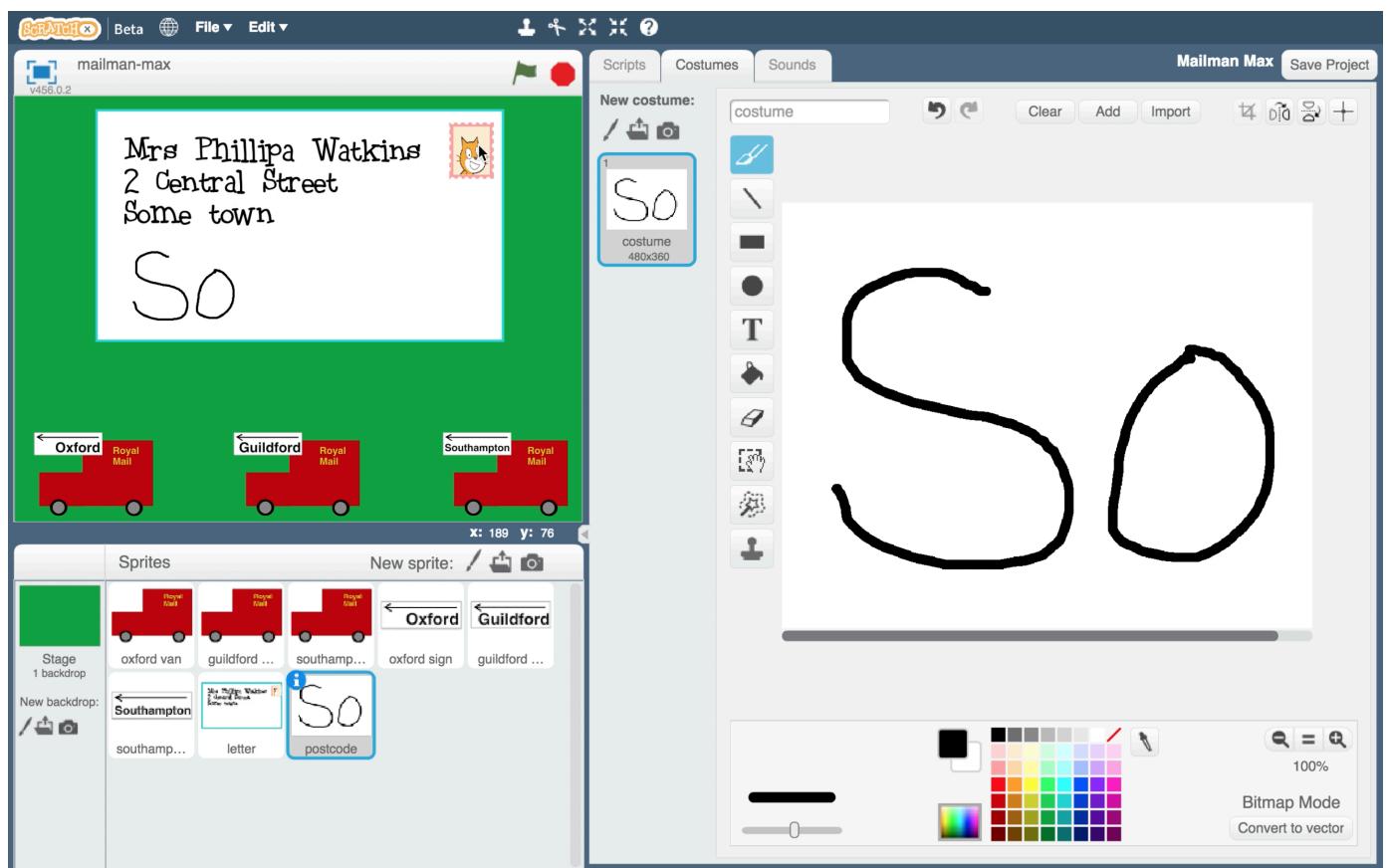


Mailman Max

In this project you will make a main postal sorting office. It will need to sort letters so that they can be put into vans going to the right local sorting offices.

The postcode is a great way to work out the next sorting office a letter should go to, so you'll use that.

You'll train the computer to recognise what the different codes at the start of a postcode look like when they are hand-written, and use that to sort letters.



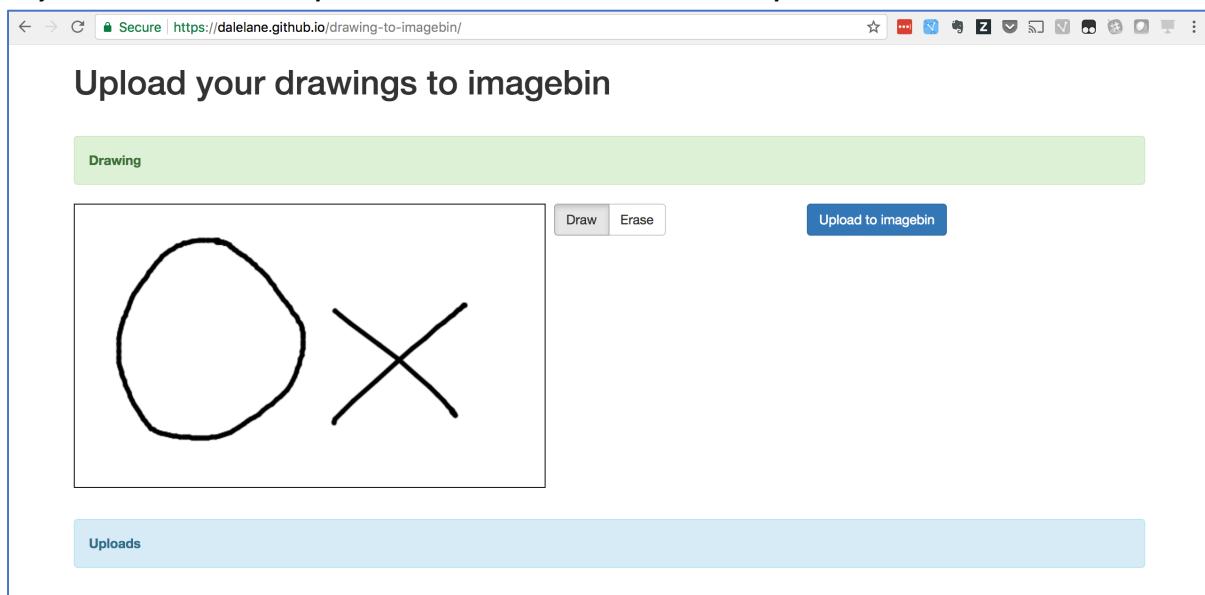
To train the computer to recognise what the start of postcodes look like, we need to make some examples.

1. Go to <https://dalelane.github.io/drawing-to-imagebin/> in a web browser

2. Use your mouse to write “OX” in the empty box.

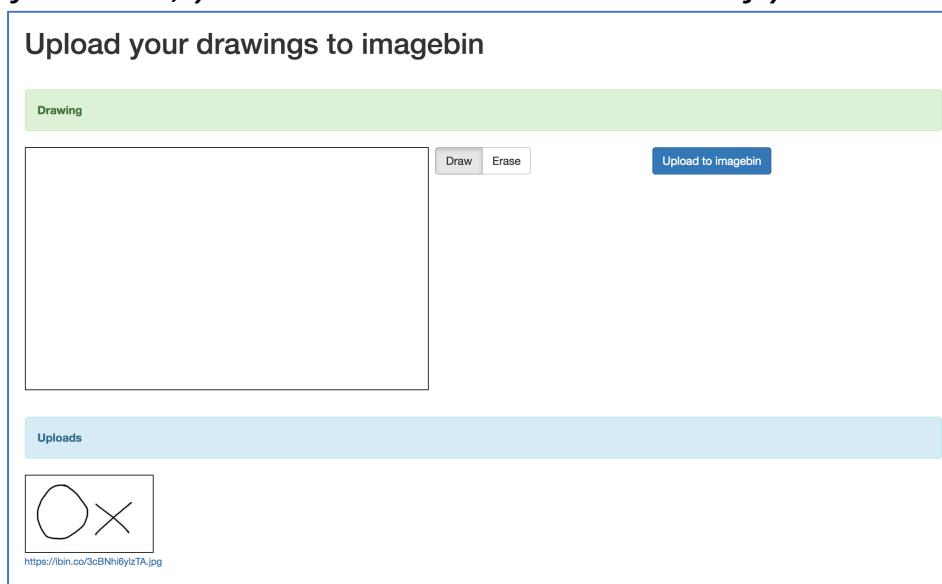
OX is the start for postcodes in the Oxford area.

Try to use all the space in the box, like in the picture below.



3. Click the “Upload to imagebin” button

If it works, you should see a small version of your writing below.



4. Draw another “OX” and repeat until you’ve got 10 examples

The screenshot shows a web-based drawing application. At the top, a green bar says "Drawing". Below it is a large white square drawing area with "Draw" and "Erase" buttons. To the right of the drawing area is a blue "Upload to imagebin" button. Below the drawing area is a section titled "Uploads" containing a grid of 10 small images, each labeled with a URL. The images show various hand-drawn "OX" characters.

| Image Preview | URL |
|---------------|---|
| | https://ibin.co/3cBNhlbyzTA.jpg |
| | https://ibin.co/3cB0imKOpBWf.jpg |
| | https://ibin.co/3cB0mpyh0Vl.jpg |
| | https://ibin.co/3cBOqGuX4v53.jpg |
| | https://ibin.co/3cBDuKYh0YF.jpg |
| | https://ibin.co/3cBOyOcqwLeO.jpg |
| | https://ibin.co/3cBP28Uq2FE4.jpg |
| | https://ibin.co/3cBPUEfmgQX.jpg |
| | https://ibin.co/3cBPYv2lns1.jpg |
| | https://ibin.co/3cBPygSECBa.jpg |

5. Draw 10 “GU” for postcodes in the Guildford area.

6. Draw 10 “SO” for postcodes in the Southampton area.

The screenshot shows a web-based drawing application. At the top, a green bar says "Drawing". Below it is a large white square drawing area with "Draw" and "Erase" buttons. To the right of the drawing area is a blue "Upload to imagebin" button. Below the drawing area is a section titled "Uploads" containing a grid of 10 small images, each labeled with a URL. The images show various hand-drawn "GU" and "SO" characters.

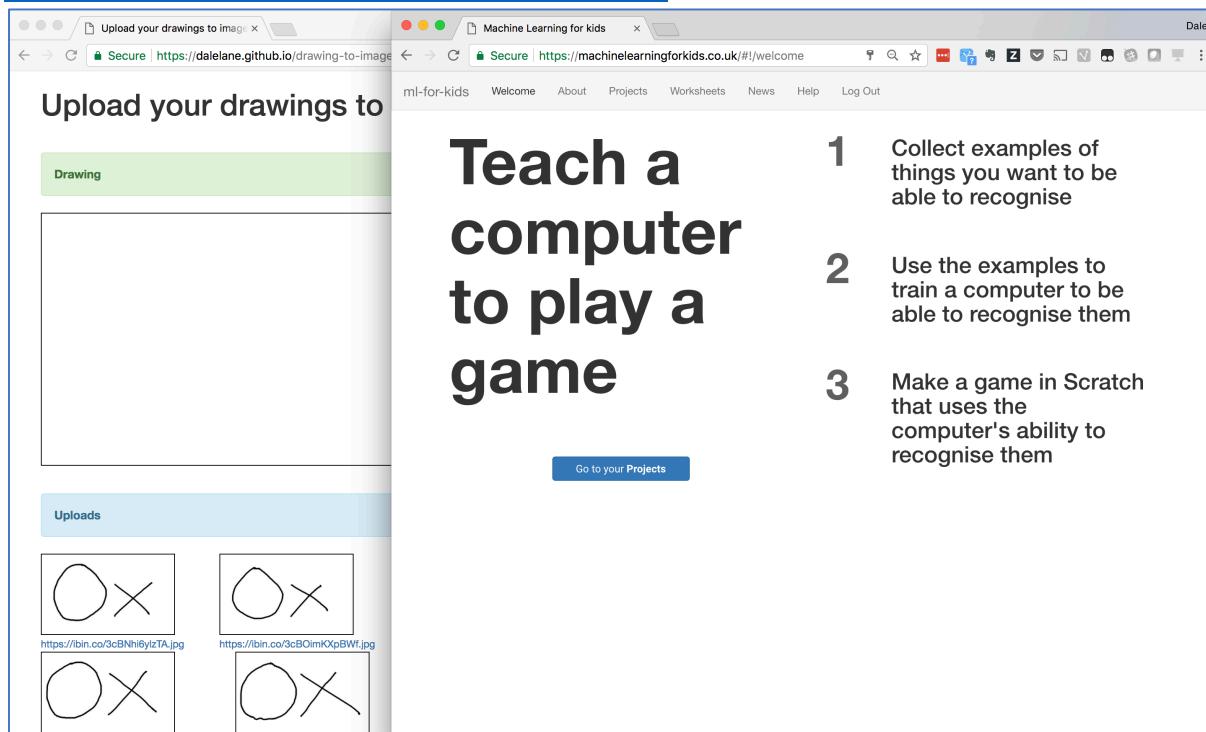
| Image Preview | URL |
|---------------|---|
| | https://ibin.co/3cBNhlbyzTA.jpg |
| | https://ibin.co/3cB0imKOpBWf.jpg |
| | https://ibin.co/3cB0mpyh0Vl.jpg |
| | https://ibin.co/3cBOqGuX4v53.jpg |
| | https://ibin.co/3cBDuKYh0YF.jpg |
| | https://ibin.co/3cBOyOcqwLeO.jpg |
| | https://ibin.co/3cBP28Uq2FE4.jpg |
| | https://ibin.co/3cBPUEfmgQX.jpg |
| | https://ibin.co/3cBPYv2lns1.jpg |
| | https://ibin.co/3cBPygSECBa.jpg |
| | https://ibin.co/3cBQnTpbn9e.jpg |

7. Keep this web browser window open!

If you lose this page, you'll have to draw the examples all over again.

8. In a separate web browser window, go to

<https://machinelearningforkids.co.uk/>



9. Click on “Get started”

10. Click on “Log In” and type in your username and password

If you don't have a username, ask your teacher or group leader to create one for you.

If you can't remember your username or password, ask your teacher or group leader to reset it for you.

11. Click on “Projects” on the top menu bar

12. Click the “+ Add a new project” button.

- 13.** Name your project “Mailman Max” and set it to learn how to recognise “**images**”.

Click the “**Create**” button

The screenshot shows a web-based form for creating a machine learning project. At the top, a grey header bar says "Start a new machine learning project". Below it, a "Project Name" field contains "Mailman Max". Underneath, a dropdown menu labeled "Recognizing" has "images" selected. To the right of the dropdown is a tooltip explaining the choice: "What type of thing do you want the computer to recognise? For words, sentences or paragraphs, choose 'text'. For photos, diagrams and pictures, choose 'images'. For sets of numbers or multiple choices, choose 'numbers'." At the bottom right are two buttons: a blue "CREATE" button and a white "CANCEL" button.

- 14.** You should see “**Mailman Max**” in the list of your projects. Click it.

- 15.** Click the “**Train**” button

- 16.** Click the “**Add new label**” button, and create a label called “Oxford”

- 17.** Click “**Add new label**” again, and create one called “Guildford”

- 18.** Click “**Add new label**” again, and create one called “Southampton”

The screenshot shows the main interface for training the project. The title at the top reads "Recognising **images** as **Oxford, Guildford or Southampton**". Below the title, there are three large rectangular input fields, each with a label: "Oxford", "Guildford", and "Southampton". Each field has the instruction "Drag pictures from other browser windows and drop them here". In the top right corner of the interface, there is a small button with a plus sign and the text "Add new label". At the bottom of each input field is a "Add example" button.

- 19.** Drag the examples you drew in the first window into the correct training bucket in the machinelearningforkids.co.uk window.
*If you get the windows side by side to start with, this will be easier.
Try not to use the same example more than once!*

The screenshot shows a web application for training a machine learning model. At the top, there is a navigation bar with links: ml-for-kids, Welcome, About, Projects, Worksheets, News, Help, and Log Out. Below the navigation bar, the title "Recognising **images** as **Oxford, Guildford or Southampton**" is displayed. A link "[< Back to project](#)" is located above the training buckets. There are three separate training buckets, each containing several examples of handwritten letters:

- Oxford** bucket: Contains examples of handwritten letters "Ox".
- Guildford** bucket: Contains examples of handwritten letters "GU".
- Southampton** bucket: Contains examples of handwritten letters "So".

Each bucket has a "Add new label" button in the top right corner and an "Add example" button at the bottom. The entire interface is enclosed in a blue border.

20. Click on the “**< Back to project**” link

21. Click the “**Learn & Test**” button

22. Click the “Train new machine learning model” button

The screenshot shows the 'Machine learning models' page. At the top, there is a header bar with links for 'ml-for-kids', 'Welcome', 'About', 'Projects', 'Worksheets', 'News', 'Help', and 'Log Out'. Below the header, the main title 'Machine learning models' is displayed. A 'Back to project' link is located at the top left of the main content area. The content is divided into two main sections: 'What have you done?' and 'What's next?'. The 'What have you done?' section contains text about collecting images for Oxford, Guildford, and Southampton, followed by a bulleted list of collected examples. The 'What's next?' section contains text about starting training and a 'Train new machine learning model' button. At the bottom, there is a 'Info from training computer:' section with a 'Train new machine learning model' button.

23. Wait for the training to complete. This might take a few minutes.

The screenshot shows the 'Machine learning models' page after training has started. The 'What have you done?' section now includes a timestamp 'It's been training since Saturday, September 30, 2017 6:05 PM.' and a note that it normally takes a few minutes. The 'What's next?' section suggests waiting for completion or taking a quiz. At the bottom, there is a 'Info from training computer:' section with a table providing details about the training status:

| | |
|--|--------------------------------------|
| Model started training at: | Saturday, September 30, 2017 6:05 PM |
| Current model status: | Training |
| Model will automatically be deleted after: | Saturday, September 30, 2017 7:05 PM |

24. Click the “< Back to project” link

25. Click the “Scratch” button

This page has instructions on how to use the new blocks in Scratch.

Keep the page open if you need to check back on how to use them.

Using machine learning in Scratch

[< Back to project](#)

Your project will add these blocks to the **More Blocks** tab in Scripts.

`recognise images [costume image] (label)`
Put images in the input for this, and it will return the label that your machine learning model recognises it as.

`recognise images [costume image] (confidence)`
This will return how confident your machine learning model is that it recognises the type of images. (As a number from 0 - 100).

`Oxford Guildford Southampton`
These blocks represent the labels you've created in your project, so you can use their names in your scripts.

`costume image`
This block is in the Looks palette for Sprites and will return the image of the currently selected costume.

This means you can do something like this:

```
if [recognise images [costume image] = Oxford then
    say [I think that is a picture of Oxford]]
```

It will look something like this - except with the name of your project.

The coloured circle next to your project name tells you if your machine learning model is okay.

- **Green** means your model is trained and ready to go
- **Yellow** means your model hasn't finished training yet
- **Red** means something went wrong. Go back to the [Learn & Test](#) page to see what went wrong with training.

[Open in Scratch](#)

What have you done so far?

You've started to train a computer to recognise pictures of handwriting as being the start of postcodes for the Oxford, Guildford, or Southampton area. You are doing it by collecting examples of your handwriting. These examples are being used to train a machine learning "model".

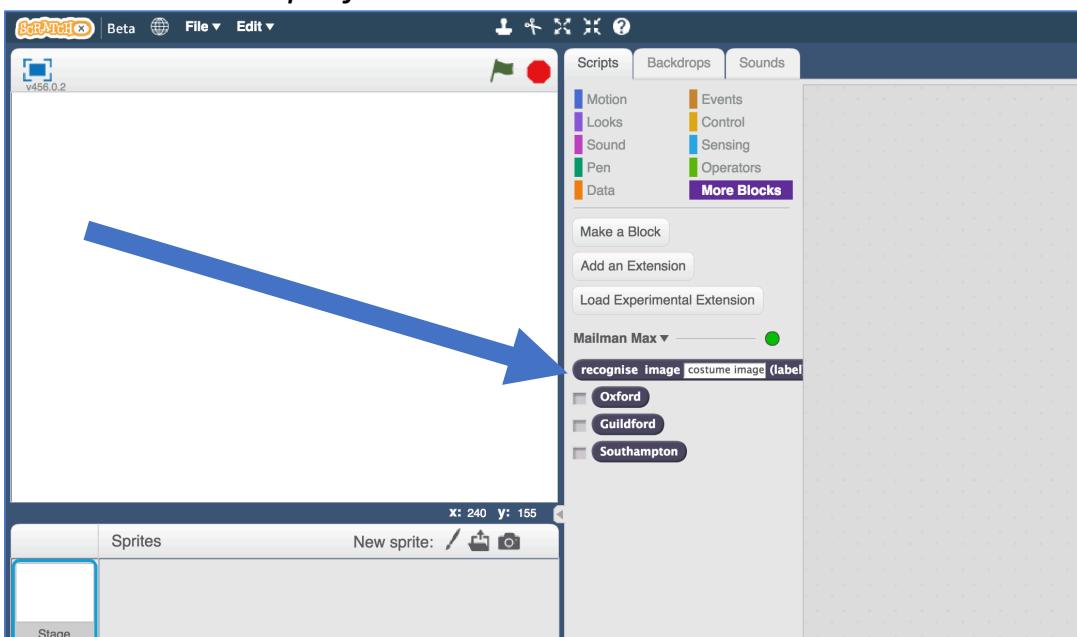
This is called "supervised learning" because of the way you are supervising the computer's training.

The computer will learn from patterns in the shapes from each of the examples you've drawn. This will be used to be able to recognise the postcodes we'll write on the envelopes to be sorted next.

26. You'll need the **mailman-max.sbx** starter file for this project.
If you haven't got this, ask your teacher or group leader.

27. Click the “Open in Scratch” button at the bottom to launch the Scratch editor.

You should see four new blocks in the “More blocks” section from your “Mailman Max” project.



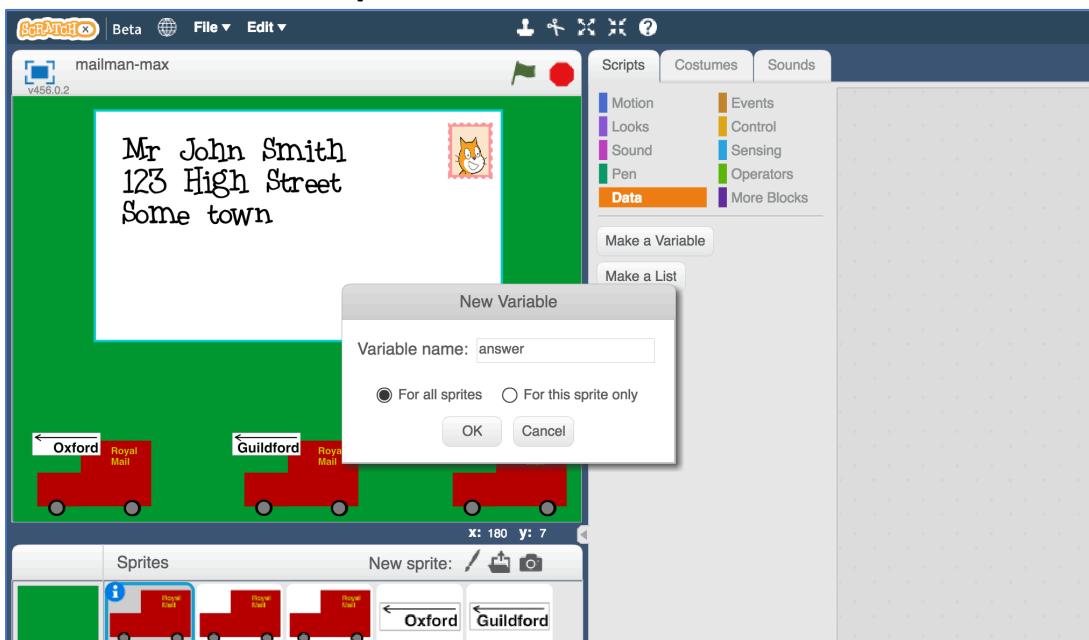
28. Open the “mailman-max.sbx” project file.

Click **File -> Load Project**

Click **OK** when it asks to replace the contents of the current project.

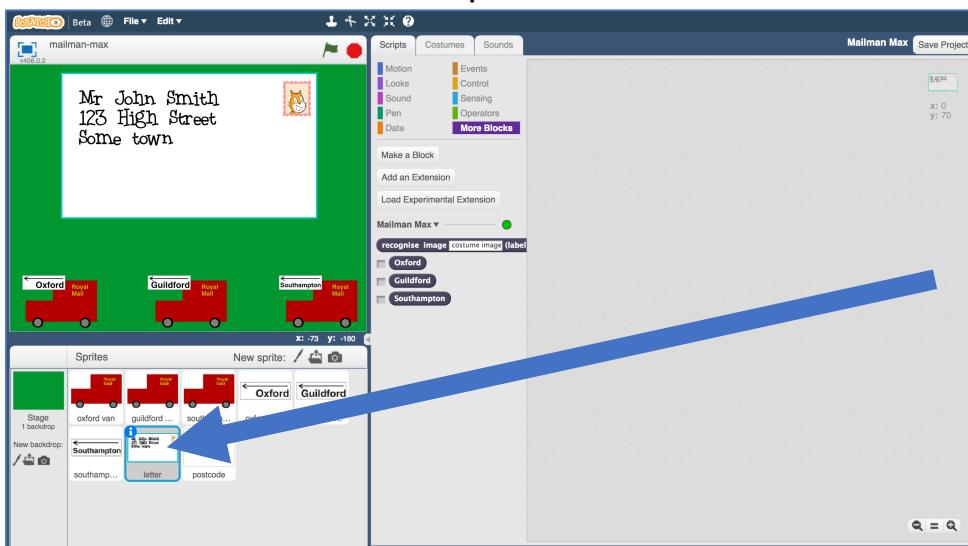
29. Click the “Data” tab

- 30.** Click “**Make a variable**” and create a variable called “**answer**” which is available “**For all sprites**”.

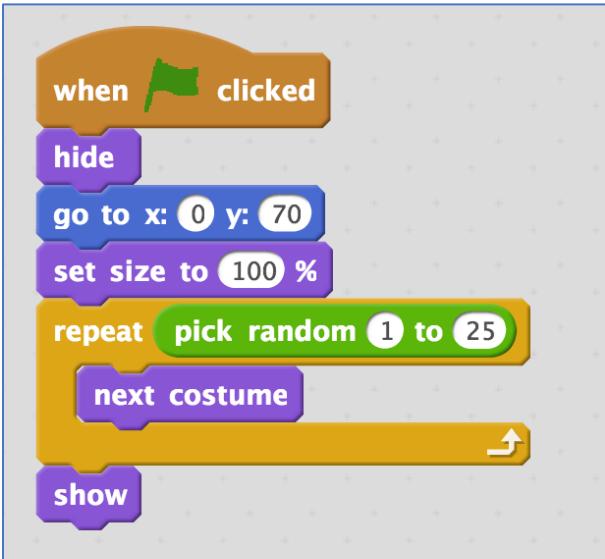


- 31.** Untick the “**answer**” variable so it isn’t shown on the Stage.

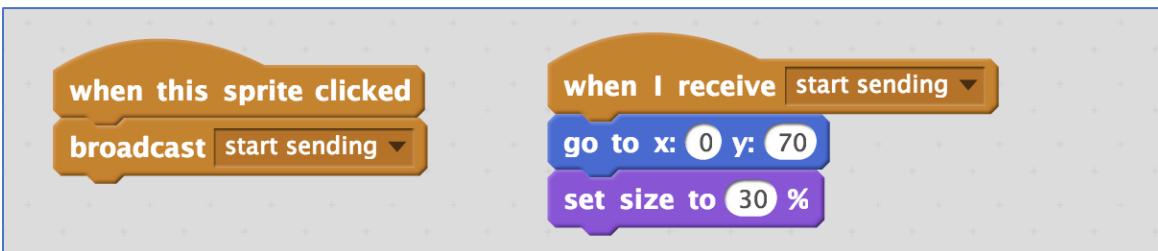
- 32.** Click on the “**letter**” sprite.



33. Enter the following script to choose us a random letter to sort.



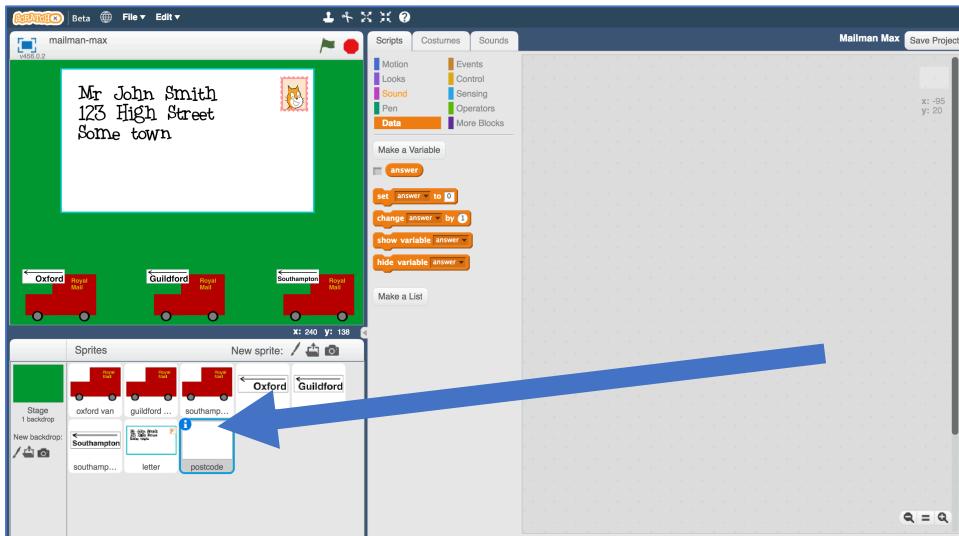
34. Enter the following script so that when we click on a letter it will get ready to be sorted.



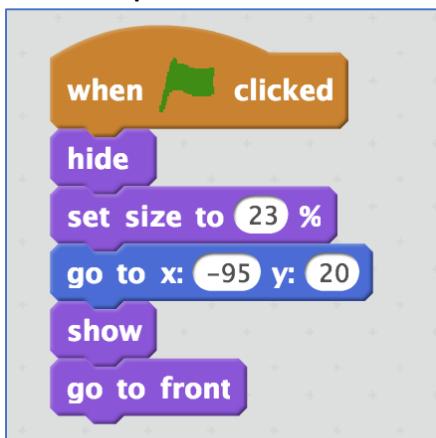
35. Enter the following script so when the letter has been sorted, it goes to the post van to be taken to the correct regional sorting office.



36. Click on the “postcode” sprite.



37. Enter the following script to get the new envelope ready for you to write a postcode on.



38. Enter the following script to get the computer to try to recognise the postcode you write on the envelope.

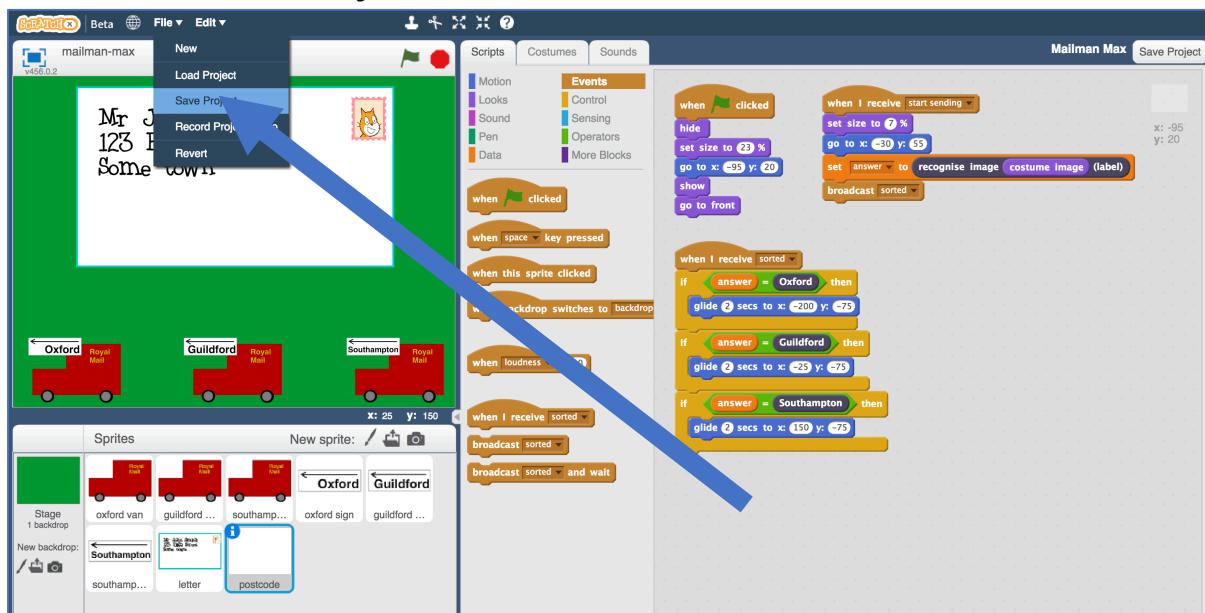


- 39.** Enter the following script to get the handwritten postcode to go, with the rest of the envelope, to the post van to be taken to the correct regional sorting office.



40. Save your project

Click **File -> Save Project**

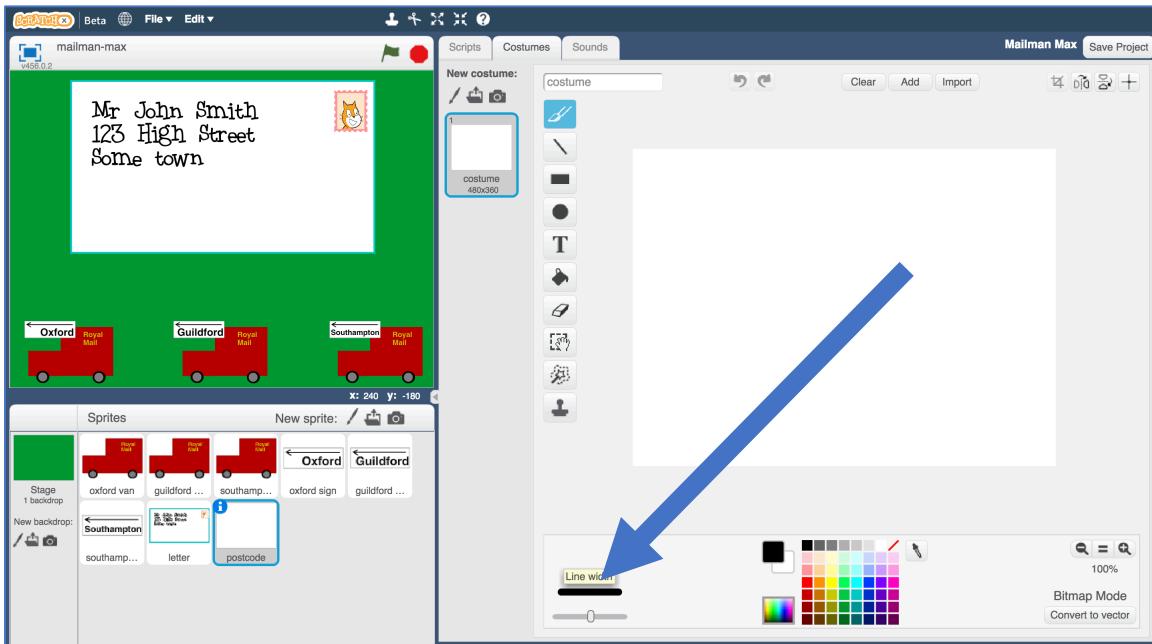


It's time to test!

41. Click the Green Flag

42. Still on the “postcode” sprite, click on “Costumes”.

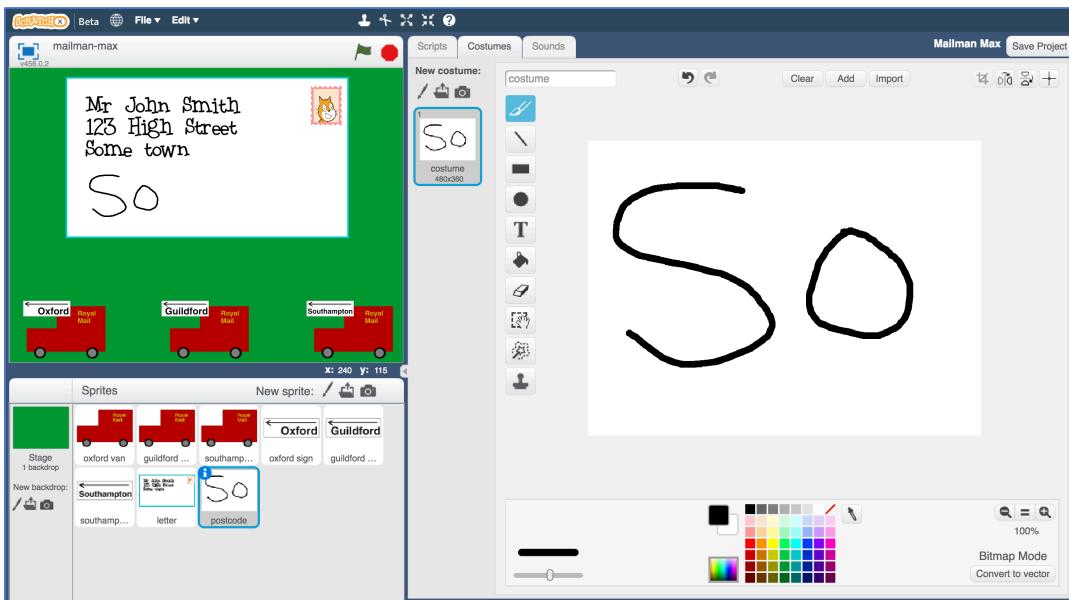
43. Use the line width slider so that we can write a thick black line.



44. Write the first two letters of a postcode using the paintbrush tool.

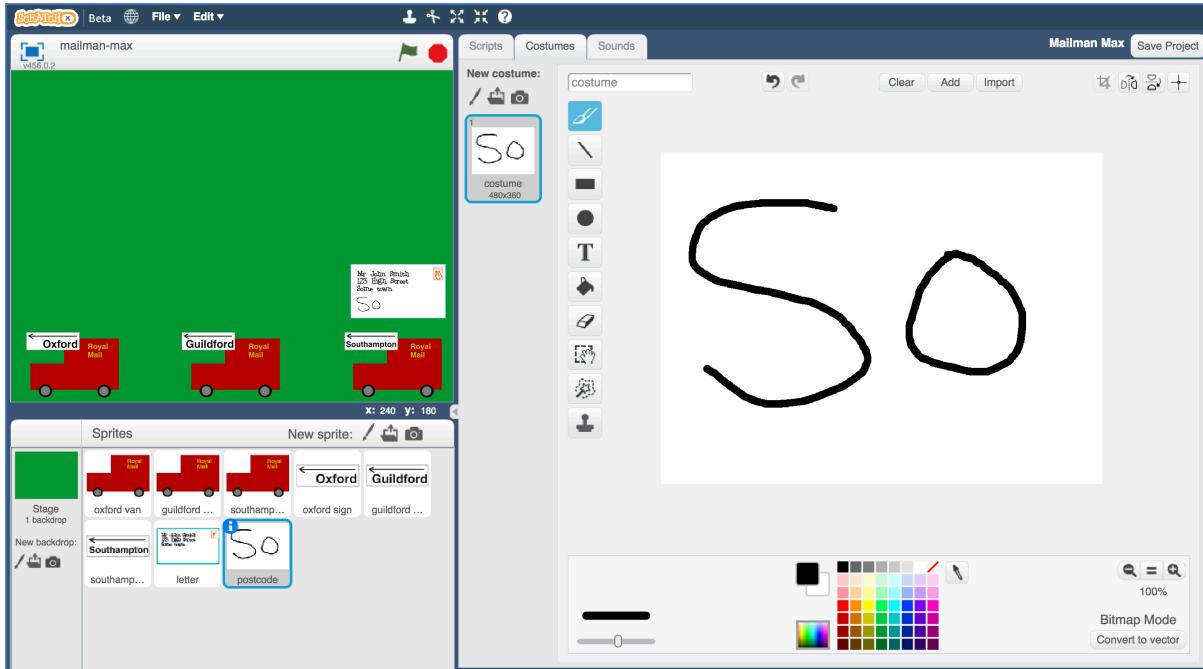
Fill the space, like you did with the training examples.

You should also see it appear on the envelope in the right place.



45. Click on the stamp in the envelope in the stage.

You should see the letter shrink. Then the computer will try to recognise the postcode letters you've written. Once it thinks it has the answer, the envelope will move to the van for the correct sorting office.

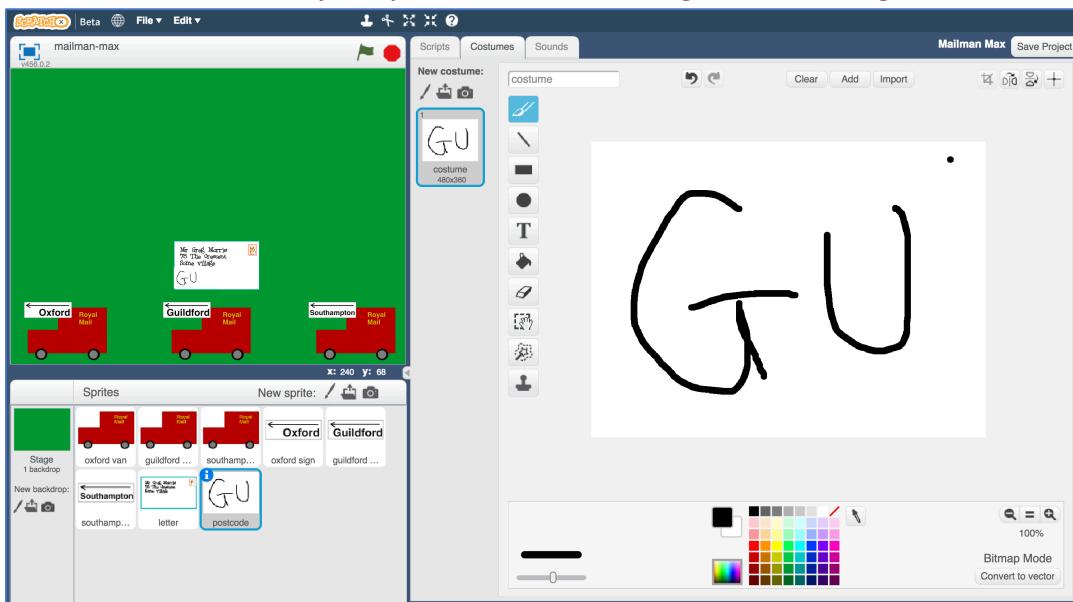


46. Did it get it right?

If it's not very good, you might need to add some more examples and train a new machine learning model with them.

47. Repeat steps 41-46 to try it again with a different postcode.

You'll need to paint over your first postcode in white first. If you use the eraser, make sure you put a white background in again.



What have you done?

You've trained a machine learning model to be able to do handwriting recognition. This is called "optical character recognition" or "OCR" for short.

You did that by collecting examples of handwriting, to train the computer to be able to recognise it.

You built a small and simple example, using just the first two letters for just three postcode areas.

Imagine doing the same thing for every postcode area in the country. You'd have to create a lot more training buckets to cover the 120 postcode areas in the UK. And you'd need to collect thousands of training examples, with lots of different people's handwriting, so that the computer could get really good at recognising them.

That is how large postal sorting offices sort our letters in real life.

Ideas and Extensions

Now that you've finished, why not give one of these ideas a try?

Or come up with one of your own?

Try someone else's handwriting

You've trained the computer to recognise how you write the postcode letters, but would it be able to recognise someone else's?

Ask a friend to test it and see if it works.

If it doesn't, you'll need to get some examples of their writing to add to your training data. The more people you can get training examples from, the better the computer will be at recognising a variety of handwriting styles.

Try more of the postcode

We made it easier for the computer by only giving it the first two letters.

But how can we get it to recognise something like "OX1 2JD" as being a postcode in the Oxford area?

If you collect a variety of different training examples of actual full postcodes (not just the first two letters) you should be able to train it to recognise them. That will probably need more than 10 examples!