

Student Guide for Machine Learning for Kids

Table of Contents

Step 1: Set up a machine learning project

A. Start a new project

B. Return to an existing project

Step 2: Create classes for your project

Step 3: Collect examples of what you want the computer to recognize

Step 4: Train the computer with your examples

Step 5: Test the machine learning model

Step 6: Make something with your ML model

Step 1: Set up a machine learning project

A. Create a new machine learning project

1. Make sure you are logged in on [Machine Learning for Kids](#).
2. From the **Projects** page, click **+ Add a new project**
3. On the new project page, complete the following:
 - a. **Project Name:** Type the name of your project
Example: Rock, Paper, Scissors
 - b. **Project Type:** Select what you want to teach the computer to do
Example: Select **recognising images** for Rock, Paper, Scissors
 - 👉 Check your project tutorial for the **Project Type**.
 - 👉 Add your values at this point if this is a **recognising numbers** project!
 - c. **Storage:** Select **In the cloud**
 - d. Click the **Create** button

The screenshot shows the 'Machine Learning for Kids' website interface. At the top, there is a navigation bar with links for 'About', 'Projects', 'Worksheets', 'Pretrained', 'Help', 'Log Out', and a language selector. Below the navigation bar, the main title 'Teach a computer to play a game' is displayed in large, bold, black font. A blue button labeled 'Go to your Projects' is visible. The main content area contains two numbered steps: '1 Collect examples of things you want to be able to recognise' and '2 Use the examples to train a computer to be able to recognise them'. Step 1 has a small icon of a hand holding a camera.

1 Collect examples of things you want to be able to recognise

2 Use the examples to train a computer to be able to recognise them

B. Return to an existing project

1. Make sure you are logged in on [Machine Learning for Kids](#).
2. On the **Projects** page: Under **Your machine learning projects**, click on your project.
3. From your project page, click on **Learn & Test**.
4. Confirm that the model you trained before is available.
 - a. If the Current model status is **Available**, [skip to Step 6](#).
 - b. If it does NOT say Available, go to [Step 4's note](#), then move on to [Step 6](#).

Model is available:

Info from training computer:

Model started training at: Friday, October 10, 2025 11:22 AM
Current model status: Available

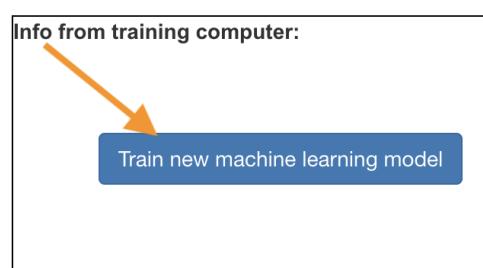
[Delete this model](#)

[Train new machine learning model](#)

Model needs to be trained:

Info from training computer:

[Train new machine learning model](#)



Step 2: Create classes for your project

One common type of machine learning (ML for short) is called **classification**. In classification, training examples (**data**) are organized into **classes**.

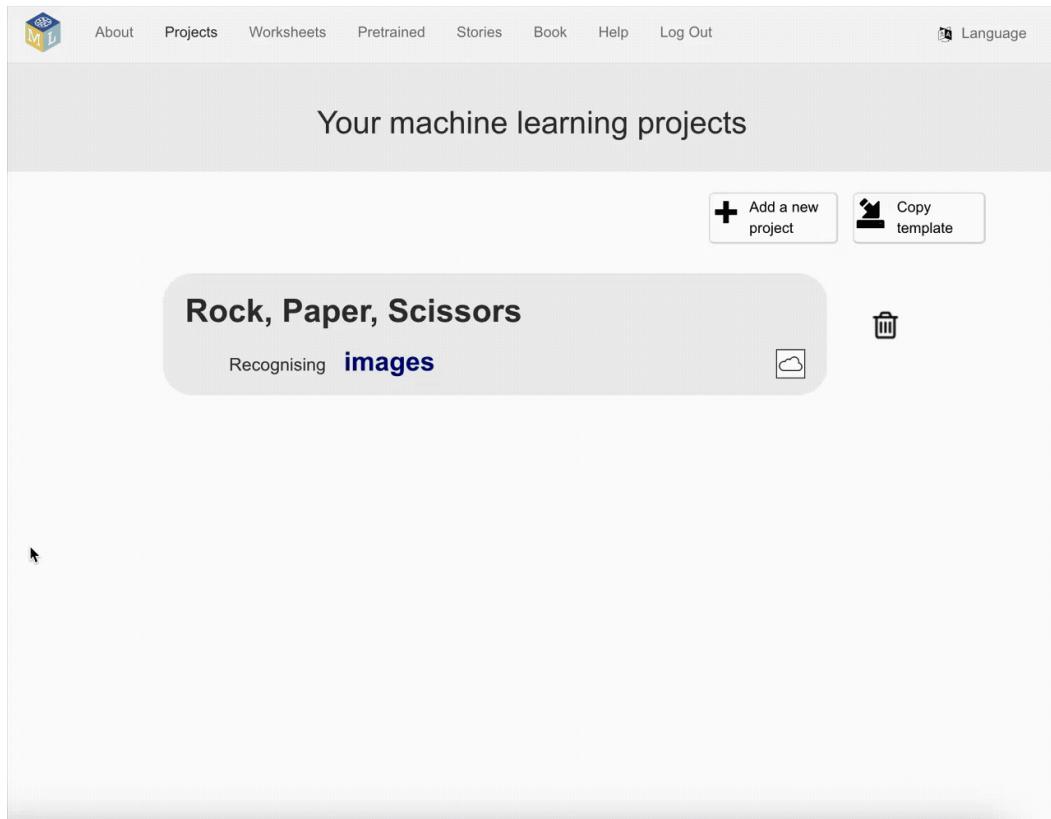
A class (or label) is a specific output that a ML model is trained to recognize, like “smile”, “cat”, or “pizza”. When given a new example, the model uses the patterns learned from training to predict a label for it.

Let's say you have two classes called Smile and Frown, and you train your model on lots of image examples of both. Now, when the model receives a new image, it will analyze it and then label it as either Smile or Frown.

Set up the classes that your examples will go into:

1. From your **Projects** page, click on the project you are working on.
2. Click the **Train** button.
3. Click the **+ Add new label button** to add a new training bucket or **class**.

 Check your project tutorial for the **classes** needed. Add each one.



Step 3: Collect examples of what you want the computer to recognize

Now we need to gather training data for the ML model. It's important to provide lots of *different* examples for this step.

Why? The model you're training doesn't know what the names you gave your classes mean. It examines all examples and identifies *patterns* that make each class unique.

For example, if all the images we provide for 'rock' include a hand that is very close up to the camera, the model may learn that anything that appears small or far away cannot be a 'rock'.

The ways you can give examples depend on the type of model you're training:

The image shows three separate windows for collecting examples:

- Add example (Text):** A text input field with placeholder "Enter an example of 'satisfied' *". Below it is a progress bar at 0 / 1000. Buttons for "ADD" and "CANCEL" are at the bottom.
- Add example (Image):** A window titled "rock" with a placeholder "Drag pictures from other browser windows and drop them here". Below is a "Record an example of 'up'" button with a microphone icon. At the bottom are "ADD" and "CANCEL" buttons.
- Add example (Sound):** A window with a placeholder "Record an example of 'up'" and a microphone icon. Below is a "Record" button with a microphone icon. At the bottom are "ADD" and "CANCEL" buttons.

Recognising text:
Type examples

Recognising images:
Paste links, take photos, or draw

Recognising sound:
Use a microphone

How to collect examples for your classes:

1. Plan out your training examples for this project. Make sure to have a *variety of examples*! If the training examples are all the same or too similar, your model will have trouble recognizing new examples that have small differences.
 - a. For example: Training a model to recognize cats in general (not one specific kind)? Include photos of different cat breeds, colors, sizes, and poses.
2. On the **Train** page, add at least 15 examples for each class in your project.
3. Keep the number of samples for each class the same: If you have 20 for one class, make sure to have 20 for the others for balance.
4. Stop and make sure you have a good variety of examples! Gather more if needed.

The model won't recognize new examples well if your training examples...

- **are too similar.** For example, if you only train your model on front-facing photos of a human face, it probably won't recognize photos taken from the side as human faces.

- **include too many tiny, unimportant details like specific backgrounds.** For example, if you want your model to recognize cats, and all your training examples have a blue background, the model will learn that blue backgrounds are a feature of a cat!

✓ Once you have a good variety of examples for each class, move to the next step.

Step 4: Train the computer with your examples

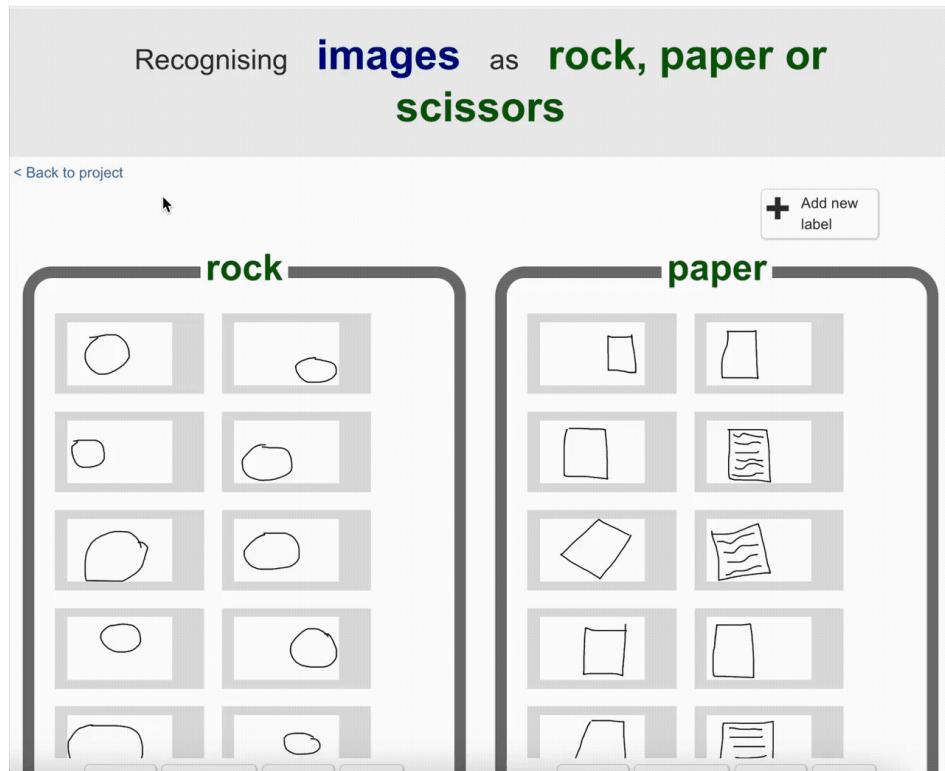
Now we can train the computer to recognize the different classes in your project. During this process, the computer evaluates every example to find patterns and relationships among them.

How to train your ML model:

1. From the Train page, click **Back to project** (top-left corner), then **Learn & Test**
2. Click **Train new machine learning model**
3. When the **Current model status** is Available, your model is ready for use!

! Note:

If you've trained a model for this project before, you should still go to the **Learn & Test** page to make sure it's available. Some model types are automatically deleted after a certain amount of time. If your model was deleted, follow "How to train your ML model" above to train it again.



Step 5: Test the machine learning model

Now let's make sure the ML model correctly classifies new examples.

Test your model:

1. On the **Learn & Test** page, provide new examples in the testing box.
2. Give testing examples for every class in your project.
3. You will see the class the model predicts this new example belongs to.
4. You will also see a **confidence** score out of 100%, which shows how strongly the model's prediction matches the patterns it learned.
 - a. The higher the confidence score, the stronger the pattern match. Aim for a confidence score of **at least 70%**.
 - b. If the confidence score is low, the model may not be finding strong pattern matches with the test example.
5. If you notice that the model is consistently incorrectly labeling your test examples, go back and repeat **Steps 3, 4, and 5**.
6. Once you are satisfied that the model can **correctly** and **confidently** classify new examples most of the time, move on to the next step.

[Train](#) page and collect some more examples

Once you've done that, click on the button below to train a new machine learning model and see what difference the extra examples will make!

Try putting in an image to see how it is recognised based on your training.

 [Test with webcam](#)

 [Test by drawing](#)

[Test with a web address for an image on the Internet](#)

[Test with www](#)

Recognised as **paper**

with 100% confidence

Info from training computer:

Model started training at: Wednesday, September 3, 2025 4:10 PM

Current model status: Available

Step 6: Make something with your ML model

It's time to use your ML model!

To connect your model with Scratch:

1. From the Learn & Test page, click **Back to project** (top-left corner), then **Make**
2. On the next page, click **Scratch 3**.
3. On the next page, click **Open in Scratch 3**
4. Begin coding:
 - a. Starting a new Scratch project?
 - In the purple navigation bar up top: Type a name for your project
 - b. In the middle of a Scratch project?
 - From purple navigation bar up top:
 - **File → Load from your computer**
 - Your teacher will help you open the .sb3 file that has your code



Careful, this version of Scratch does NOT automatically save as you work!

- ★ When you finish your coding session, keep the page open.
- ★ Your teacher will help you save the Scratch file for next time.