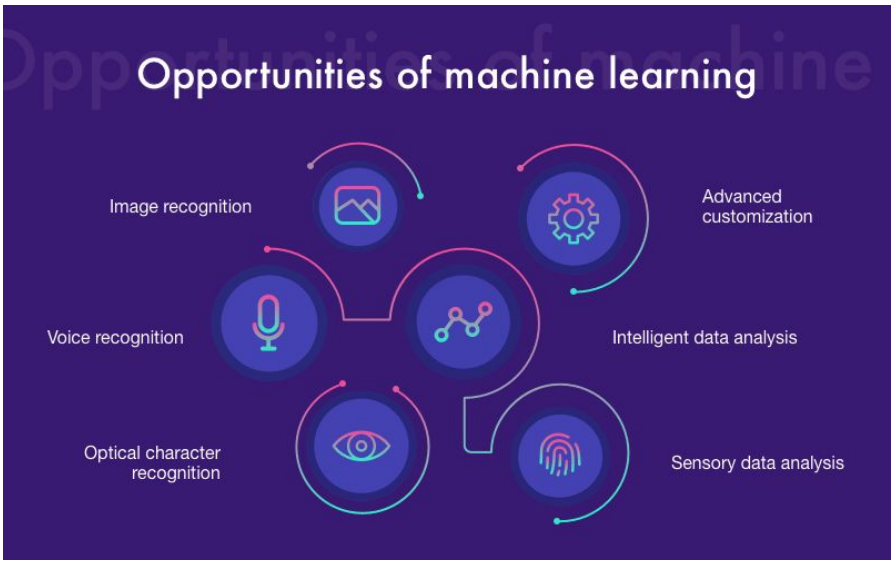
Image Classification: WhatIsIt

You're going to create an app that can classify objects in images using the concept of machine learning.

What is Machine Learning?

- It's a way for a machine to analyze data and output determinations or predictions based on the given data.
- It's basically a way to give a machine a brain of its own.
- With more experience (data), the brain (model) gets better at producing results.

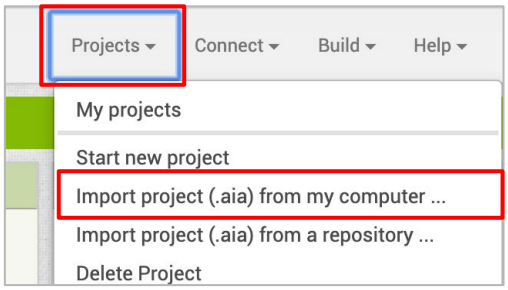


Courtesy: https://www.cleveroad.com/blog/importance-of-machine-learning-applications-in-various

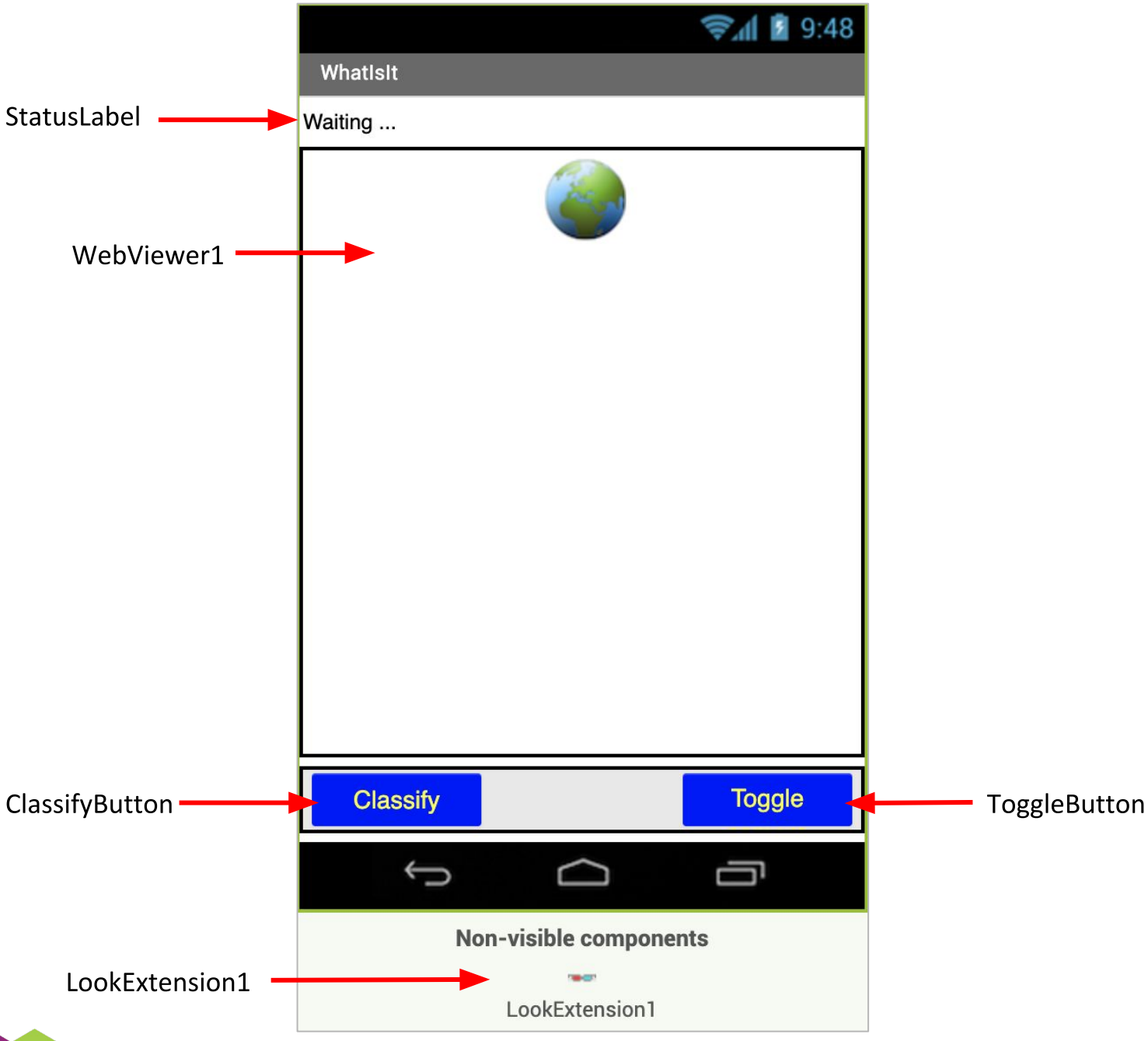
"Computers are able to see, hear and learn. Welcome to the future."
- Dave Waters

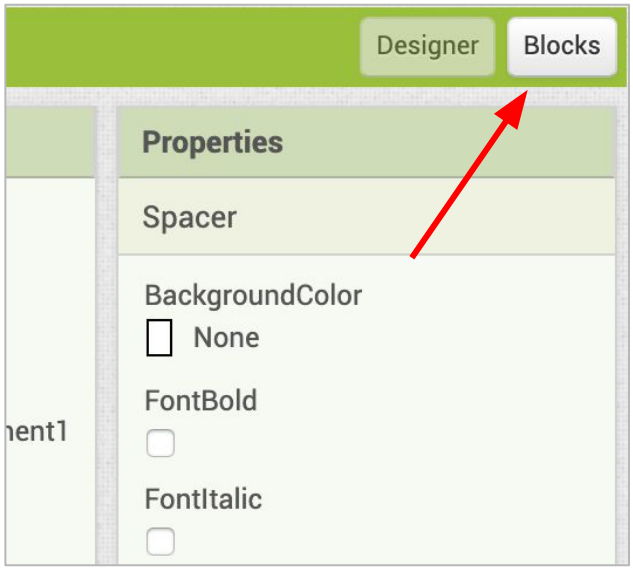
START HERE:

1. Download the WhatIsItTutorial.aia template file to your computer, then import the aia file into MIT App Inventor..



2. Look at the user interface components that are included in the template, and try and figure out what each component does.

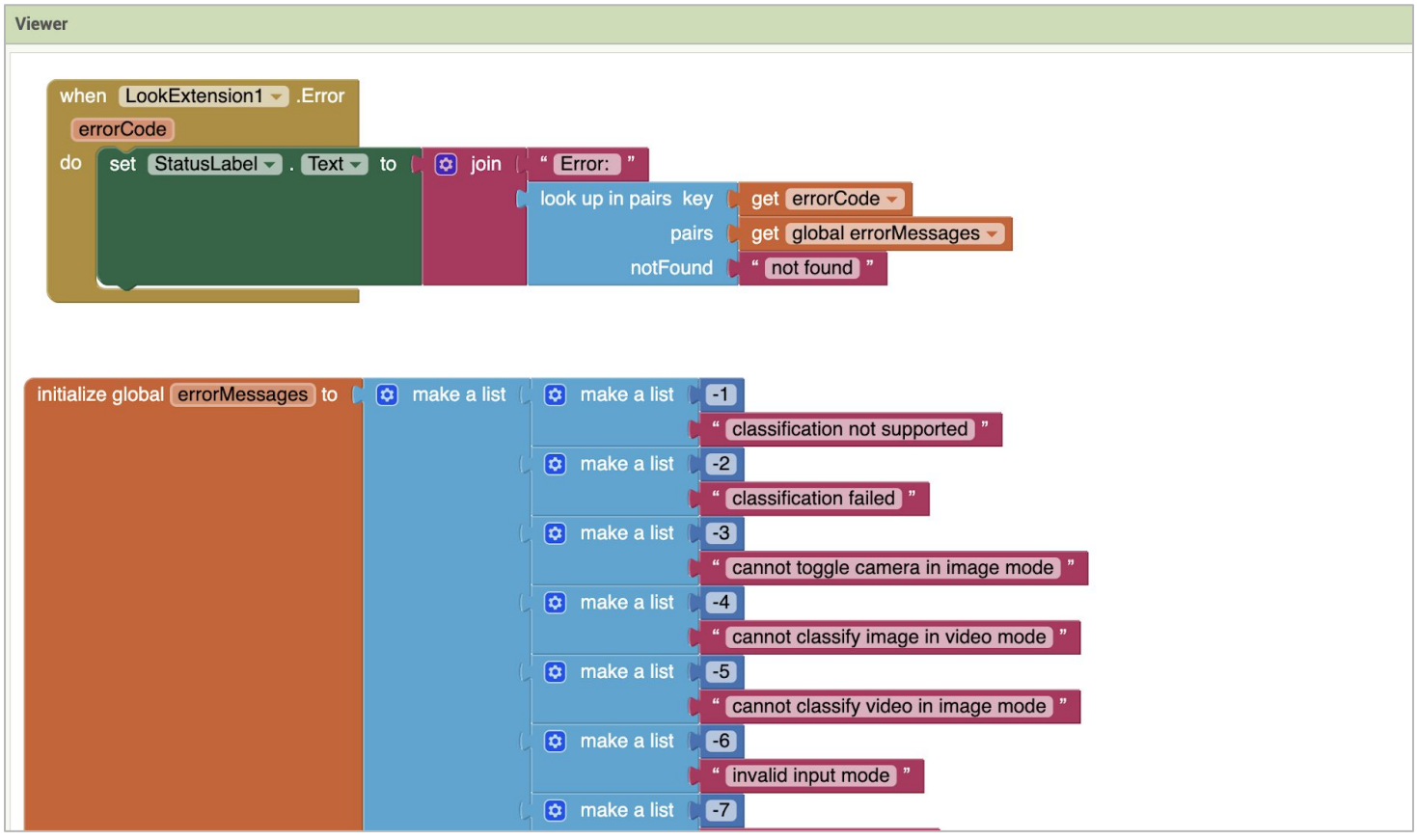




BLOCK EDITOR SPACE:

3

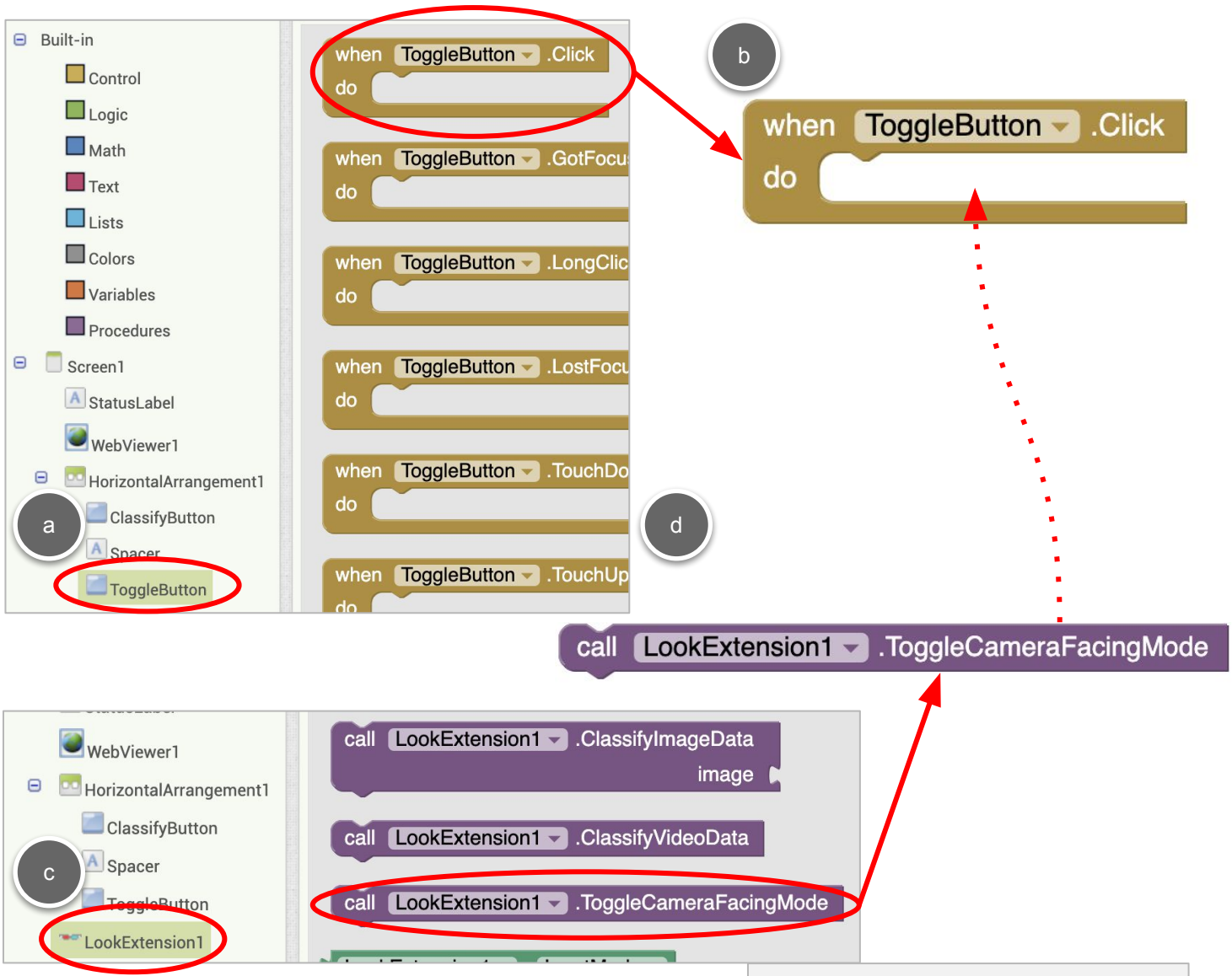
If you press the Blocks button on the right side of the top bar, you will see the Blocks Editor space. This is where you will code your app. There are some blocks there already, but we will discuss those later.



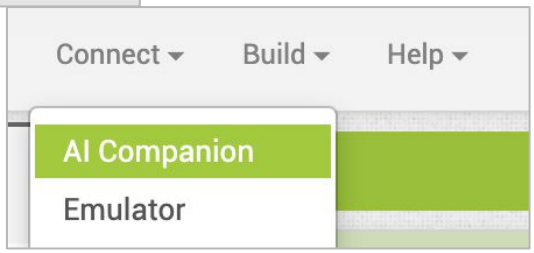
TOGGLE THE CAMERA!

First, you need to create a way to toggle which direction the camera is facing.

4 Pull out the **when ToggleButton.Click** block and add a **call LookExtension1.ToggleCameraFacingMode** block to it.



5 Test your app using MIT AI2 Companion. Your toggle button should now allow you to switch cameras!



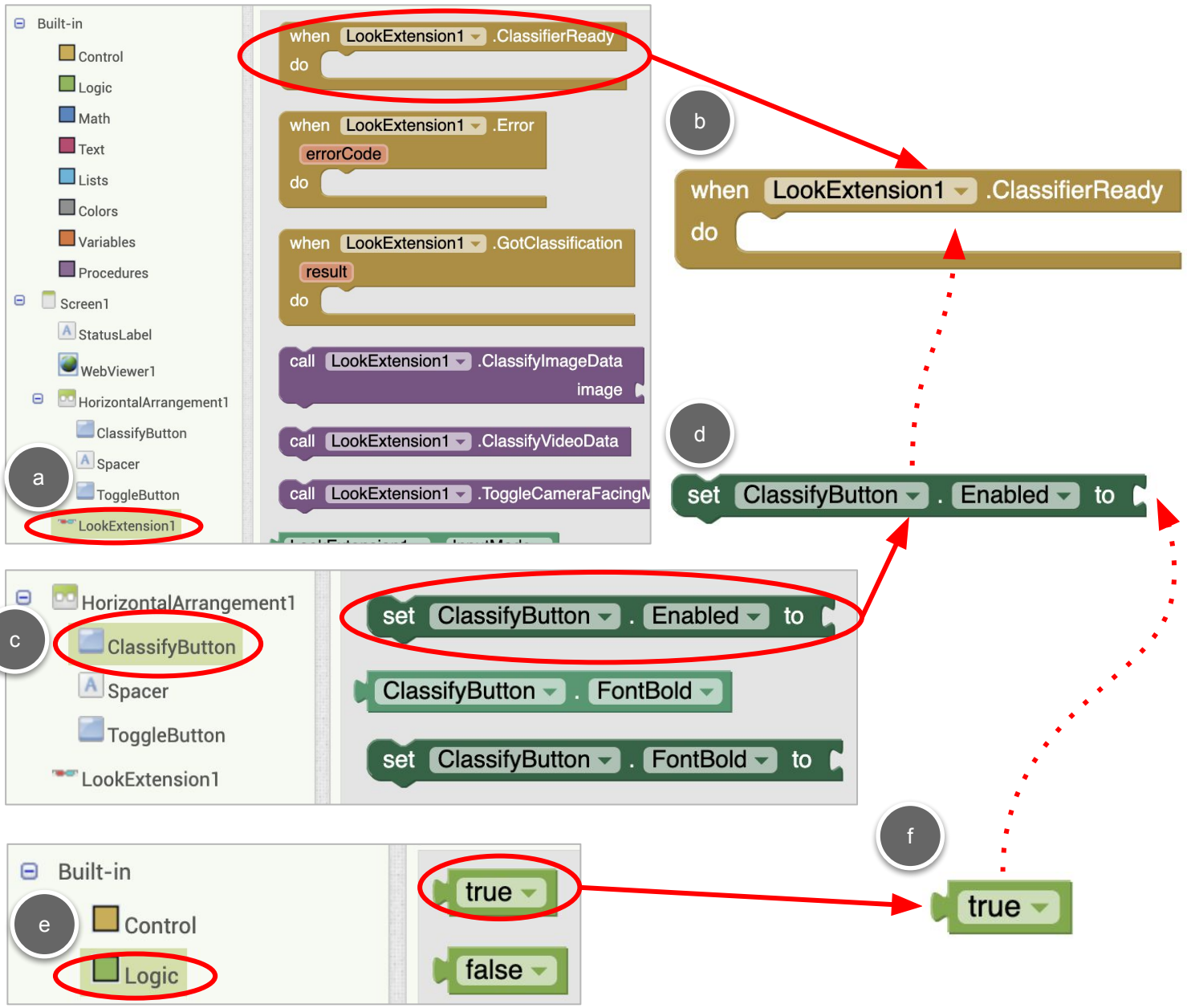
PREPARING LOOKEXTENSION:

Some components need to be prepared in order for your image classifier to work.

First, you need to tell your classify button that LookExtension is ready to go!

6 Pull out the **when LookExtension1.ClassifierReady** block.

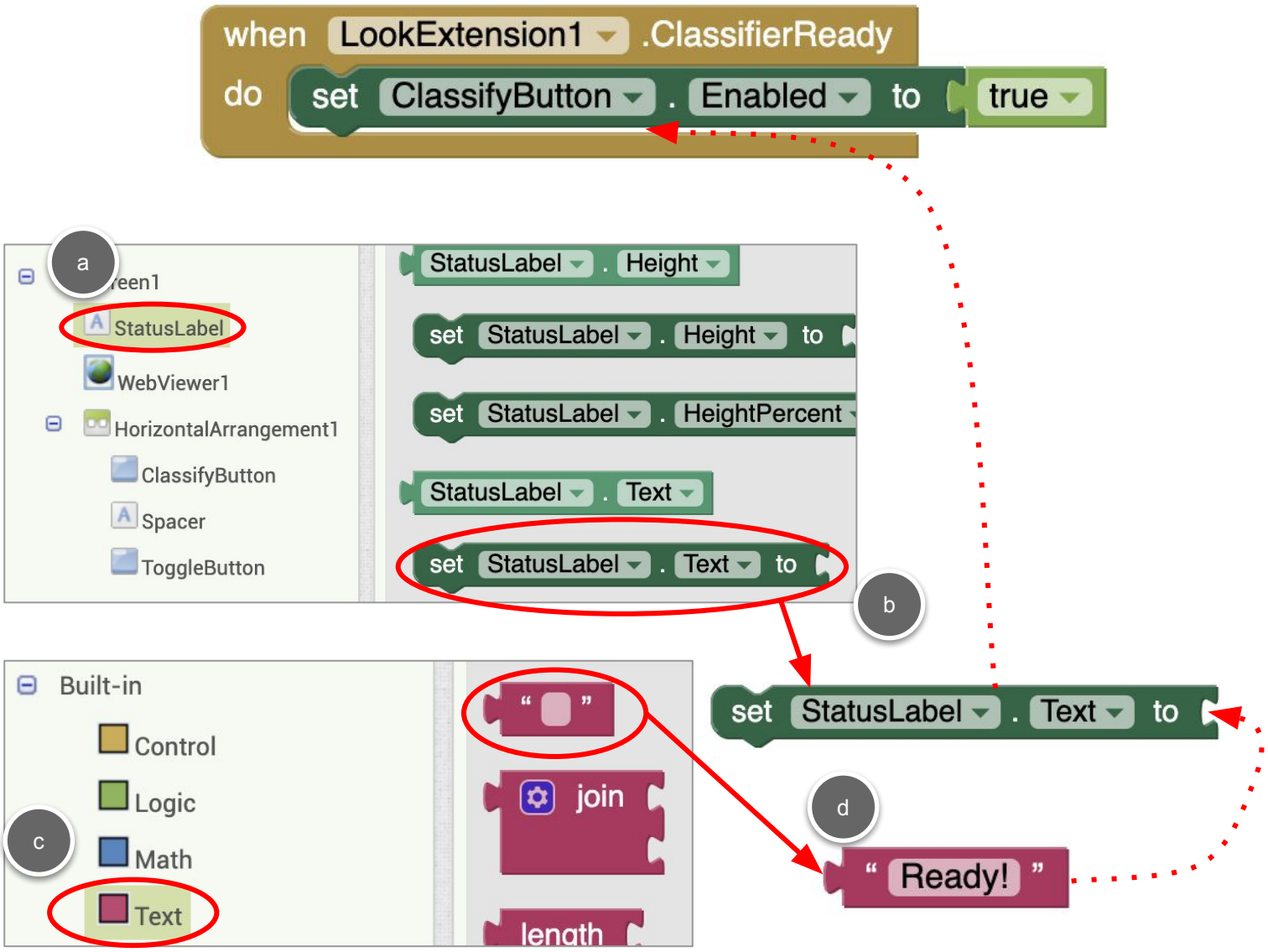
7 Add a **set ClassifyButton.Enabled** block along with a **true** block attached to it. The **true** block turns on the Classify button when LookExtension is ready.



PREPPING LOOKEXTENSION:

Second, you need to know when the Look extension is ready to classify.

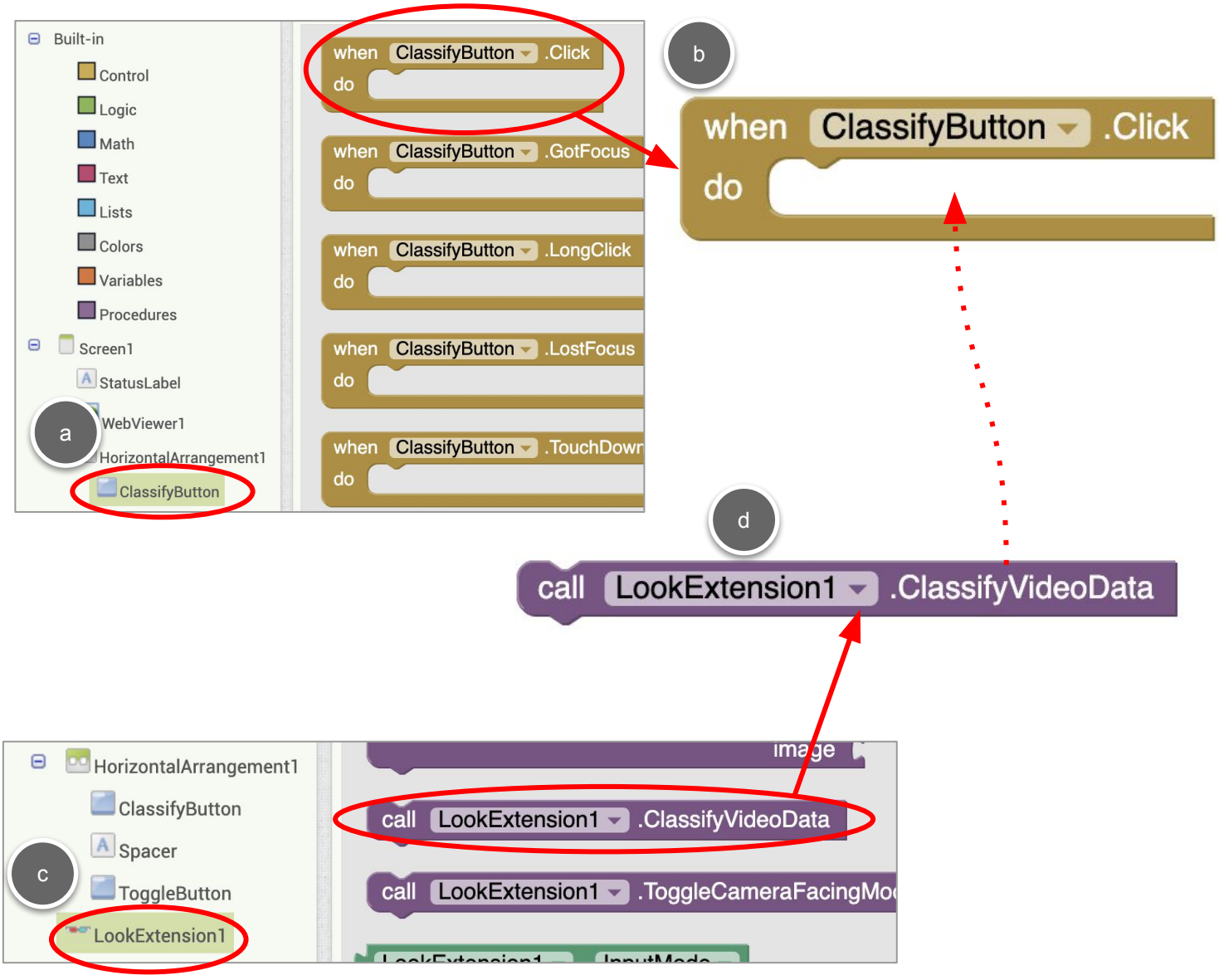
8 Pull out **set StatusLabel.Text** block and attach a blank text block to it. Add this block to the **LookExtension1.ClassifierReady** block, right underneath the **set ClassifyButton.Enabled** block. Change the blank text block to the text “Ready!”



CLASSIFY!

Now, you need to code the event when the user clicks on the Classify button!

9 Pull out the **ClassifyButton.Click** block and add a **call LookExtension1.ClassifyVideoData** block to it.

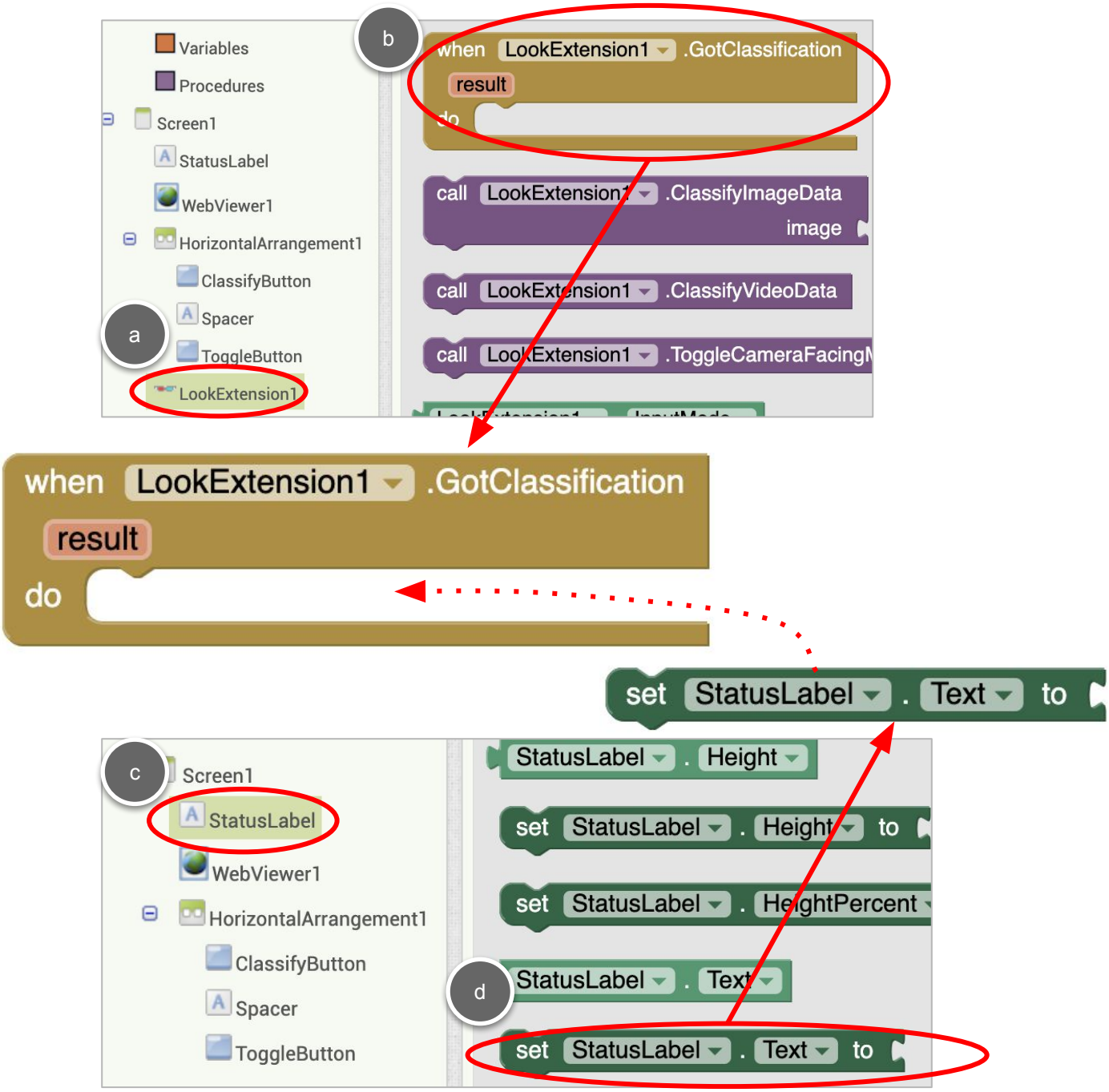


DISPLAYING THE CLASSIFICATION:

You got your classification system working, but where are the results?

Yup, you need to display them! You'll set your **StatusLabel** text to the classification result!

10 Pull out the **when LookExtension1.GotClassification** block and add a **set StatusLabel.Text** block to it.



DISPLAYING THE CLASSIFICATION:

11 You need to set your StatusLabel to the result now. You should remember that result is a list of lists of the top classifications in the form:
[[classification1, confidence1], [classification2, confidence2], …, [classification10, confidence10]]
where classification1 is the best classification of the object in the image.

So if you think about it, you only want the first item in the list, right? That will give you the first list, which includes the first classification and its confidence level.

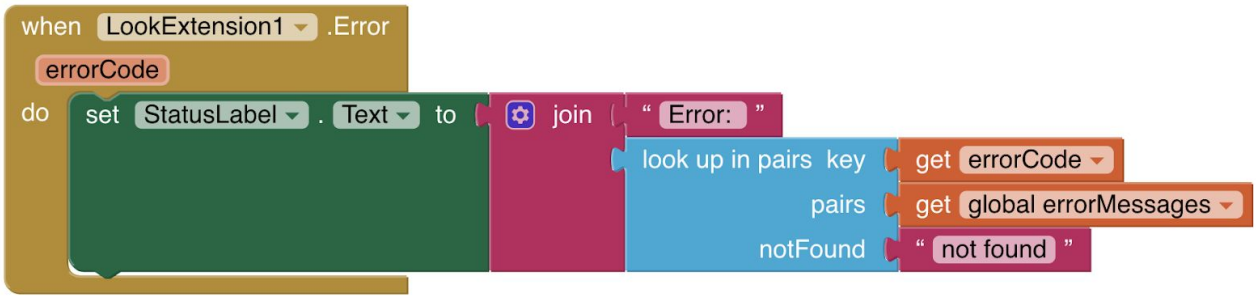
12 Pull the select list item block and attach a value of 1 to its index slot. Click on **result** in the **LookExtension1.GotClassification** block and attach this to the list slot of the **select list item** block. Snap this block to the **set StatusLabel.Text** block in the **LookExtension1.GotClassification** block.

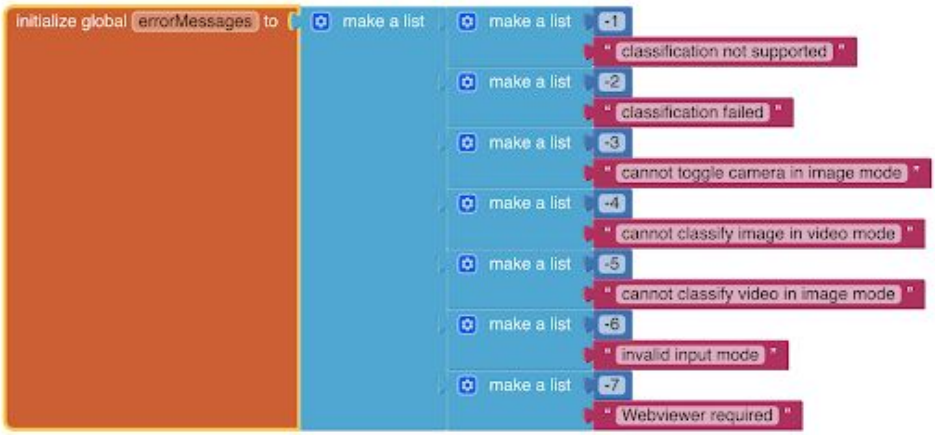
EXTRA BLOCKS:

13

The other two blocks already given to you handle any error messages the app may give you. When making modifications to the app, these may be useful in guiding you.

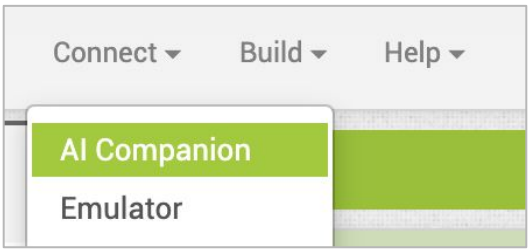
When the LookExtension encounters an error, it returns an error code made of numbers. These code blocks translate the error number into text that humans can understand.





TESTING THE APP:

13 Congratuations, you've finished making the app! Now it is time to test it out. Test your app using the MIT AI2 Companion. Your classification app should be up and running!



14 Test your app on classifying various objects in the room. What types of objects is it good at classifying?

14 Now test your app on the same object at different distances and angles. Does the classification stay consistent?

Keep Exploring!

Extend Your App

Here are a few features you could add if you wanted to expand your app

Make the device speak the classification (without the confidence value) rather than just showing it in the status label.

Show the top 3 classifications rather than just the best; or if the top two classifications are close enough, classify the object as either item1 or item2.

Create another button to toggle to a still-image mode. Modify the demo to classify images other than those from the video camera.

What other ideas do you have?