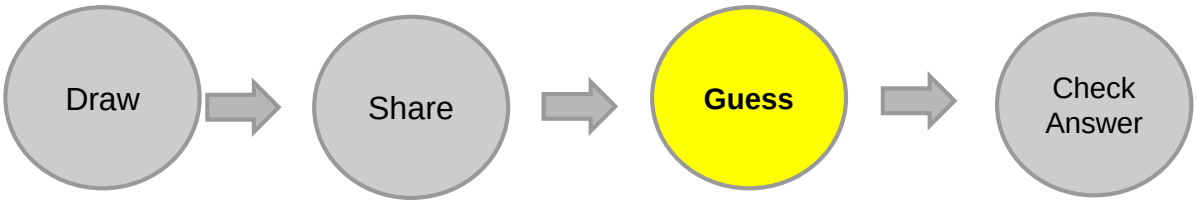
SKETCH AND GUESS

In this lesson, you will code the sharing of what the Sketcher is drawing, so the guesser can check whether they are correct.

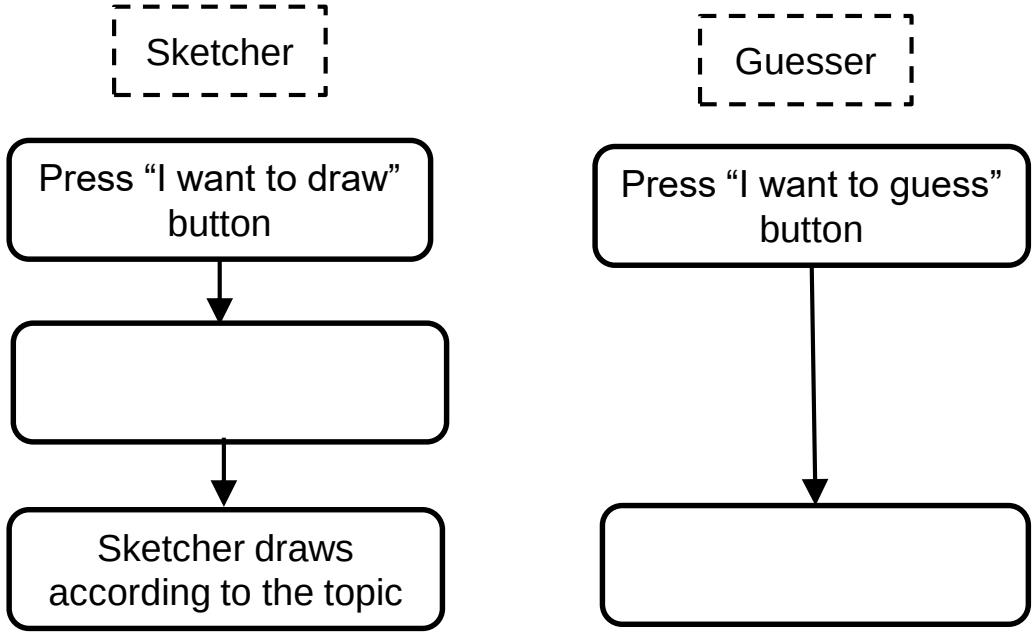


ORDER OF EVENTS

In Lesson 4, you built a Sketch and Guess app, but it would be nice for the guesser to be able to guess what is being drawn, and to check whether or not their answer is correct.

To do that, the Sketcher needs to send the current drawing to the other devices so the app can check whether or not the guess is correct.

A. See what the Sketcher draws	B. Show a random item for drawing



SKETCH AND GUESS

UNDERSTAND THE TAGS

A new tag and value will be stored in CloudDB to help all users play the game. “CurrentDrawing” will be set randomly by the Sketcher and stored so the Guessers can get the information to check for a correct guess.

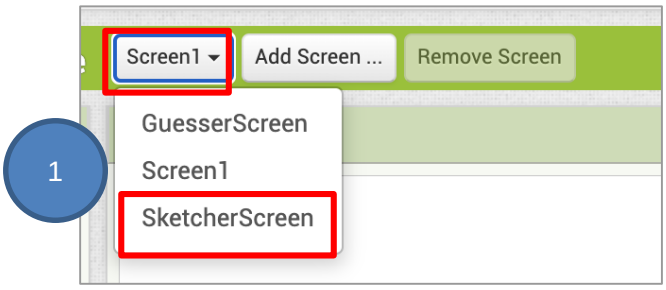
See the table below for the tags that are used in this app.

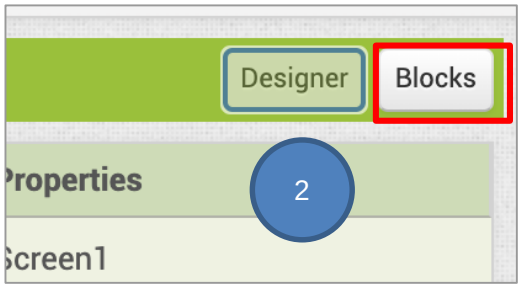
Tags	Meaning	Sketcher	Guesser
	The start point and end point for drawing	Store the coordinates of drawing	Get the coordinates of drawing
	What is being drawn	Store the randomly generated item to draw	Get the item being drawn for answer checking

SKETCH AND GUESS

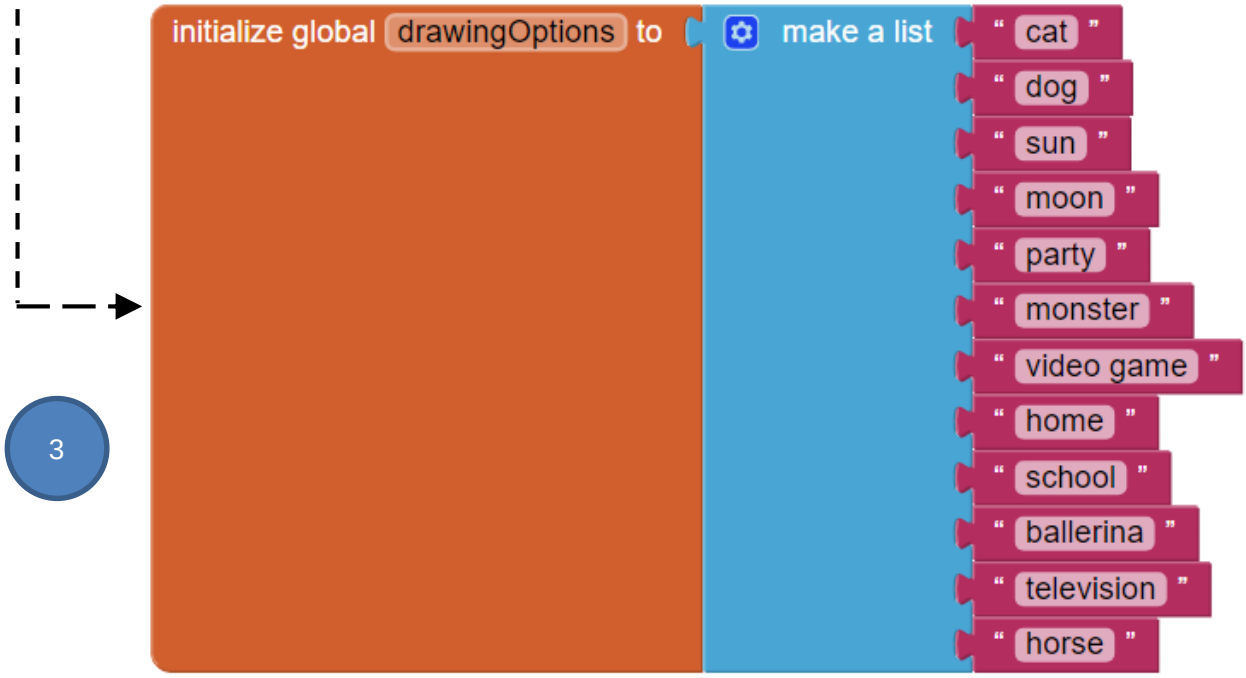
REVIEW THE GIVEN BLOCKS

❑ Open your project developed in Lesson 4. If you have not finished Part 4, please ask your teacher for help. Make sure you are in the **SketcherScreen** and in the **Blocks Editor**.





❑ Check the items in the drawingOptions list, which are used for the drawing topics (already built-in for you). Modify the items as you wish.

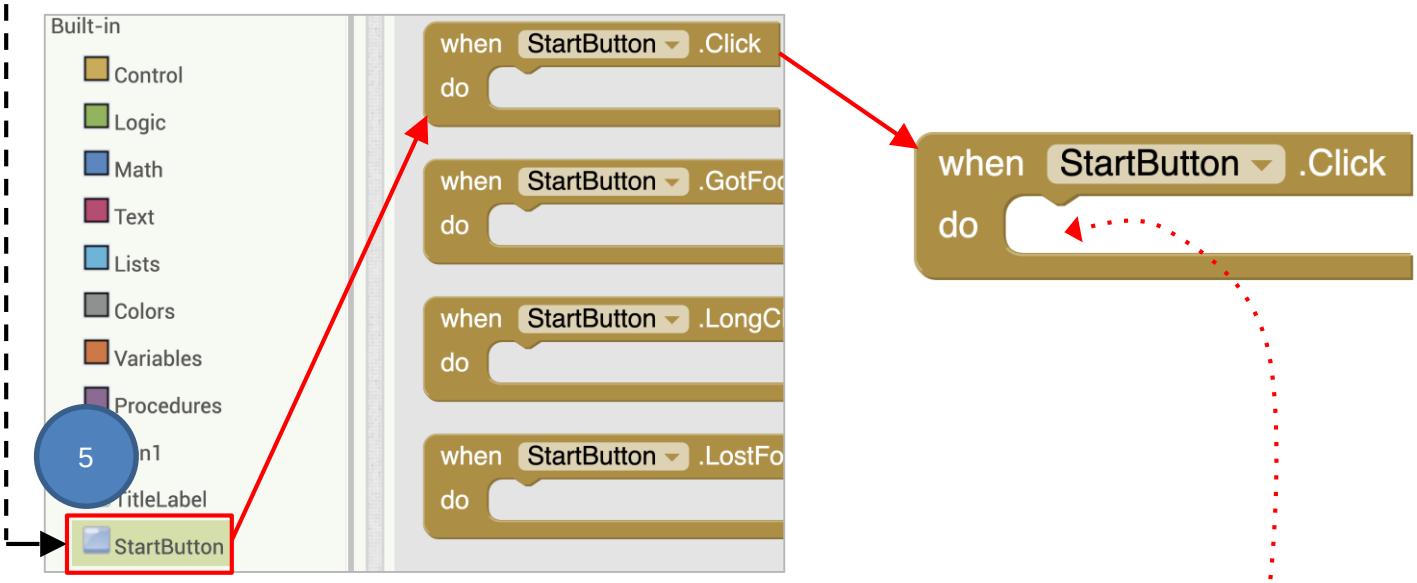


❑ Also note the “currentDrawing” variable, which will be used to save a random item from the drawingOptions list.

SKETCH AND GUESS

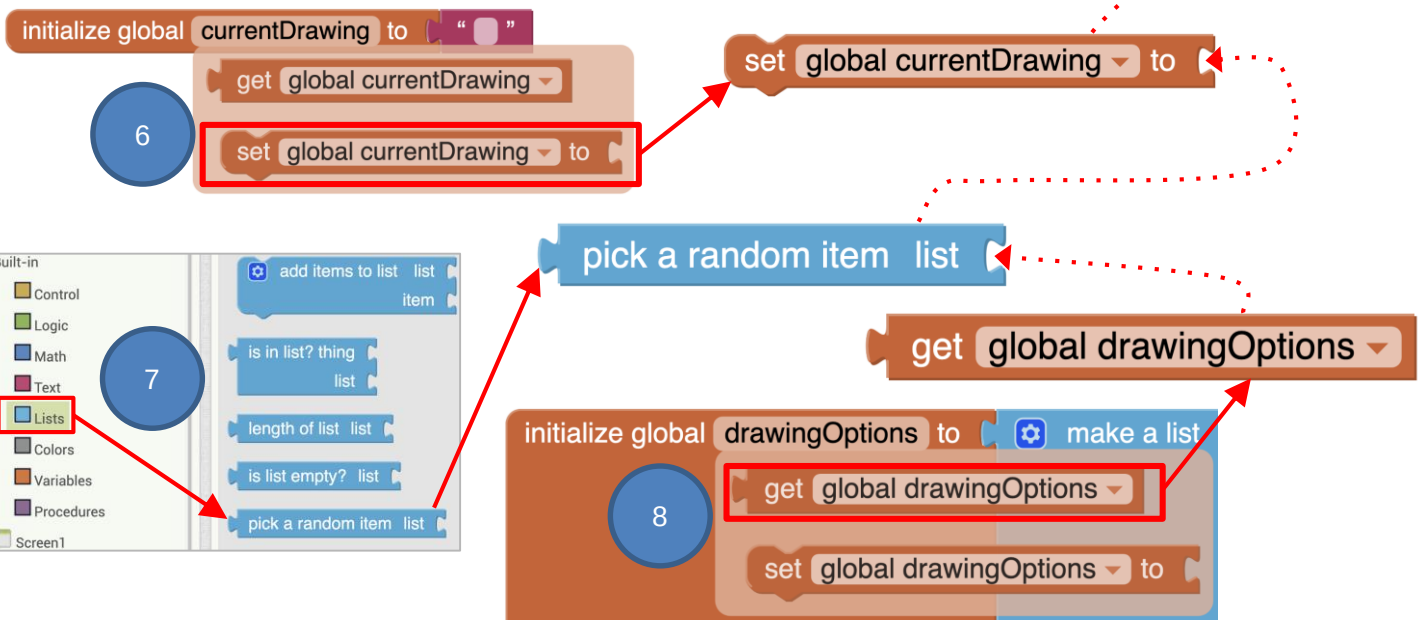
PICK A RANDOM ITEM WHEN STARTBUTTON IS CLICKED

❑ Pull out a **when StartButton.Click** block.



❑ Pick a random item from the drawingOptions list, then save it to the variable, currentDrawing.

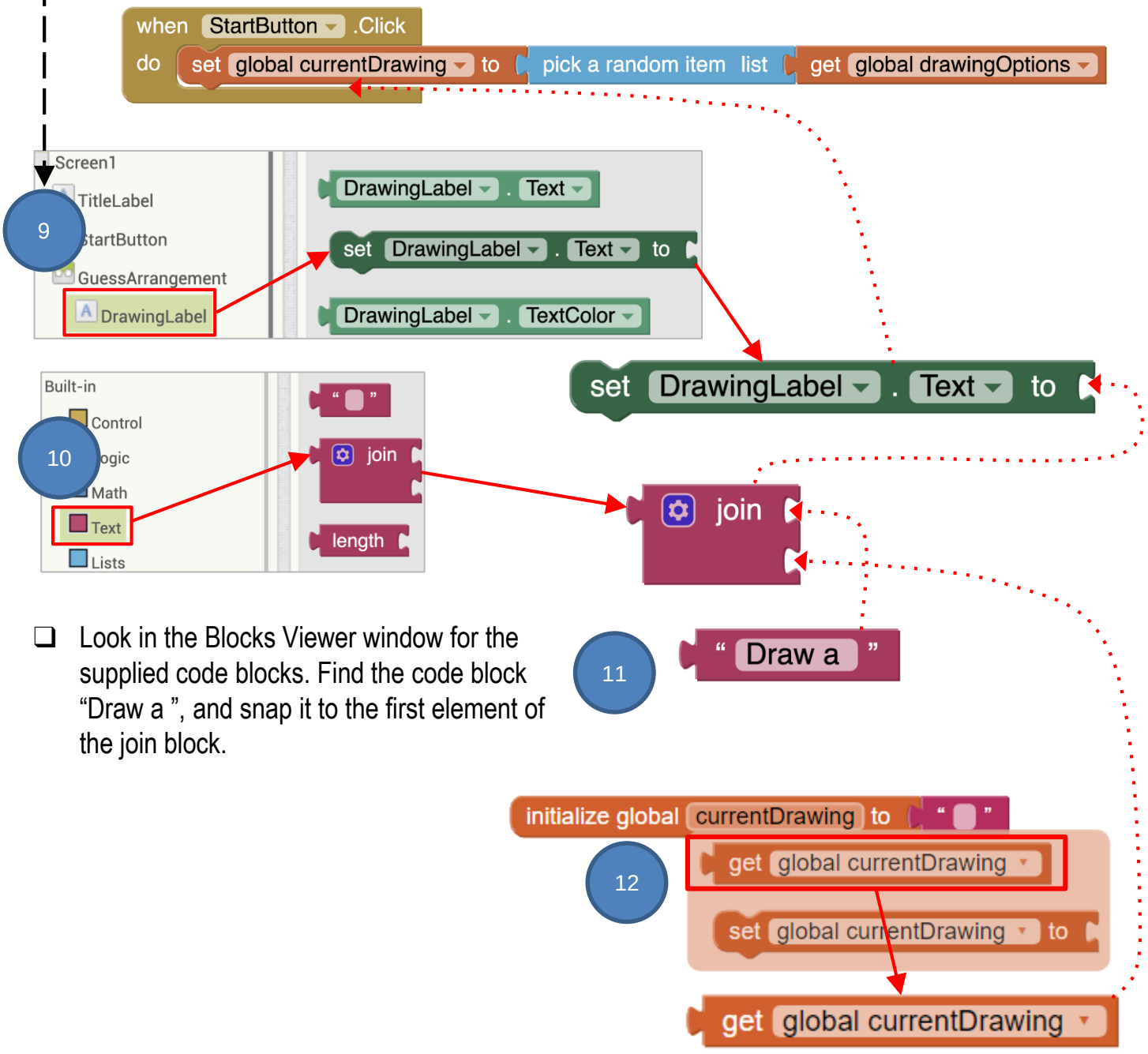
Mouse over currentDrawing



SKETCH AND GUESS

DISPLAY THE RANDOM ITEM IN THE DRAWINGLABEL

❑ Pull out a **set DrawingLabel.Text to** block.

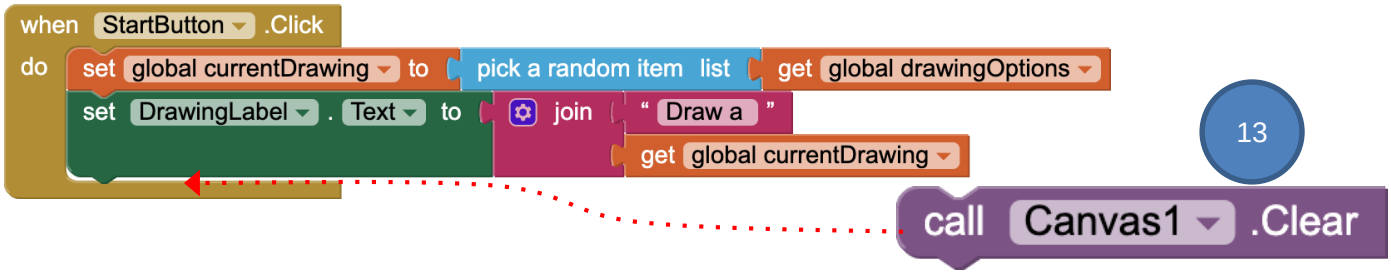


❑ Look in the Blocks Viewer window for the supplied code blocks. Find the code block “Draw a ”, and snap it to the first element of the join block.

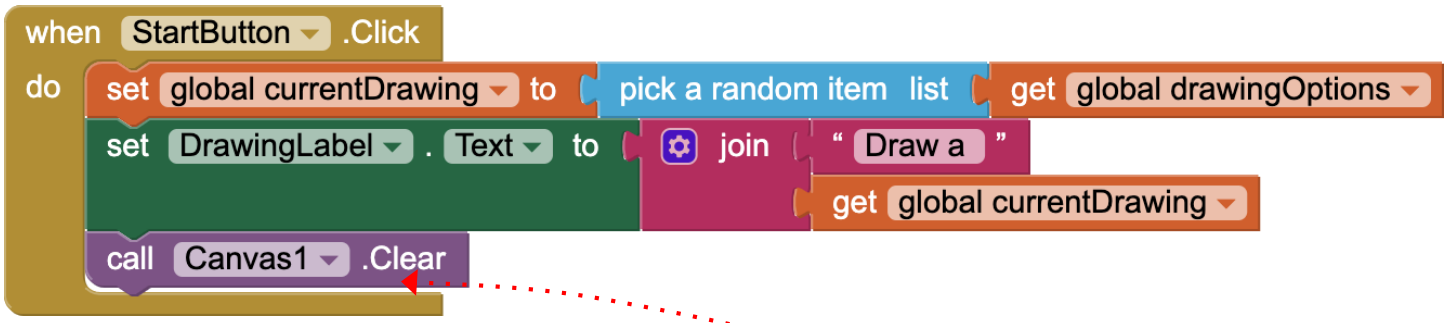
SKETCH AND GUESS

STORE CURRENTDRAWING IN CLOUDDB

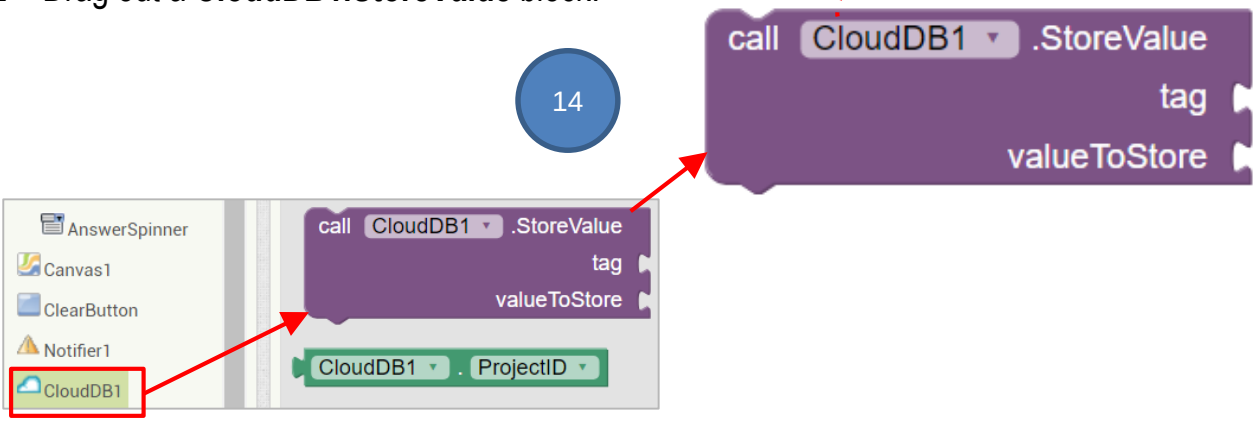
❑ Since the Sketcher is starting a new picture, clear the Canvas.



❑ The Sketcher will store the current drawing in CloudDB, so it can be passed to the Guessers. You'll use the tag, "CurrentDrawing".



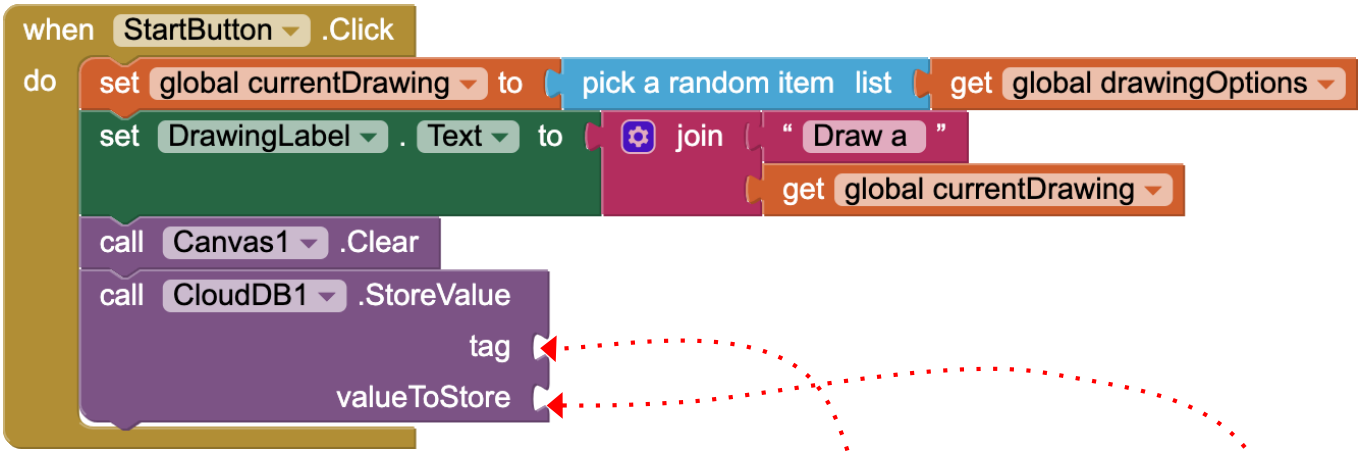
❑ Drag out a **CloudDB1.StoreValue** block.



SKETCH AND GUESS

STORE CURRENTDRAWING IN CLOUDDB (CONT'D)

❑ Add the **DrawingData** block to the tag slot and the **create empty list** block to the valueToStore slot, so that the canvas of GuesserScreen will be cleared when the StartButton of SketcherScreen is pressed.



❑ Look in the Blocks Viewer window for the supplied code blocks. Find the code block **DrawingData**", right-click on it and duplicate the block.

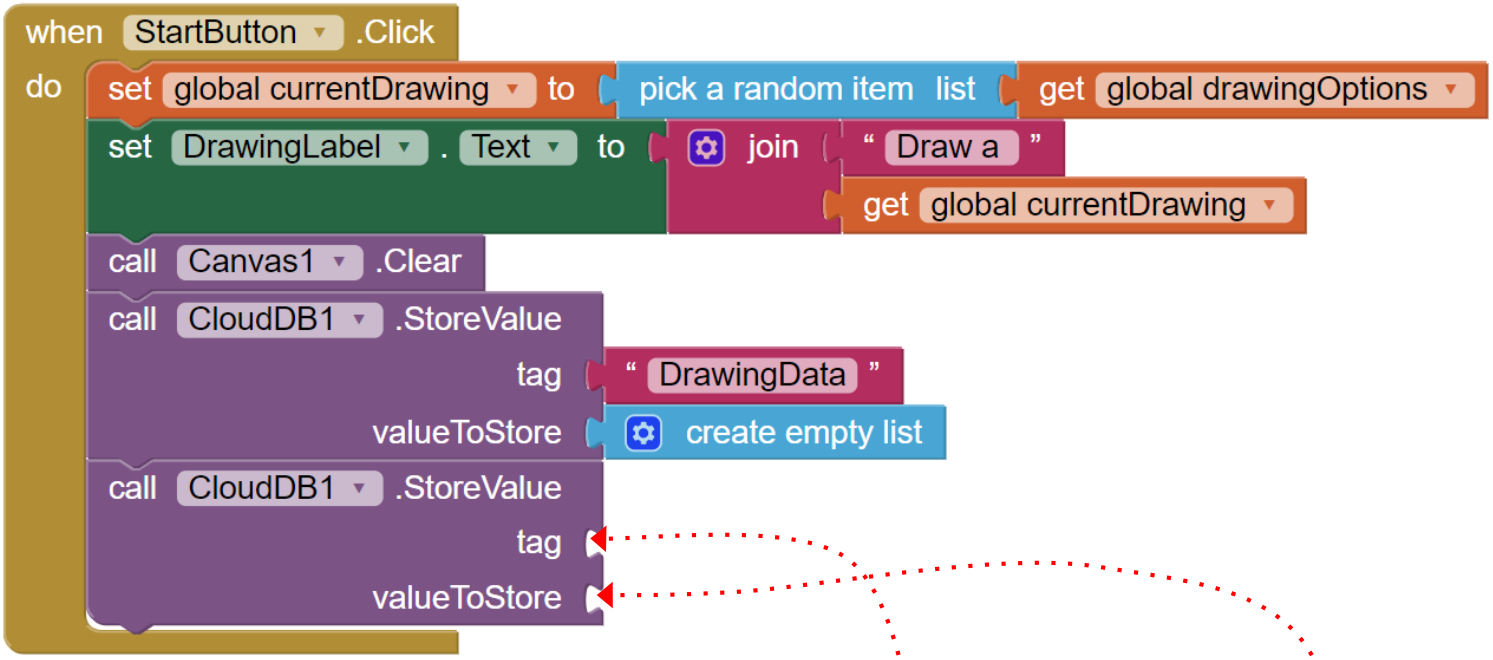
15

16

SKETCH AND GUESS

STORE CURRENTDRAWING IN CLOUDDB (CONT'D)

❑ Add the tag and valueToStore.



17

" CurrentDrawing "

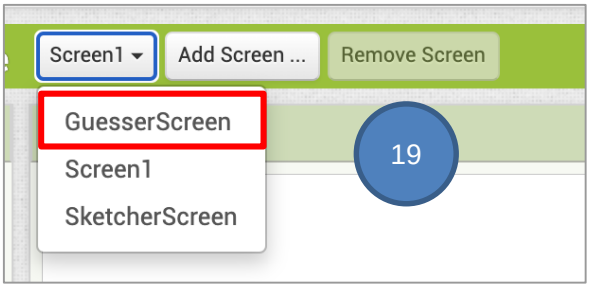
❑ Look in the Blocks Viewer window for the supplied code blocks. Find the code block **CurrentDrawing**, and drag it to **tag**.

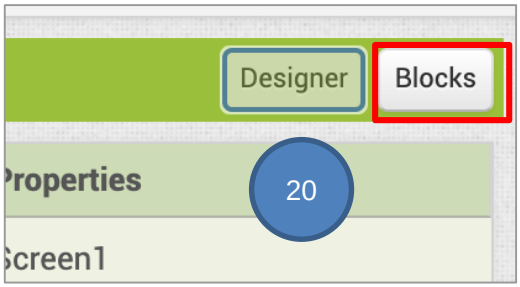
18

SKETCH AND GUESS

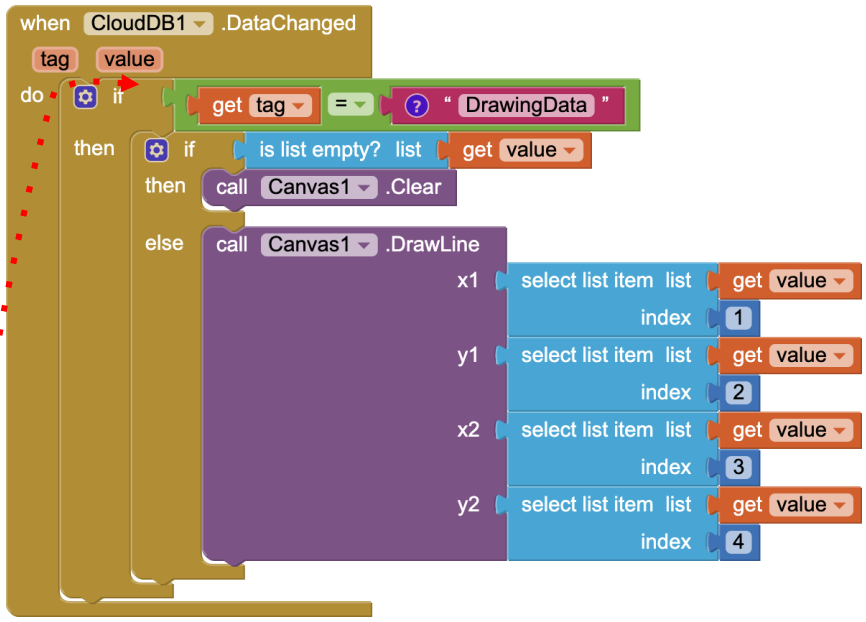
GET CURRENTDRAWING FOR GUESSER

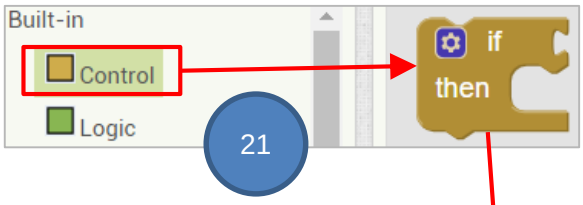
❑ Switch to the GuesserScreen and make sure you are in the Blocks Editor.

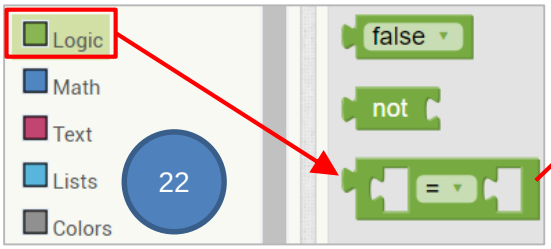




❑ The Sketcher stored the **CurrentDrawing** to CloudDB, and CloudDB needs to update the GuesserScreen so it has the correct **CurrentDrawing** to check for a correct guess.



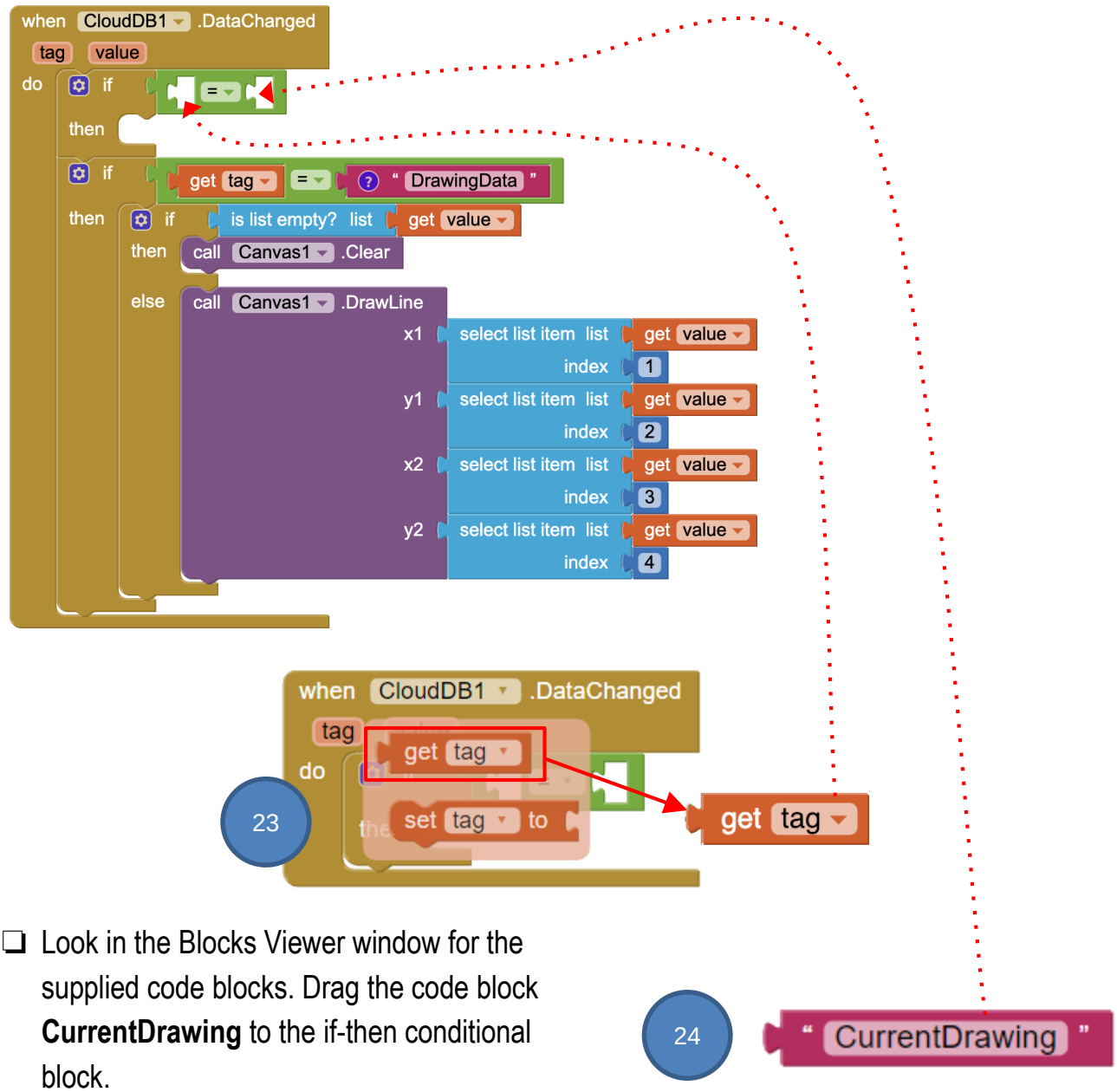




SKETCH AND GUESS

GET CURRENTDRAWING FOR GUESSER (CON'T)

❑ Test whether the tag is "CurrentDrawing".

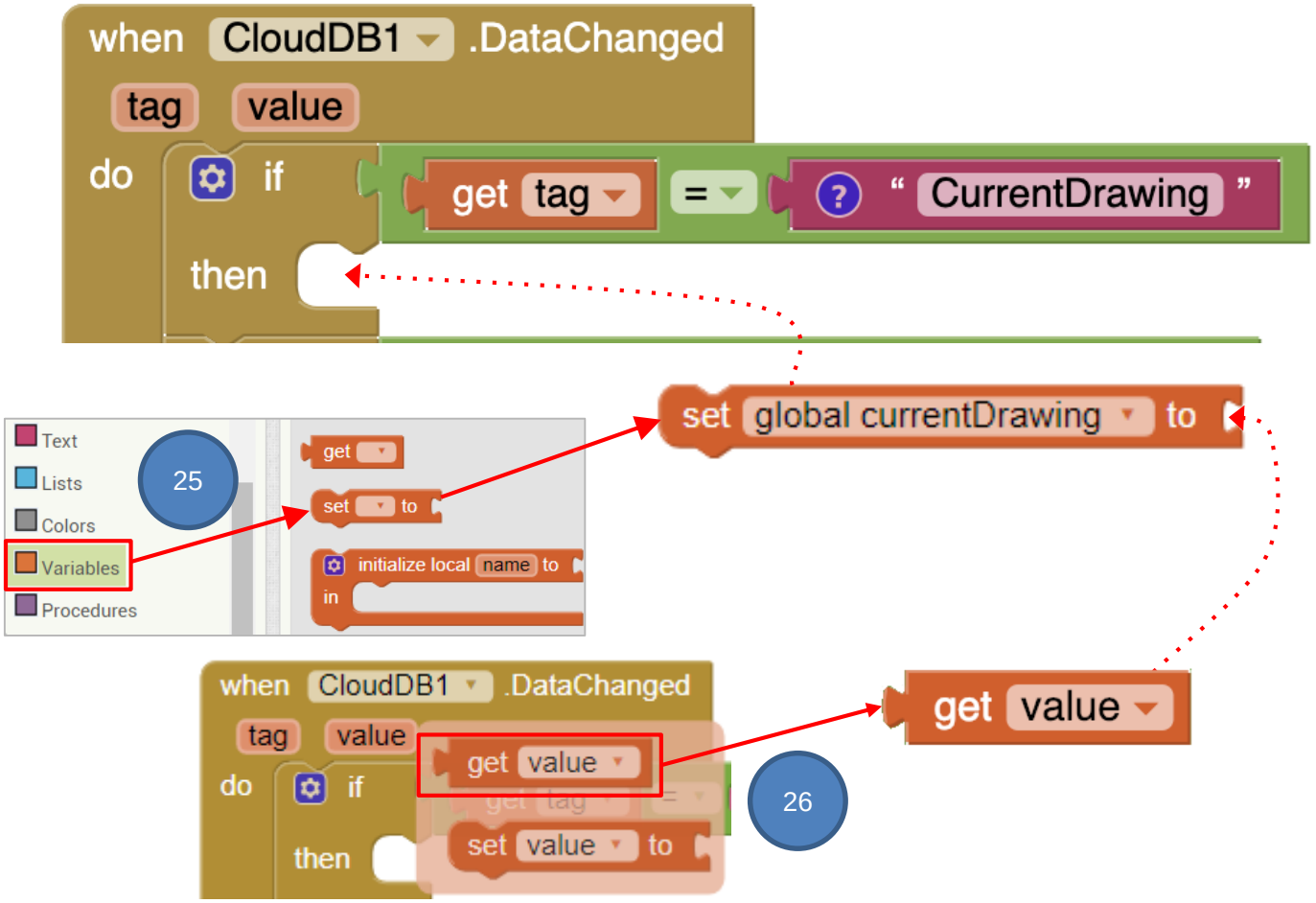


❑ Look in the Blocks Viewer window for the supplied code blocks. Drag the code block **CurrentDrawing** to the if-then conditional block.

SKETCH AND GUESS

GET CURRENTDRAWING FOR GUESSER (CON'T)

❑ Set the **currentDrawing** variable to the value received from CloudDB.



In Lesson 6, you'll use the CurrentDrawing information to check whether or not the Guesser's guess is correct.

SKETCH AND GUESS

COMPUTATIONAL THINKING CONCEPTS

The following are the computational thinking concepts learnt in Lesson 5.

L3U8.6-8.8 Sketch And Guess
1. Conditionals 
2. Naming and Variables 
3. Data Structures