●), and is showing the second backdrop – the one with lots of detail.
- Add this code block (all on its own), and click it to shrink the bug by an exact amount:

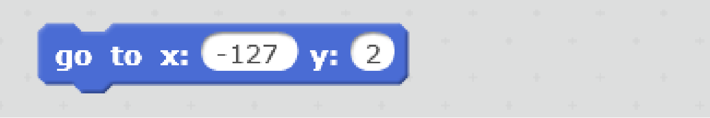


```
set size to 20 %
```

- Drag the tiny bug sprite to a hiding place on the backdrop, so it is hard to see.
- Then add the following code blocks, and join them together with the 'set size' block in the middle:

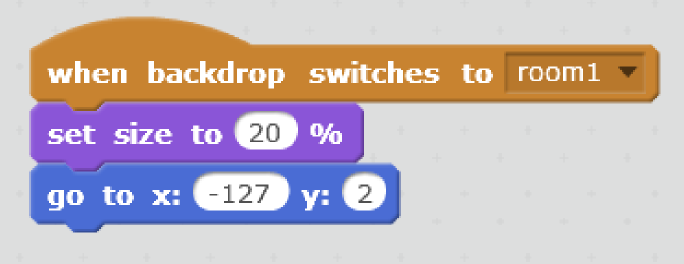


```
when backdrop switches to room1
```



```
go to x: -127 y: 2
```

- So you should have:



```
when backdrop switches to room1
set size to 20 %
go to x: -127 y: 2
```

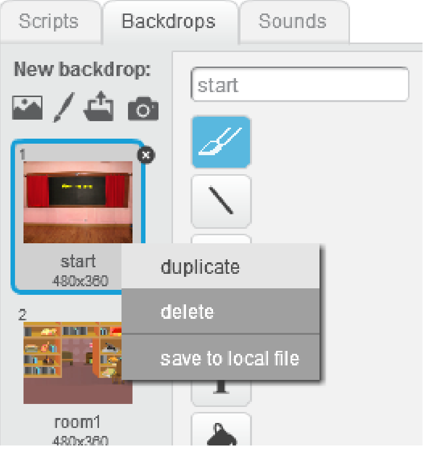
- Remember, you may have used a different backdrop, and the numbers in your 'go to x: y:' will probably be different – you don't need to copy mine!

Save your project

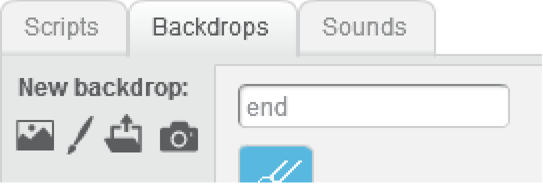
Step 5: Finishing off

- Test your code. Does everything look ok?
- You have probably found that your bug stays small when it gets back to the start screen. Let's add an 'end' screen to finish the project nicely.
- Select your backdrop, and go to the Backdrops tab (same as Costumes, for sprites)

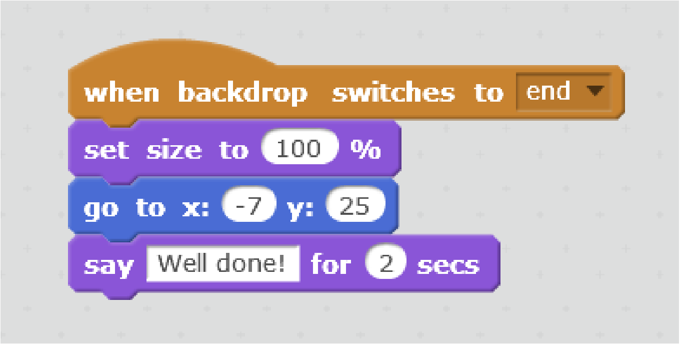
- Right-click on the backdrop we called ‘start’ and choose ‘duplicate’, to copy it:



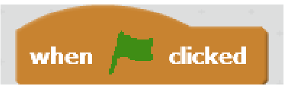
- In the same way we renamed the first one, rename this copy of the chalkboard backdrop to ‘end’:



- Now go back to your bug sprite, and add this code:



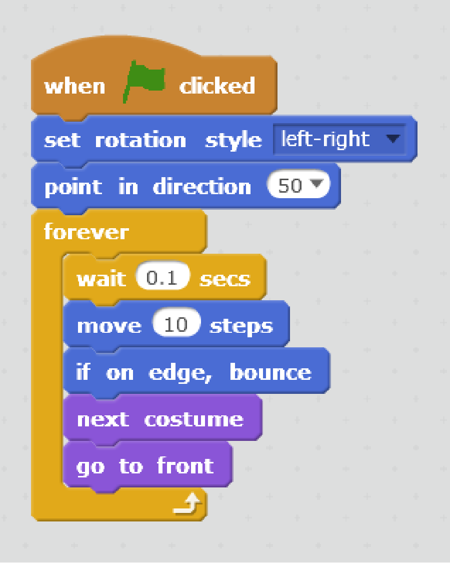
```
when backdrop switches to end
set size to 100 %
go to x: -7 y: 25
say Well done! for 2 secs
```

- Hint, apart from the hat block, you may be able to copy this from your  code!

Save your project

Step 6: Annoying parrot!

- To make it harder to find the bug, we will have an Annoying parrot (or whichever flying sprite you like) flapping about the screen!
- Add the code below to your annoying sprite:



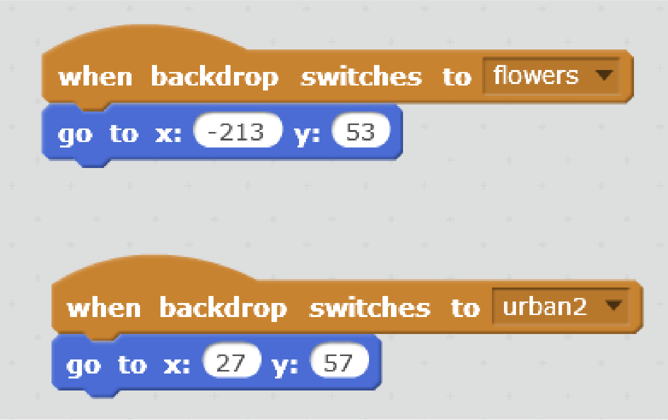
```
when ⚑ clicked
set rotation style [left-right ▼]
point in direction (50 ▼)
forever
    wait (0.1) secs
    move (10) steps
    if on edge, bounce
    next costume
    go to front
```

- There are several things going on here:
- – the ‘set rotation style’ block stops the parrot from flipping upside down (try it without!).
- – you need to change the value 90 in the ‘point in direction’ block to something else (I chose 50), otherwise the parrot will only flap from side to side, not all over the screen.
- – you are probably familiar with ‘forever/wait/move/if on edge, bounce’
- – ‘next costume’ makes the parrot flap its wings. You can create a second costume that is slightly different, if your sprite only has one.
- – ‘go to front’ makes sure that the parrot stays in front of the bug, so it can keep on being annoying! If you are playing on the small screen and have accidentally dragged your bug, it will get stuck in front of the parrot if you don’t have this block.

Save your project

More ideas

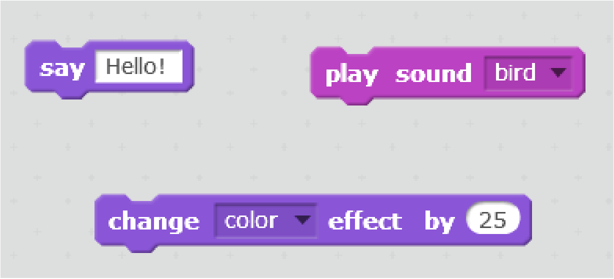
- Can you add more backdrops to your game, and hide the bug in different places? You can do this using the same code blocks you have already used, for example:



```
when backdrop switches to flowers
go to x: -213 y: 53

when backdrop switches to urban2
go to x: 27 y: 57
```

- Can you add a timer to see how quickly you can find all the bugs?
- Can you make the bug get smaller and smaller with each change of backdrop?
- Can you play different sounds on each backdrop, when the bug is found?
- Can you add colour changes or other graphic effects to the annoying parrot, to make it even more annoying? Here are some blocks to try:



```
say Hello!
play sound bird
change color effect by 25
```

- Be aware that Scratch can slow down if you add some effects in a loop, so experiment to find what looks best without spoiling your game.