

Teacher Info

TRY IT FOR:



Smile to Drive is a coding activity that combines learning about **AI with coding** the intelino train using Scratch. Students will control the intelino train - with their face! This activity can be used for the Hour of Code™, since it is as interesting for beginning coders as for advanced ones. However, neither kids nor teachers need to know anything about AI, Scratch, or the intelino smart train!

The activity is designed to be **self-guided** and works for grades **1-8**, although younger kids may need some help with Scratch and with reading the instructions. Students will learn how to use the **AI Scratch extension "Face Sensing"**, developed by the Scratch team, to make the computer recognize facial expressions. Then, they will code in **intelino Scratch** to control the train.

The main activity is on pages 1-7 and should take one session. Page 8 has an optional reflection and more information about AI that the class can go through to reflect on what they learned. There is an optional Challenge on page 9, which can be done in another session. It's a bit more advanced, but builds on the skills that students have acquired in the first session.

The Face Sensing extension is directly available in intelino Scratch. Please check [here](#) to learn more about intelino Scratch and its requirements. Note that students start coding with a **sample program** that can be downloaded to their device and then opened in Scratch, or simply assembled from the blocks during the activity. (download the program [here](#))

If you are new to teaching with intelino, take a look at our [Teacher's Quick Start Guide](#). We also have more lessons and activities (unplugged or using Scratch or Python) in the [intelino lab!](#)

Group

up to 4 students per group

Code Modes

mode 3 (intelino Scratch)

Prerequisites

none - no knowledge of coding or intelino is required

Grades

1-8

Time

1-2 sessions

Standards

- CSTA: 1A-AP-8, 1A-AP-10, 1A-AP-14, 1B-AP-10, 1B-AP-12, 1B-AP-15, 1B-IC-18
- Common Core: CCSS.MATH.PRACTICE.MP1, CCSS.MATH.PRACTICE.MP3
- ISTE: 1.1.c, 1.7.b, 1.7.c

Supplies

- per group:
- 1 intelino engine, charged
 - device
 - printout of pages 1-9
 - scissor

Questions?

[email julia@intelino.com](mailto:julia@intelino.com)



Ages: 6+
Difficulty: beginner
Code Modes: mode 3 - intelino Scratch
 1-2 sessions

TRY IT FOR:



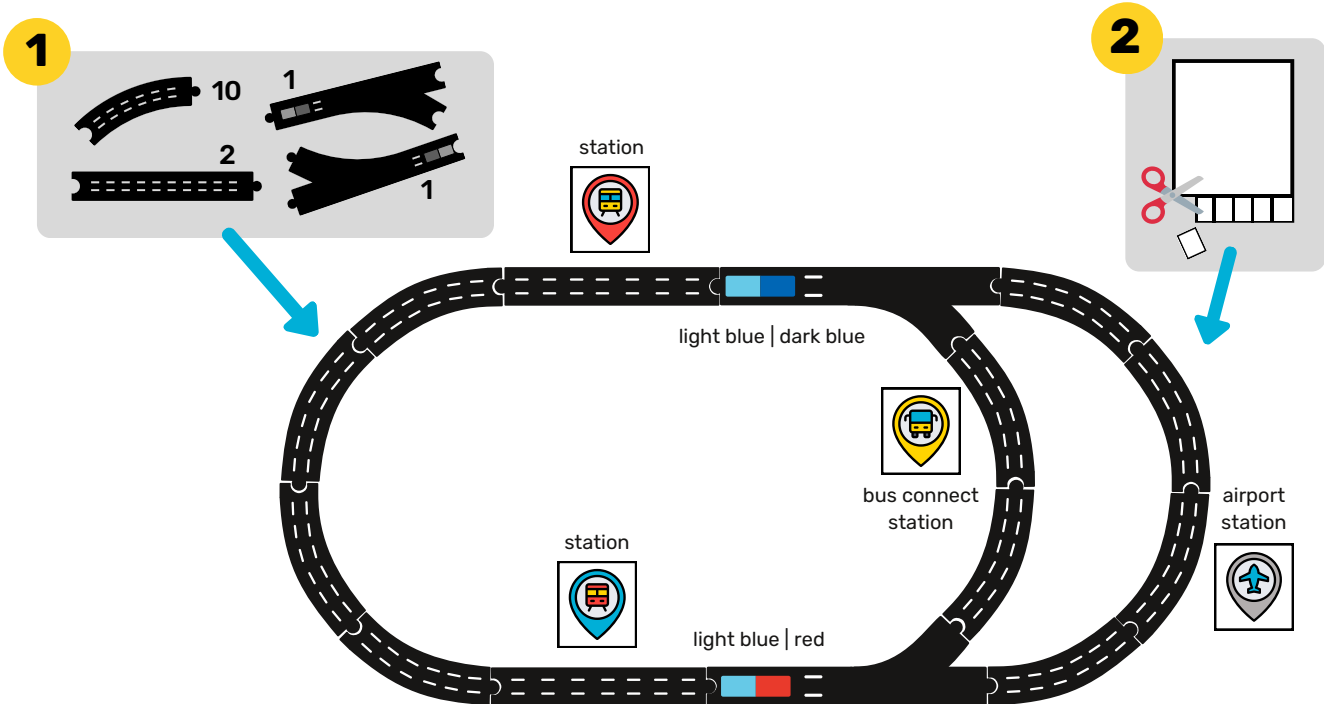
intelino®

NAME _____



Hi, I'm Jamy! I'm an intelino smart engine.

I'm an autonomous train, which means I don't have an engineer that drives me. You will be controlling me today. How you ask? With your face! Let's see how that works!



GOAL Make Jamy drive and stop at every station!

HOW? Make a program that recognizes facial expressions and makes Jamy drive and stop depending on the expression. Then control Jamy to stop at every station.

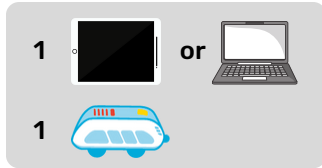




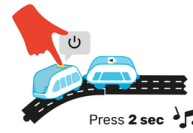
Now:

Let's start intelino Scratch and connect it to the engine.

1 Get engine and device



2 Power on engine



3 Start intelino Scratch

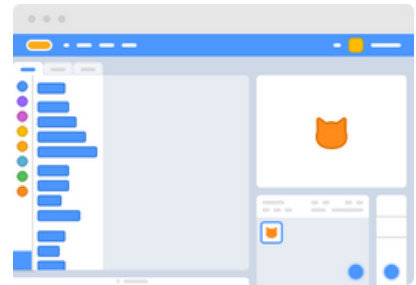


Open in web browser:
scratch.intelino.com

or

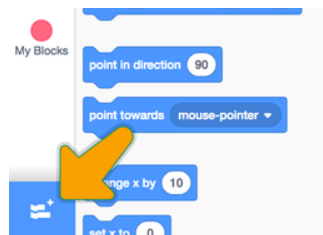


Launch intelino Scratch app

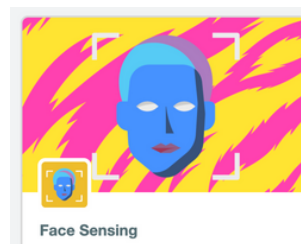


4 Open Scratch Extensions

Click on the blue button in the lower left corner

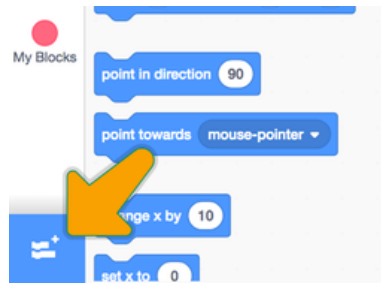


5 Select Face Sensing



6 Open Scratch Extensions again

We have to add another one!

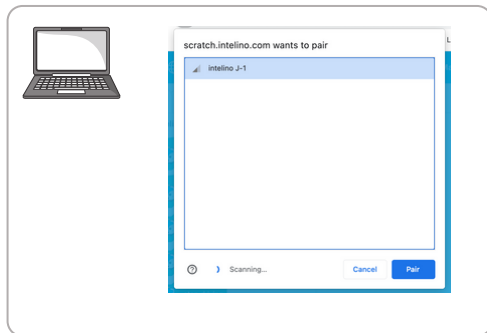


7 Select intelino smart train 1

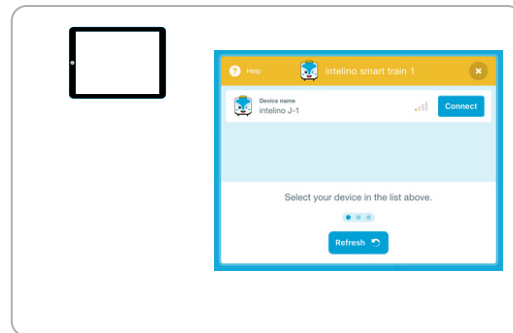


8 Connect engine

Find your engine and [Pair](#) or [Connect](#)

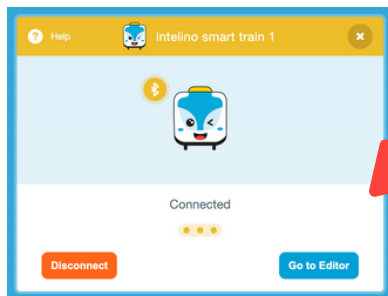


or

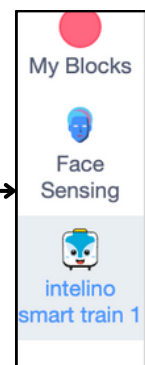


9 Success!

Now go to the Editor



editor

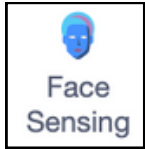


You can see the two new extensions in the left side bar



Now let's look at our task:

How can we control the train with our face?



Face Sensing

It's a program that detects when you are doing gestures with your face like smiling or blinking.

When face gestures are detected,
then control the train with scratch blocks



intelino Scratch blocks

Commands that we can use to make control the train (make it do certain actions).

Example

When face sensing detects a smile



then make the train drive.



Code Talk:
The **When** is call an event, and
the **Then** is the response!



How does this look like in code?

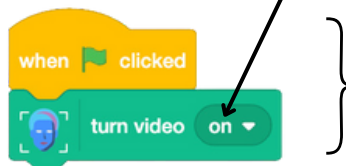
Let's see!

▶ Open Program

Use **File -> Load from your computer** and open the file "smile-to-drive.sb3" **or assemble it** from the blocks so it looks like this program below.

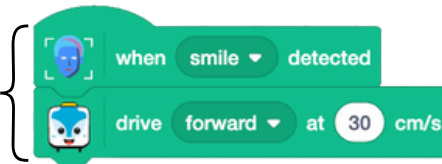
Study the blocks!

You can change the options for many of the blocks if needed

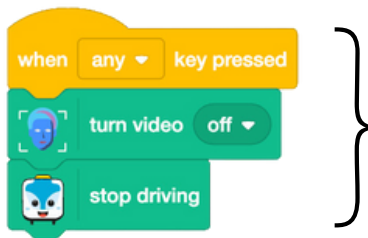


These blocks just start the program and turn on the camera

This is where the things from our example happen!



The camera and train do not automatically turn off when we're done running our program so these blocks make it easy: just press any key on the keyboard!





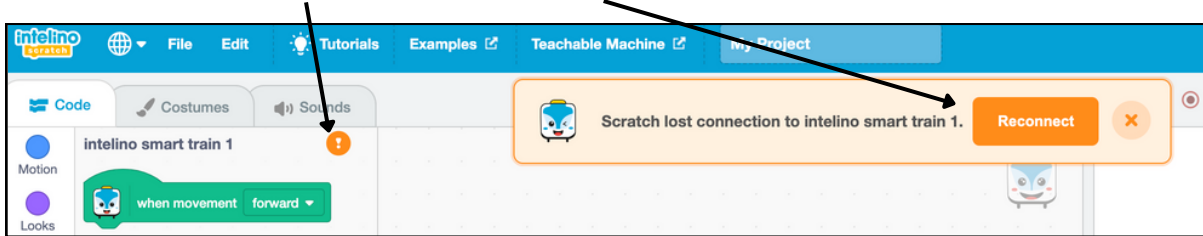
Ok, let's run the program!

These are the steps:

▶ Make sure that the train is still connected

After a while, the train may turn off or disconnect to save energy.

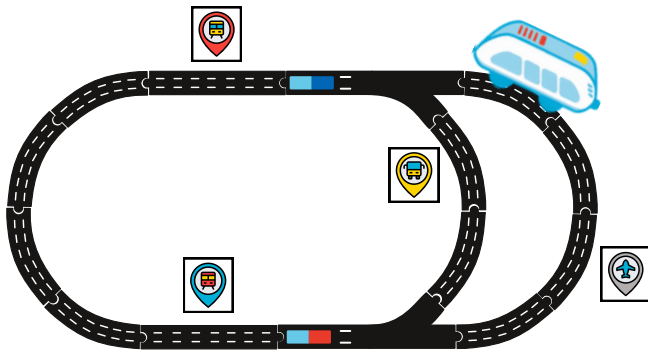
Check if you see this orange circle or this box.



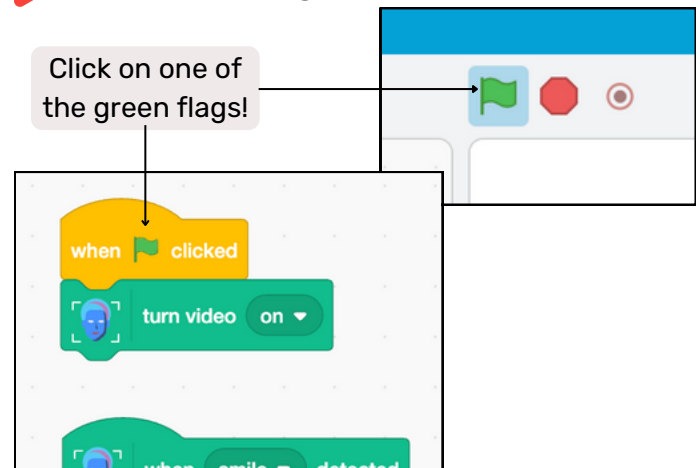
If so, do this:

- make sure the engine is turned on, and
- click on either the orange circle or the Reconnect button and connect again like before.

▶ Place the engine anywhere on the track

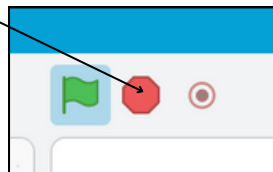


▶ Run the program!



Tip: how to stop a program

Click on the red flag



AND

Press any key to stop the engine and camera, or just pick up the engine and quick press the button.



How about stopping at stations?

Now it's your turn to program.

▶ Program Jamy so that it can stop at every station it comes across



Add to your program

Many things don't have to change, just add the blocks you need.



To duplicate a set of blocks:

Right-click (on a computer) or long-press (on a tablet)

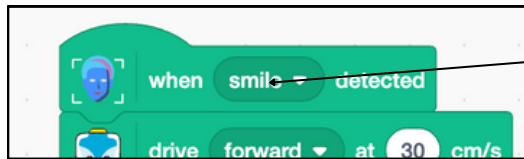


Try different face gestures

You want to use a different gesture for each train action. Pick gestures that are not similar. We found that "smile", "whistling" or "mouth open" work well in the same program (if you smile and whistle without opening your mouth).



Click on the blocks to change the options



Click on these parts of the blocks to change the options



Divide the work!

It is really hard to watch the train AND do the expressions at the same time. Have one person watch the train and shout out when to stop. Another person can do the facial expression.



Debug: change and try again

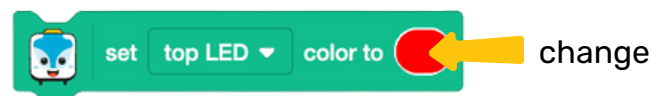
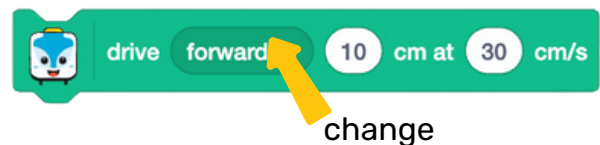
If something doesn't work:

- stop running the program
- fix the commands (blocks)
- try again

This is called "debugging"

▶ If you're done, try the following

- add another facial expressions for driving backwards. You will need to change this block to drive backwards:
- change the LED light color when Jamy drives forwards or backwards. Add this block to your code:



A Look Inside



How does this all work?

Face Sensing is something we call Artificial Intelligence, or AI for short. But what is all of this and how was it made?

What is Face Sensing?

It's a special computer program that we call a model. It's really good at recognizing facial expressions. Your computer uses it as a smart helper for this task.

But aren't computers bad at things like recognizing faces and emotions?

True, computers are bad at some things that humans can do much better, and reading facial expressions or emotions is one of those. But the Face Sensing model learned how to do it and is now really good at it:

- The model looked at a lot of facial expressions and learned to recognize and tell them apart. This is called "Training the Model".
- It became smart and that's why we call this process and model Artificial (not human) Intelligence, or AI for short.

Why don't we just use humans for some of these tasks?

AI models can be our assistants for some task that would take us too long. AI models have become so good that they are sometimes much faster and better. And one model can be used in many places at the same time - all over the world! Just as your class is using the face sensing model, other classrooms may be using it at the same time right now.

In real life, AI models have made so many tasks much easier for us. For example, models that recognize faces are used to keep us safe. They can make sure people are who they say they are.

Is AI scary?

New things can be scary if we don't know what they can and cannot do and if we don't know how they work. This is why we started to learn about AI today!

With knowledge and rules about how to use AI, we can make sure that it is safe for everyone!

Challenge **Take a Turn**

Difficulty ●●○



Here's a challenge for you:

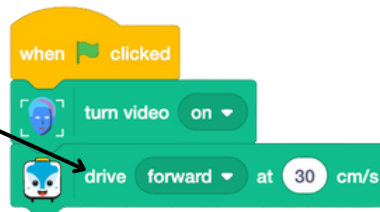
You may have noticed that I turn or go straight randomly. Add face sensing to control my steering! This way, you can make sure that I stop at every station.

▶ Program Jamy so that

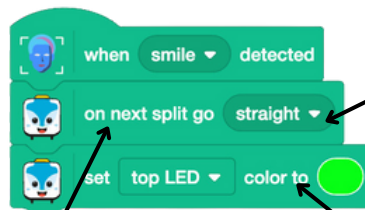
- Jamy drives at a speed of 30 cm/s.
- Use Face Sensing to control Jamy at the next split track (go straight or turn).
- Use Face Sensing to make Jamy pause at a station for 2 seconds.
- When Jamy is driving, control the train so that it stops at every station - you will have to get to and pause at every station. Don't skip any!

You can use this part of the program to get started:

We can start driving right away. This means that we just have to pause driving at the stations. Which block makes Jamy pause?

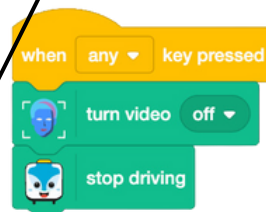


This makes Jamy go straight. But make sure you are choosing the right direction for the turn - it could be left or right depending on the way you let Jamy drive!



Note: if you happen to give more than one direction command before Jamy reaches the split, then they will overwrite each other.

What does that mean? Imagine, for example, that you tell Jamy to go straight, but then give a turn command before Jamy reaches the split. Then Jamy will drive according to the last command you have given. In this case, Jamy would turn.



Tip: setting a different colors for different steering directions makes it easier to see which one is set for the next split.

Solutions

page 7



when clicked

- turn video on

when smile detected

- drive forward at 30 cm/s

when whistling detected

- stop driving

when any key pressed

- turn video off
- stop driving

page 7



when clicked

- turn video on

when smile detected

- drive forward at 30 cm/s
- set top LED color to green

when whistling detected

- stop driving

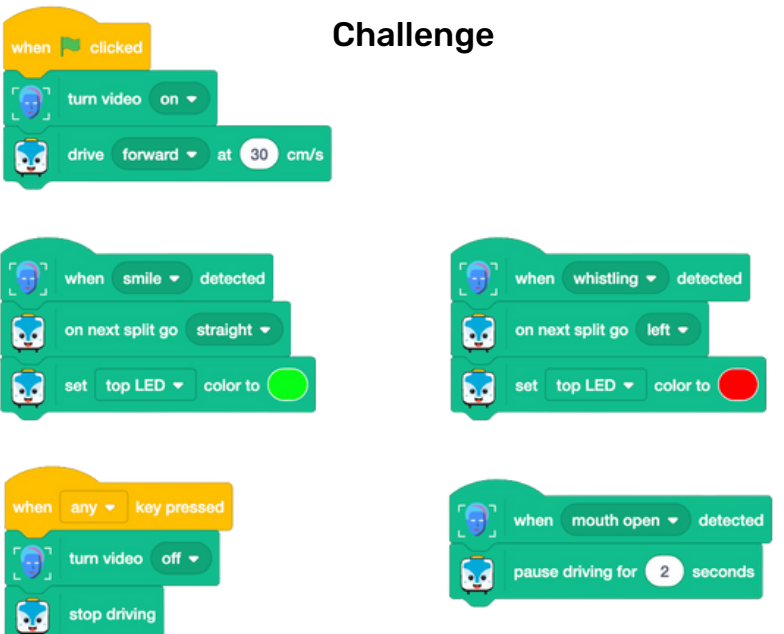
when any key pressed

- turn video off
- stop driving

when mouth open detected

- drive backward at 30 cm/s
- set top LED color to red

Challenge



when clicked

- turn video on
- drive forward at 30 cm/s

when smile detected

- on next split go straight
- set top LED color to green

when whistling detected

- on next split go left
- set top LED color to red

when any key pressed

- turn video off
- stop driving

when mouth open detected

- pause driving for 2 seconds