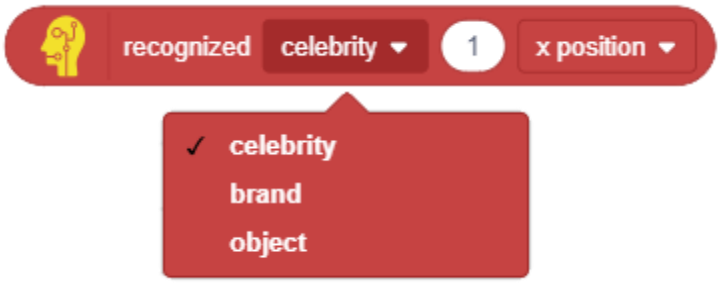
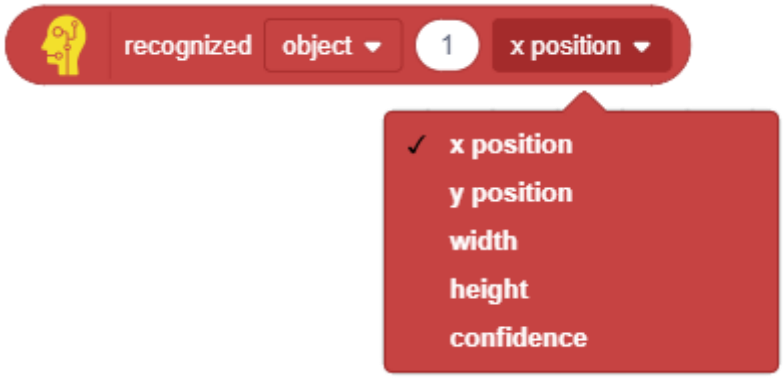
Activity 2: Locating Objects

In the previous activity, you made a project which identifies celebrities in the image. But how to locate them is what we will learn about this topic. Let's begin:

recognized () () () block



In feature recognition, you can get the location and other parameters of the recognized celebrity, brand, and object using the recognized () () () block. You can get the following parameters using this block:



1. x position: Reports the x position of the identified object.
2. y position: Reports the y position of the identified object.
3. width: Reports the width of the identified object.
4. height: Reports the height of the identified object.
5. confidence: Reports the confidence of the identified object. 0 is less likely and 1 is more likely.

Project

Let’s make a script that makes a bounding box on the identified object.

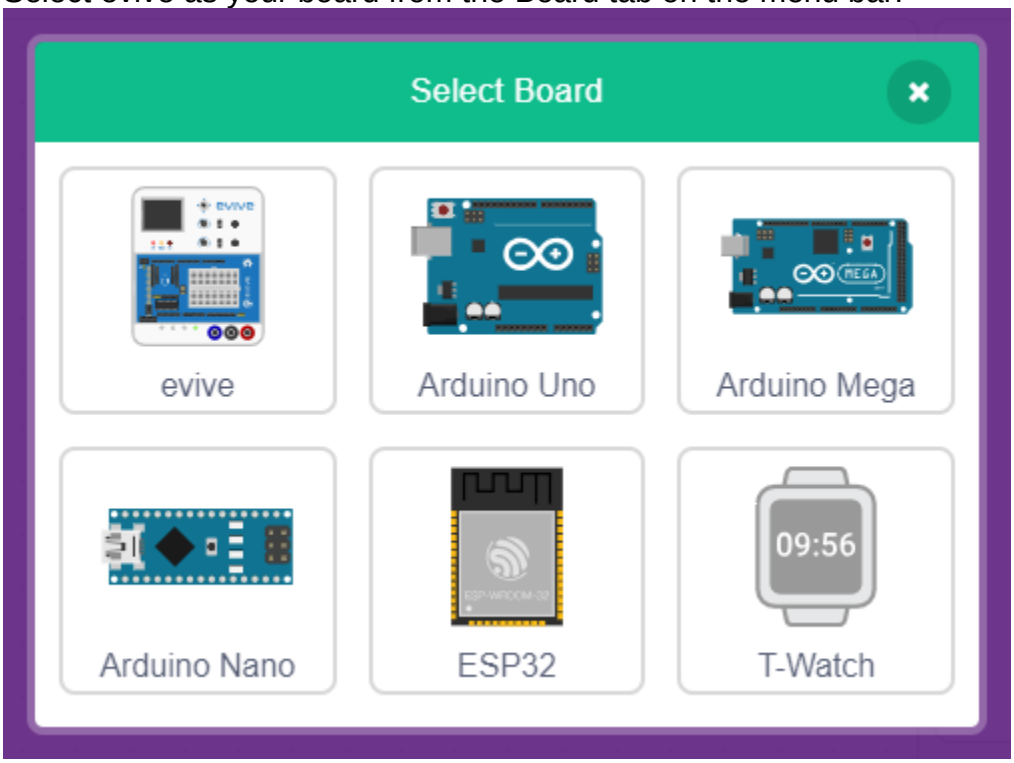
We’ll follow the following process:

1. Identifying the objects from the image.
2. Running the script in a loop for each object.
3. Making the bounding box for each object based on its X and Y location.

Let’s begin!

Setting Up the Stage

1. Open a new project in PictoBlox.
2. Select evive as your board from the Board tab on the menu bar.

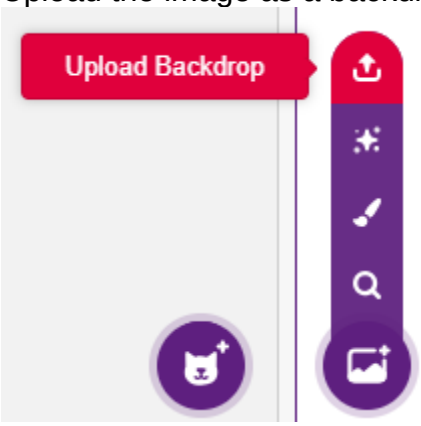


3. Click the Add Extension button in the bottom left corner.

4. A modal will open will all the available extensions. Select the *Artificial Intelligence* extension from the library.

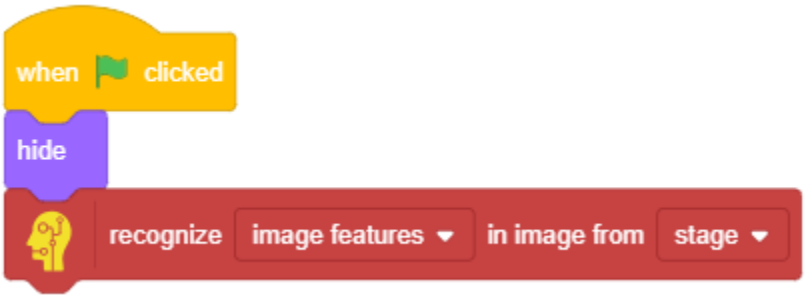
5. Download the image from here: https://learn.thestempedia.com/wp-content/uploads/2020/04/Kids-and-Bus.jpg

6. Upload the image as a backdrop.



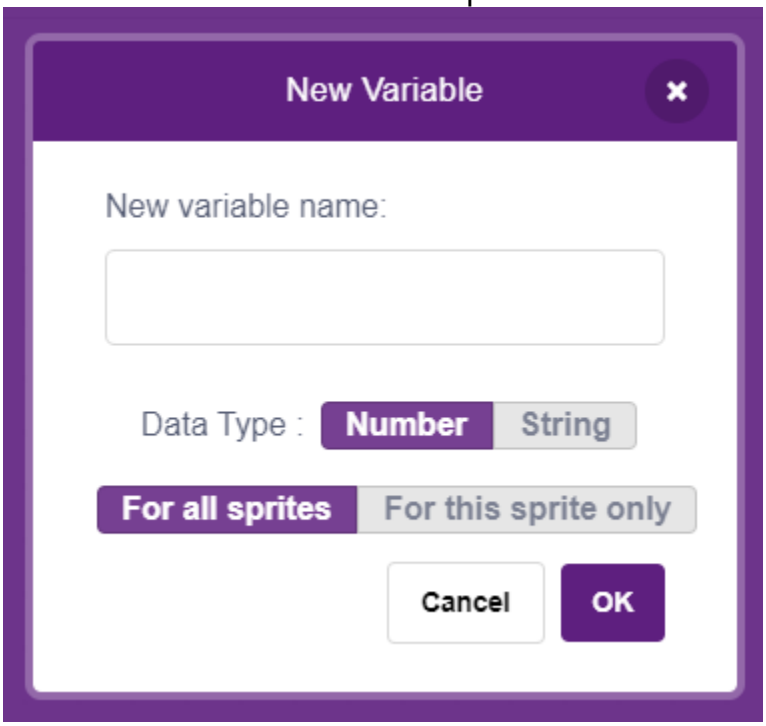
7. Add a new sprite named Box from the sprite library:

8. Delete Tobi, select the Box sprite, and switch to the Code tab.
9. Add a when flag clicked block in the scripting area.
10. Snap a hide block from the *Looks* palette.
11. Add a recognize () in image from () block and select image features and stage as inputs.

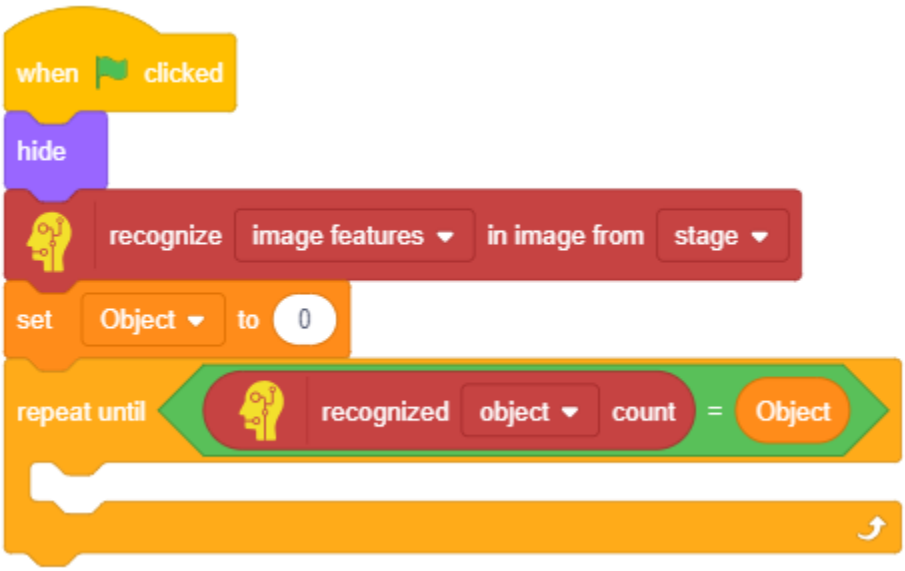


Locating Objects on the Stage

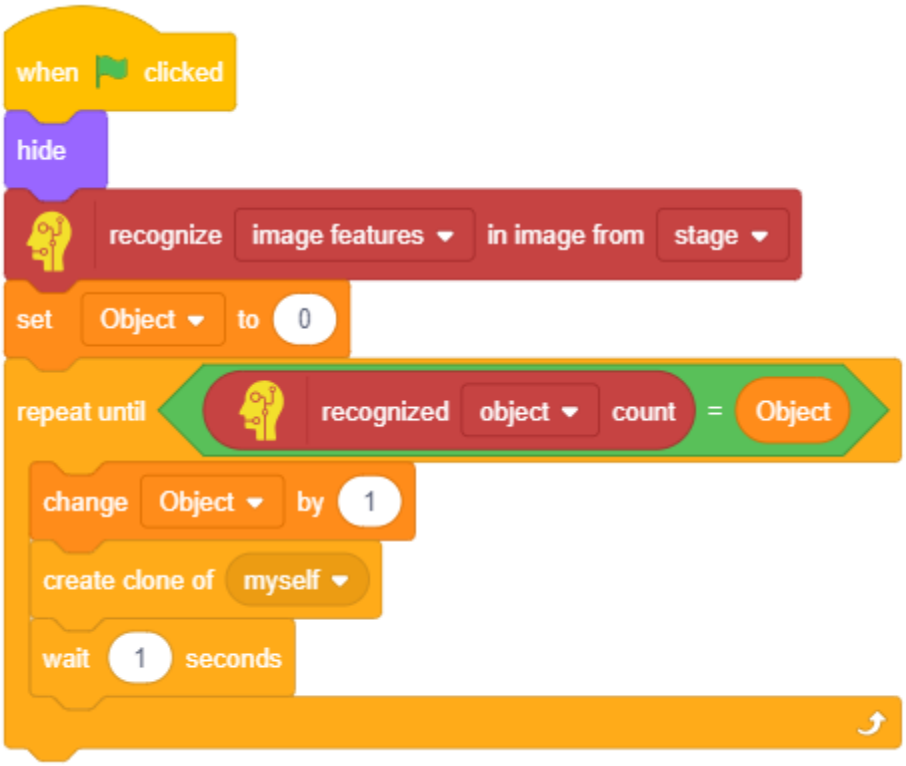
1. We'll continue with the same script. Make a variable named Object.



2. Add a set () to () block. Change the variable to Object and set the value as 0.
3. Add a repeat until () block. Drop an () = () block in the condition.
4. In the first input, add a recognized () count block and select object from the drop-down. In the second input, add the Object variable.



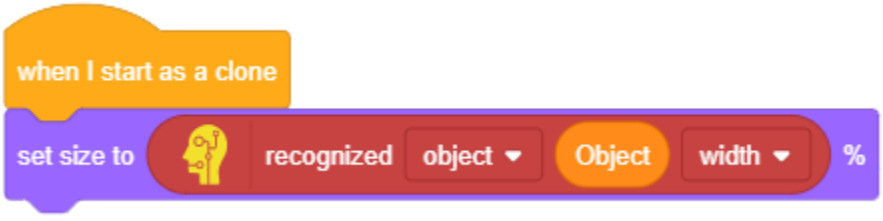
5. Inside the loop, add a change () by () block from the *Variables* palette. Change the variable to Object.
6. Snap a create clone of () block from the *Control* palette and select myself from the drop-down. Selecting myself means that you want to clone the same sprite you are writing the script for.
7. Add a wait () seconds block in the loop. With this, our main script is ready.



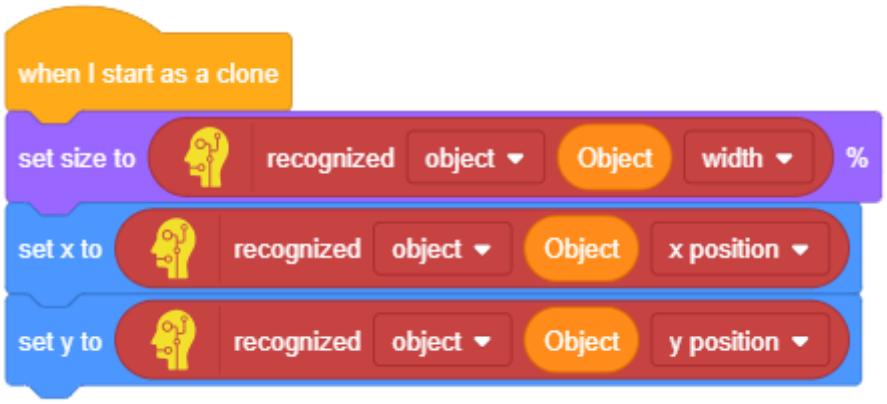
Drawing the Bounding Box

Follow the steps below for drawing the bounding box:

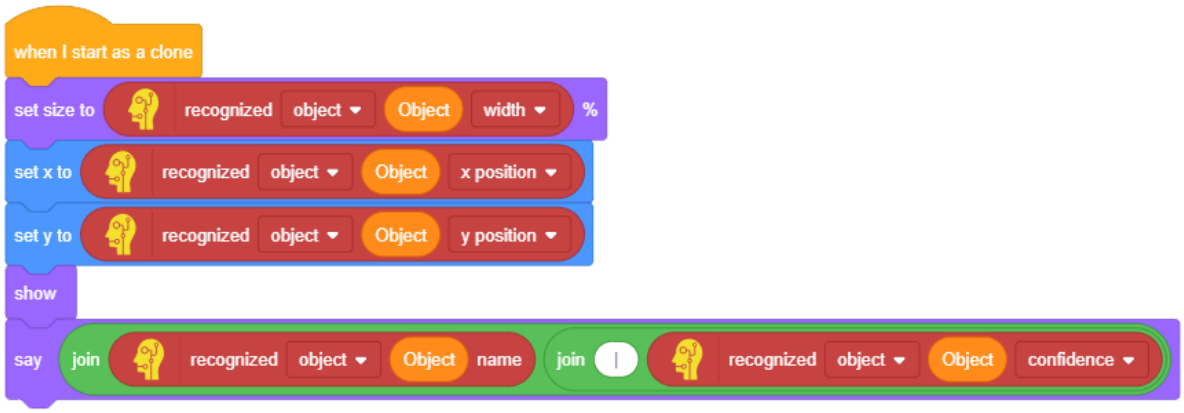
1. Add a when I start as a clone block into the scripting area from the *Control* palette.
2. Snap a set size to ()% block below the when I start as a clone block.
3. Add a recognized () () () block as the input of the set size to ()% block. Change the type to object and option to width. Next, add the Object variable as the input of the object number.



4. Next, add a set x to () block and a set y to () block and repeat step 3 for both the blocks as shown below:



5. Snap a show block from the *Looks* palette.
6. Add a say () block below the show block.
7. Add two join () () blocks.
8. Display the object name and object confidence using the say block as shown below.



Your project is complete! Click the green flag to run the script.

(END)