



# Alien Language

In this project you will train the computer to understand an alien language.

You'll use that to control an alien character so that it can understand what you tell it to do.

The image shows a Scratch project titled "alien-language". The stage features a green alien with one eye and a single antenna, standing on a purple planet with a city skyline in the background. The script area contains the following blocks:

- A green "when green flag clicked" hat block with a blue "go to x: 0 y: 0 point in direction 90" block attached.
- Two orange "when green flag clicked" hat blocks with blue "wait 0.1 seconds" blocks attached.
- Two orange "when green flag clicked" hat blocks with blue "go to x: (x position - 5) y: 0" and "turn 5 degrees" blocks attached.
- Two orange "when green flag clicked" hat blocks with blue "go to x: (x position + 5) y: 0" and "turn -5 degrees" blocks attached.
- A blue "train new machine learning model" block attached to a green "when green flag clicked" hat block.
- A blue "start listening" block attached to the same green hat block.
- A blue "wait until [Is the machine learning model ready to use?]" block attached to the same green hat block.
- Two yellow "when I hear left" sound blocks with blue "walk left" blocks attached.
- Two yellow "when I hear right" sound blocks with blue "walk right" blocks attached.

The script palette on the left lists categories: Motion, Looks, Sound, Events, Control, Sensing, Operators, Variables, My Blocks, Images, and Alien Language. The Alien Language category contains the custom blocks used in the script.



This project worksheet is licensed under a Creative Commons Attribution Non-Commercial Share-Alike License  
<http://creativecommons.org/licenses/by-nc-sa/4.0/>

This project requires a **microphone**. If you don't have a computer with a microphone, you might prefer to try a different worksheet.

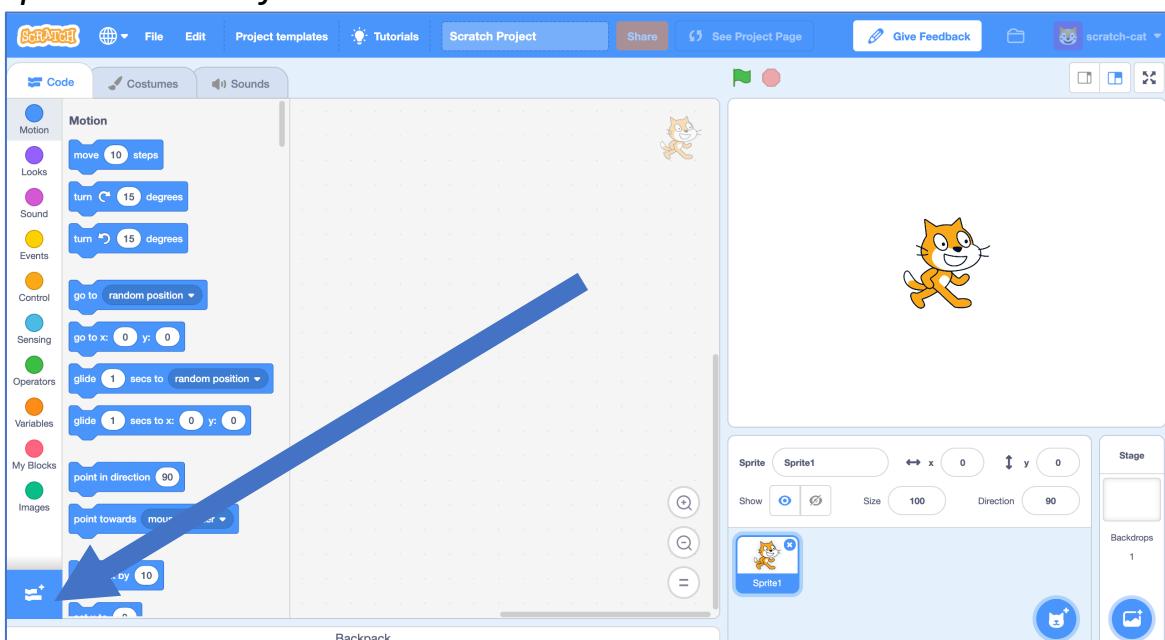
## 1. Go to <https://machinelearningforkids.co.uk/scratch3/>

The next few steps only work with **Google Chrome**.

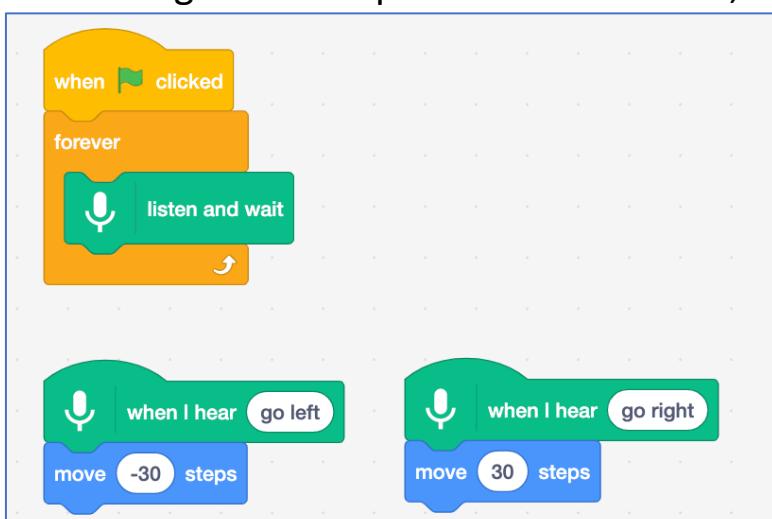
If you don't have access to Google Chrome, skip to **step 5** and start from there.

## 2. Load the **Speech to Text** extension

*Click on the Extensions (plus) button in the bottom left, and then choose Speech to Text from the list.*



## 3. Using the new Speech to Text blocks, create the following scripts.

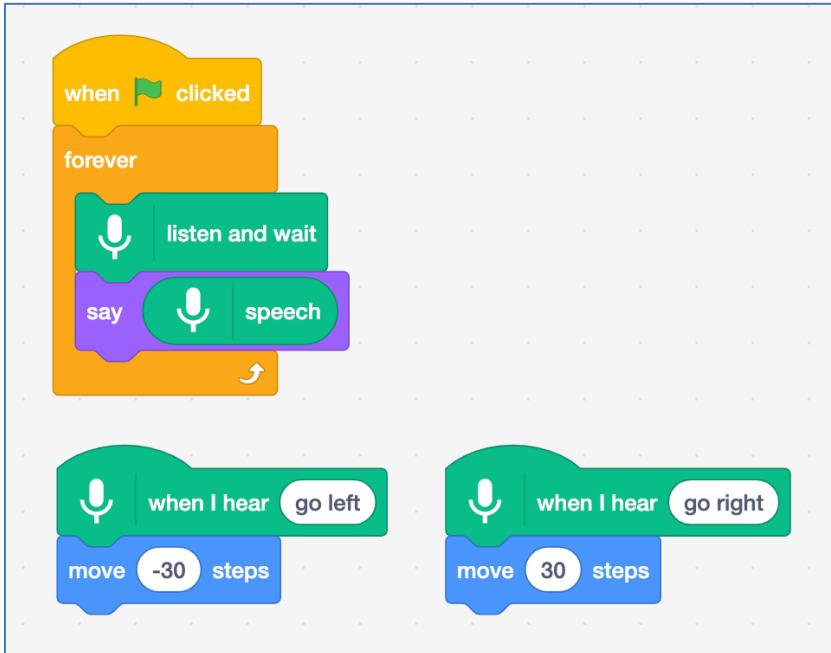


#### 4. Click on the **Green Flag** and give it a try

Say “left” or “right”. The cat should move in the direction that you tell it to. Try and move it back and forth across the screen using your voice.

It can be difficult to get it to work. Try to speak calmly and clearly.

If it doesn’t work, modify your script to show what it thinks you’re saying:



#### What have you done so far?

You’ve used **speech recognition** to control a character in Scratch. To get this working quickly, you’ve used a machine learning model that has already been trained for you. This is a general machine learning model that has been trained to recognize English dictionary words.

Next, you’ll train a machine learning model for yourself to see how it was done.

For the next part of the project, you’ll use your voice to control an alien character that doesn’t understand English! You’ll invent two new words, that wouldn’t be found in an English dictionary, to control your character and train a machine learning model to recognize them.

## 5. Invent your alien language!

You need two words – an alien word for “left” and an alien word for “right”. Invent new words that wouldn’t show up in an English dictionary. They can be random noises as long as you can repeat them in the same way every time and will be recognisably different from each other. If you don’t want to make weird noises with your voice, that’s okay - find other ways to make noises. You can click your fingers, clap your hands, squeeze a squeaky toy or do anything else you can think of!

## 6. Go to <https://machinelearningforkids.co.uk/>

## 7. 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.*

## 8. Click on “Projects” on the top menu bar

## 9. Click the “+ Add a new project” button.

## 10. Name your project “Alien Language” and set it to learn how to recognise “sounds”.

Click the “Create” button

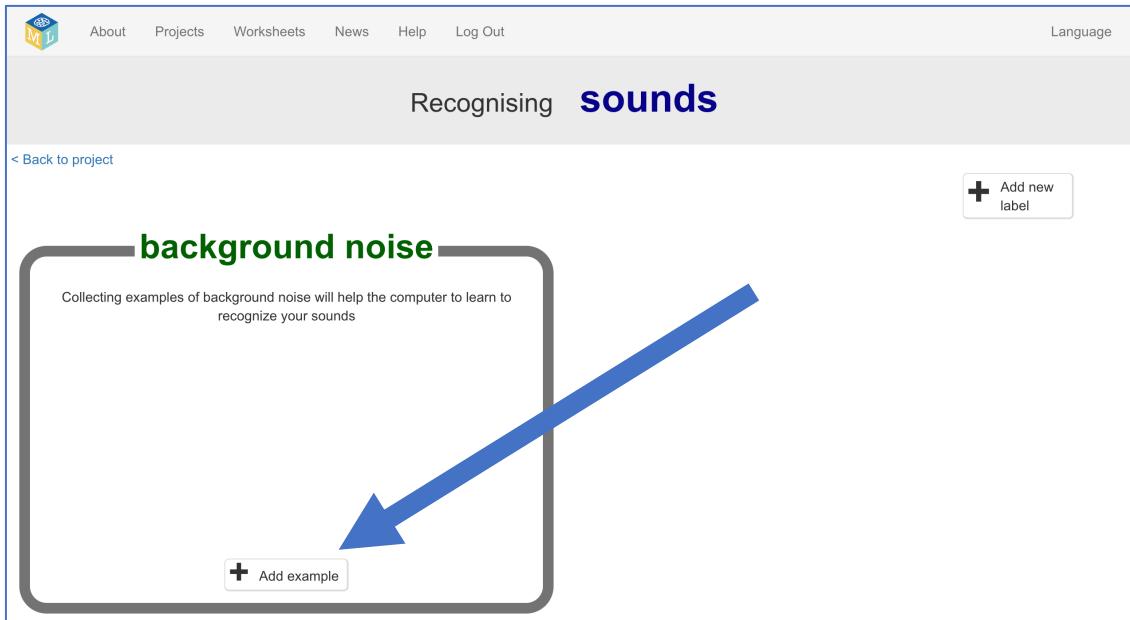
The screenshot shows a web page titled "Start a new machine learning project". At the top, there is a navigation bar with links for "About", "Projects", "Worksheets", "News", "Help", "Log Out", and "Language". The main area has a light gray background. A form is centered on the page. It starts with a "Project Name" input field containing "Alien Language". Below it is a dropdown menu labeled "Recognising" with "sounds" selected. To the right of the dropdown is a tooltip box with the following text:  
What type of thing do you want to teach 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"  
For voices and sounds, choose "sounds"  
At the bottom of the form are two buttons: a blue "CREATE" button and a white "CANCEL" button.

**11.** You should now see “**Alien Language**” in the list of your projects.

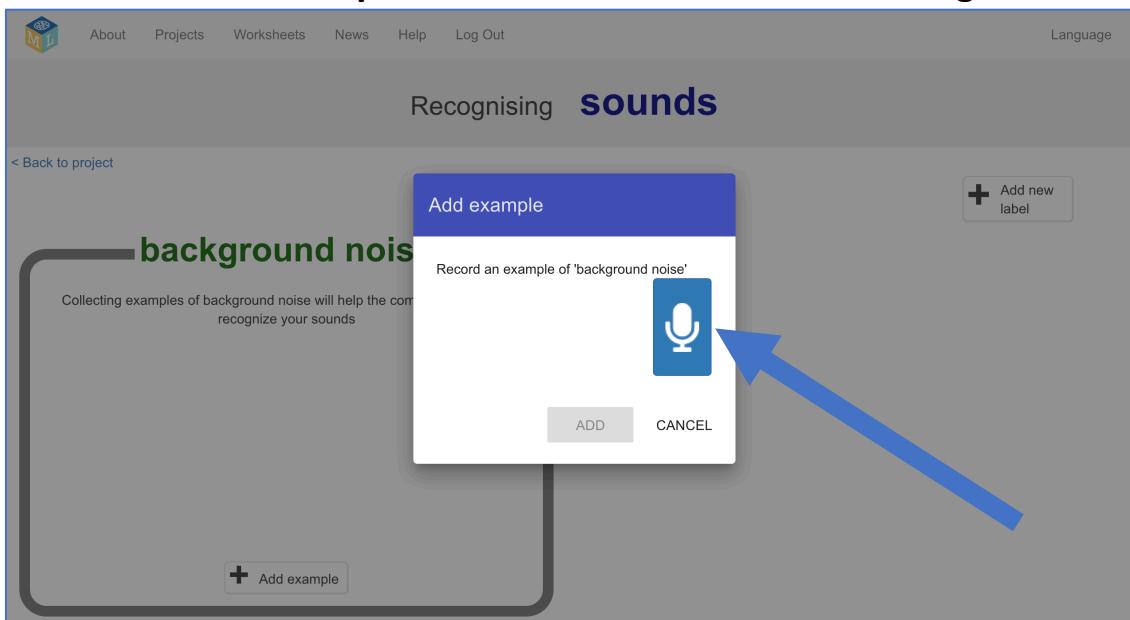
Click on it.

**12.** Click on the **Train** button to start collecting examples.

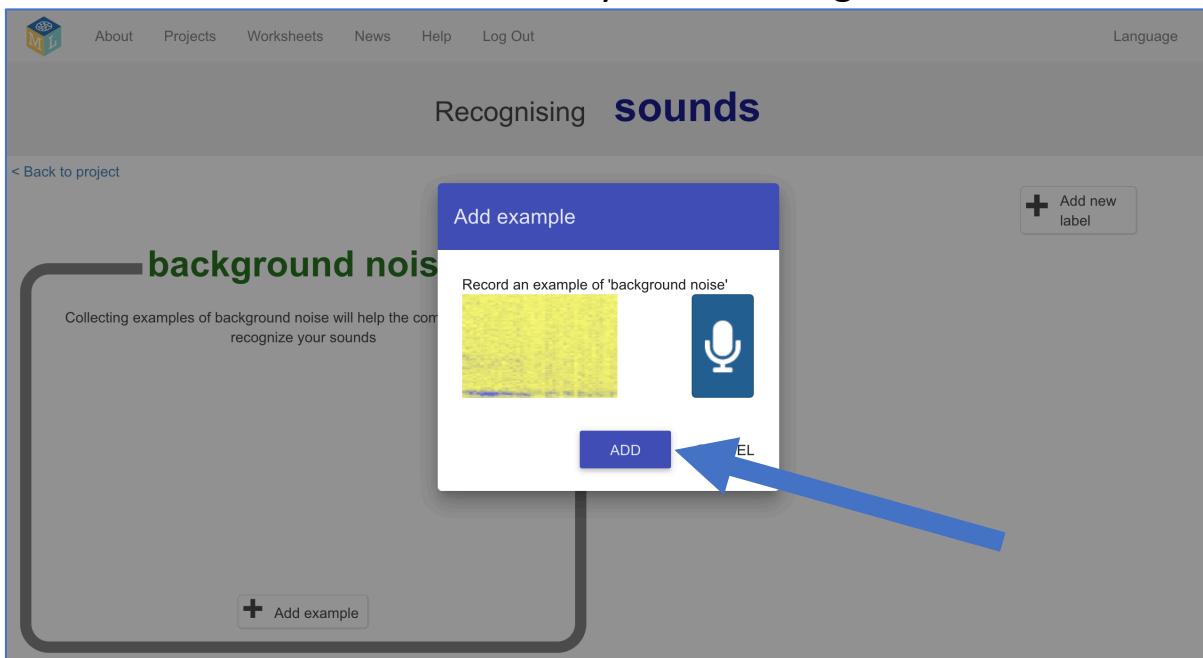
**13.** Click on the **Add example** button in the **background noise** bucket  
*Recording background noise will help your machine learning model to tell the difference between the sounds you will train it to recognize, and the background noise where you are.*



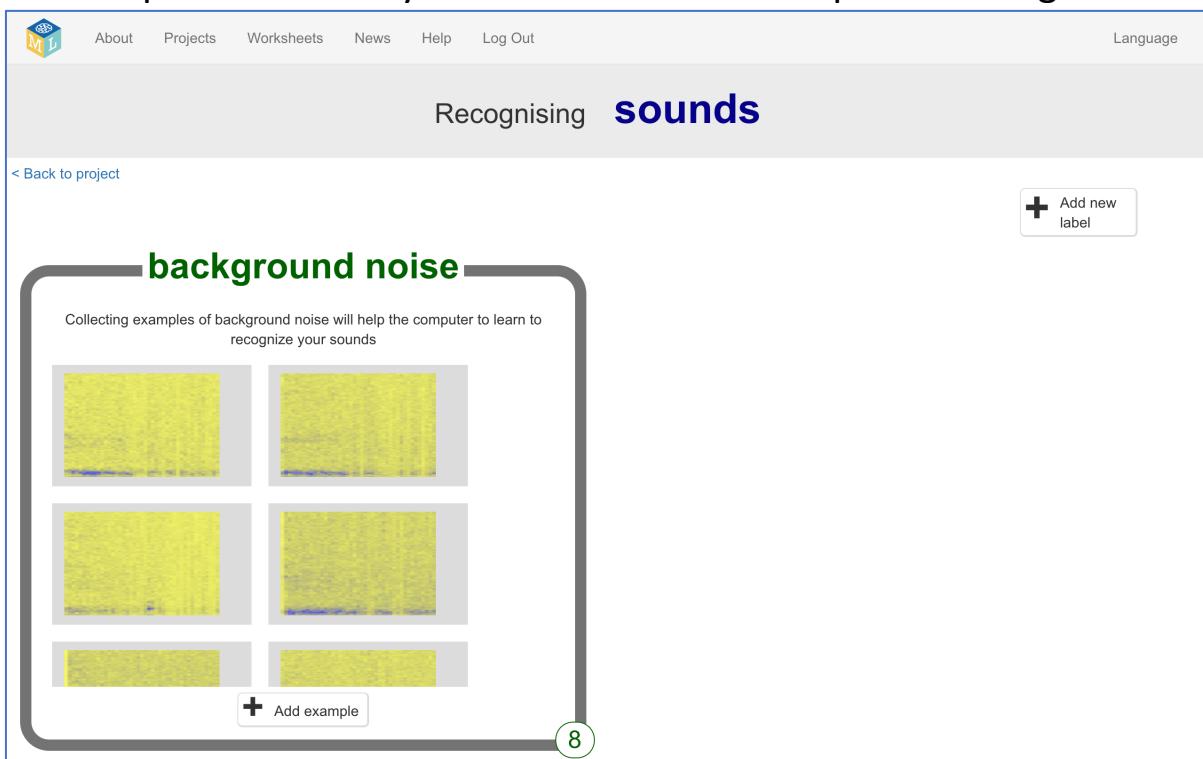
**14.** Click the **microphone** to record 2 seconds of background noise



## 15. Click the **Add** button to save your recording



## 16. Repeat that until you have **at least 8** examples of background noise



## 17. Click the **Add new label** button in the top right, and create a new training bucket called “left”

## 18. Click the Add example button in the new left bucket

The screenshot shows the Soundkit interface with the title "Recognising sounds as left". On the left, there is a bucket labeled "background noise" containing four spectrograms. On the right, there is a bucket labeled "left" containing four spectrograms. A large blue arrow points from the "background noise" bucket towards the "left" bucket. In the top right corner of the "left" bucket, there is a button labeled "+ Add new label". At the bottom of each bucket, there is a button labeled "+ Add example". A green circle with the number "8" is positioned at the bottom center of the "left" bucket.

## 19. Record at least 8 examples of your alien noise for “left”

The screenshot shows the Soundkit interface with the title "Recognising sounds as left". The "background noise" bucket on the left now contains eight spectrograms, indicated by a green circle with the number "8" at the bottom center. The "left" bucket on the right also now contains eight spectrograms, indicated by a green circle with the number "8" at the bottom center. Both buckets have their respective labels ("background noise" and "left") above them. Each bucket has a "+ Add example" button at the bottom. In the top right corner of the "left" bucket, there is a button labeled "+ Add new label".

## 20. Click the Add new label button in the top right, and create a new training bucket called “right”

## 21. Record at least 8 examples of your alien noise for “right”

The screenshot shows a web-based application for collecting sound examples. At the top, there's a navigation bar with links for About, Projects, Worksheets, News, Help, Log Out, and Language. Below that, the title "Recognising sounds as left or right" is displayed. Underneath the title, there are three main sections: "background noise", "left", and "right". Each section contains a grid of 8 spectrograms. At the bottom of each section is a button labeled "+ Add example". A small "8" is also visible at the bottom of each section.

## 22. Click the “Back to project” link in the top left

## 23. Click the Learn & Test button

The screenshot shows a project titled "Alien Language". It features three main buttons: "Train", "Learn & Test", and "Make". A large blue arrow points directly at the "Learn & Test" button. Below each button is a brief description: "Train" says "Collect examples of what you want the computer to recognise" and has a "Train" button. "Learn & Test" says "Use the examples to train the computer to recognise sounds" and has a "Learn & Test" button. "Make" says "Use the machine learning model you've trained to make a game or app, in Scratch or in Python" and has a "Make" button.

## 24. Click “Train new machine learning model”

The screenshot shows a page titled "Machine learning models". It has two main sections: "What have you done?" and "What's next?". The "What have you done?" section lists the user's collections: 8 examples of background noise, 8 examples of left, and 8 examples of right. The "What's next?" section asks the user to start training the computer and provides a "Train new machine learning model" button. A large blue arrow points to this button. At the bottom, there's a box labeled "Info from training computer:" with a "Train new machine learning model" button inside it.

**25.** Once the training is finished, click the **Start listening** button to test your machine learning model

*Make one of the sounds you've trained the computer to recognize as meaning "left" or "right". If your machine learning model recognizes it, it will display what it thinks you did.*

The screenshot shows the 'Learn & Test' section of the Scratch ML interface. At the top, there's a note about collecting examples: '• 8 examples of left, • 8 examples of right'. Below that, a message says 'Try making a sound to see how it is recognised based on your training' with 'Start listening' and 'Stop listening' buttons. A large blue arrow points from this section down to the results area. The results area shows 'Recognised as right with 94% confidence'.

**26.** If you're not happy with how the model is working, go back to the **Train** page and add more examples to all three training buckets.

**27.** When you're happy with your machine learning model, click on the **Make** button

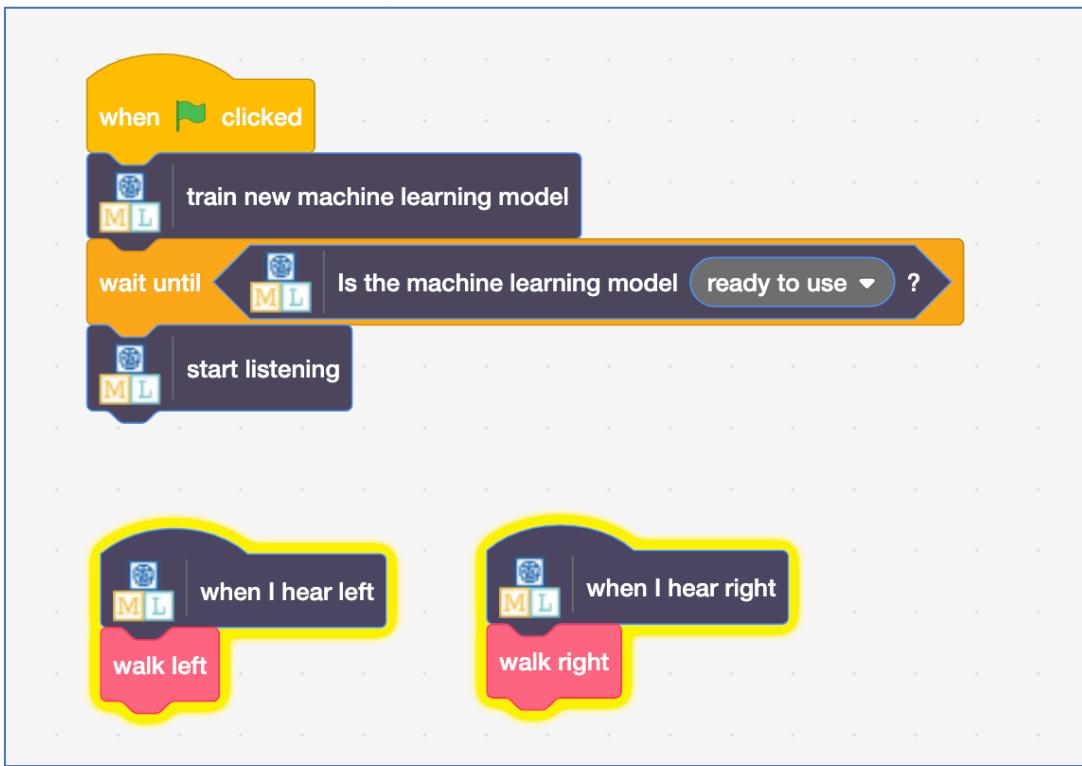
The screenshot shows the 'Make' section of the Scratch ML interface. It features a title 'Alien Language' with three main buttons: 'Train', 'Learn & Test', and 'Make'. A large blue arrow points from the 'Learn & Test' button down to the 'Make' button. The 'Train' button has the sub-instruction 'Collect examples of what you want the computer to recognise'. The 'Learn & Test' button has the sub-instruction 'Use the examples to train the computer to recognise sounds'. The 'Make' button has the sub-instruction 'Use the machine learning model you've trained to make a game or app, in Scratch or in Python'.

**28.** Click on the **Scratch 3** button and then click **Open in Scratch 3**

**29.** Click on the **Project templates** button at the top of the screen and open the "Alien Language" project template

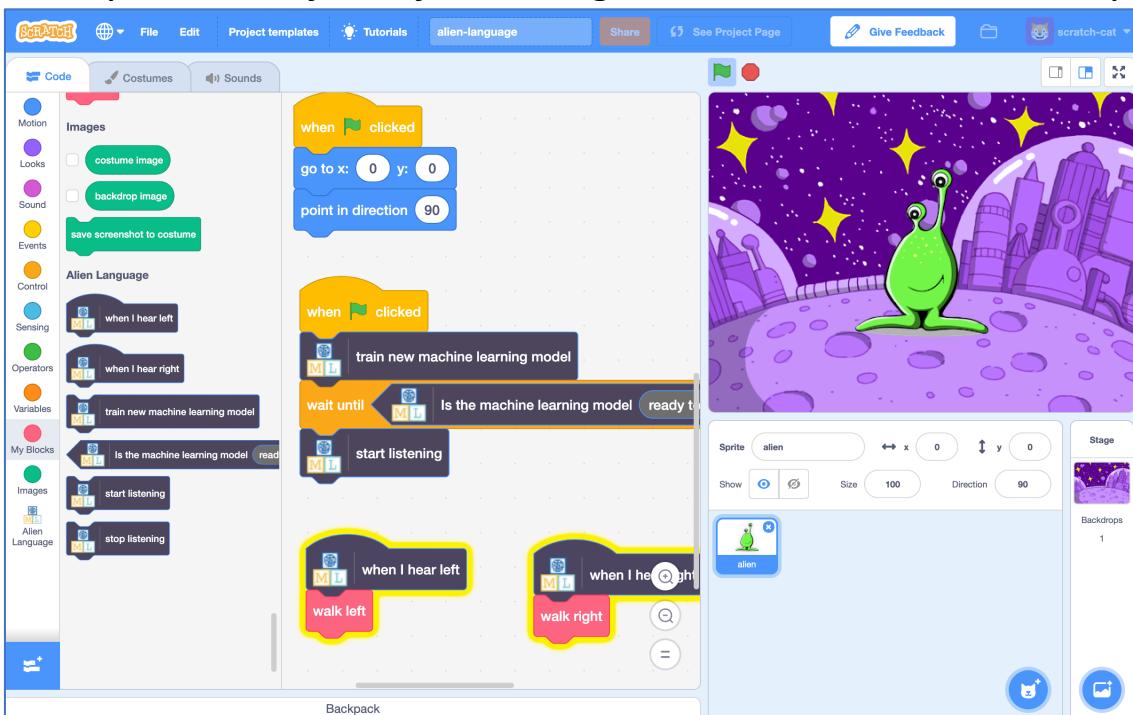
## 30. Add the following script to the alien sprite

*There are already some scripts in the alien sprite to put it in the right place at the start and animate how it walks. **Don't** delete these.  
You can add these scripts underneath them.*



## 31. It's time to test! Click the Green Flag

*Make your noises for "left" and "right" to tell the alien which way to walk.*



## What have you done?

You've trained your own machine learning model to do speech recognition. You used that to control a character in Scratch.

Unlike the pre-trained model you used before, which has been trained to recognize tens of thousands of words, you've only trained it to recognize two different words. But the principle is the same.

You've also seen the importance of training the machine learning model to work with a certain background noise.

Can you think of an example of a system like this you've seen used before? For example, some cars use speech recognition systems that have been trained to recognize the different commands you can give to the in-car computer. What other examples have you used?

## 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?

### Add new commands

Try adding two more training buckets for “up” and “down” so you can control the alien to move in all four directions.