SKETCH & GUESS: PART 4

In this lesson, you will add answer-checking to the sketching app.

SEQUENCE OF EVENTS

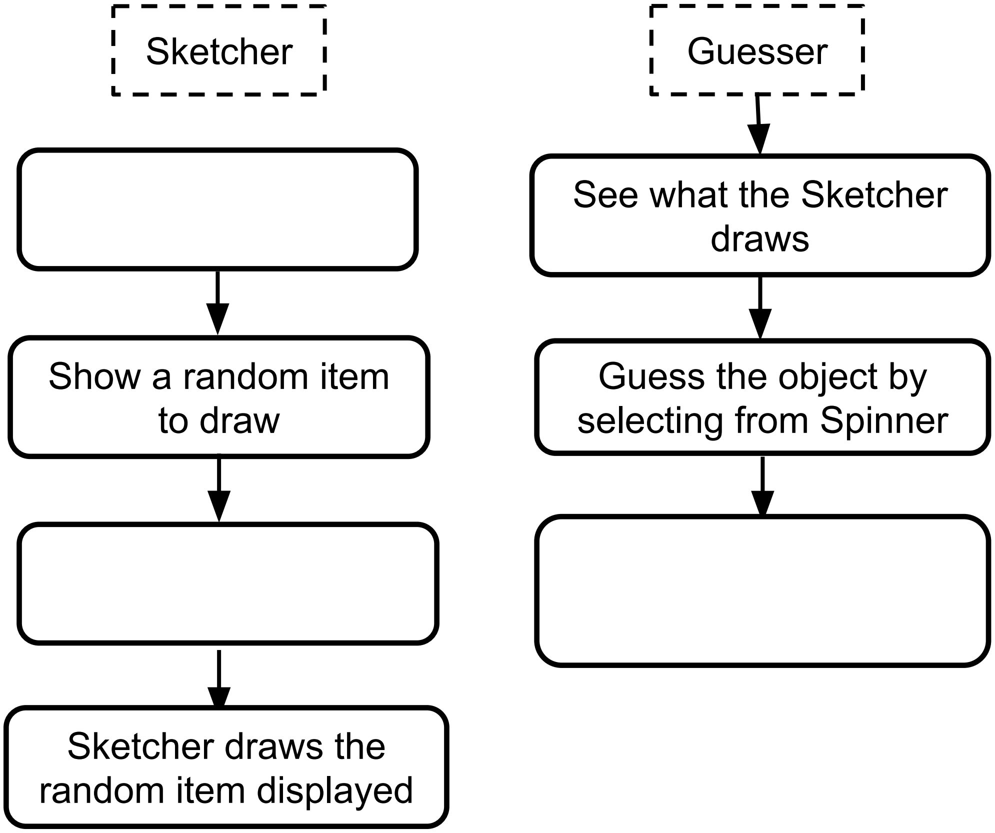
In previous lessons, you built a Sketch and Guess app.

Review with your partner the diagram below. Fill in the empty spaces with A, B, or C.

A. Check whether the selected answer is correct

B. Store the randomly selected drawing item in CloudDB

C. Sketcher presses New Picture button

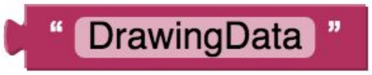
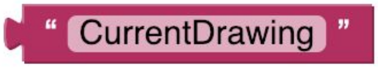
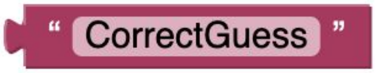

Sketcher
Show a random item to draw
Sketcher draws the random item displayed
Guesser
See what the Sketcher draws
Guess the object by selecting from Spinner

CLOUDDB TAGS

You’ve already used CloudDB to store the drawing data from Sketcher to Guesser.

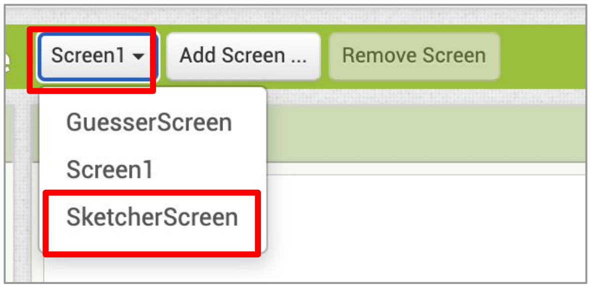
Some new tags and values will be stored in CloudDB to help all users play the game. “CurrentDrawing” will be set randomly by the Sketcher and stored so the Guesser can get the information to check for a correct guess. “CorrectGuess” is the guesser’s way to letting other players know they have guessed correctly.

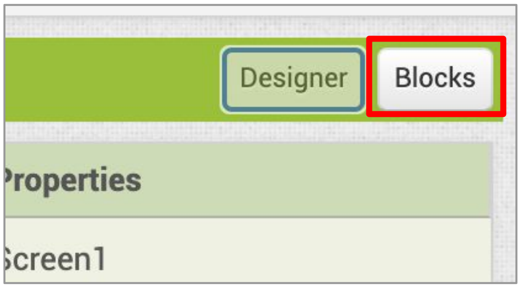
See the table below for the tags that are used in this app.

Tags	Meaning	Sketcher	Guesser
 “ DrawingData ”	The start point and end point for drawing	Store the coordinates of drawing	Get the coordinates of drawing
 “ CurrentDrawing ”	What is being drawn	Store the randomly generated item to draw	Get the item being drawn for answer checking
 “ CorrectGuess ”	Guesser made a correct guess of the drawing	Get notification of a correct guess	Store the correct guess

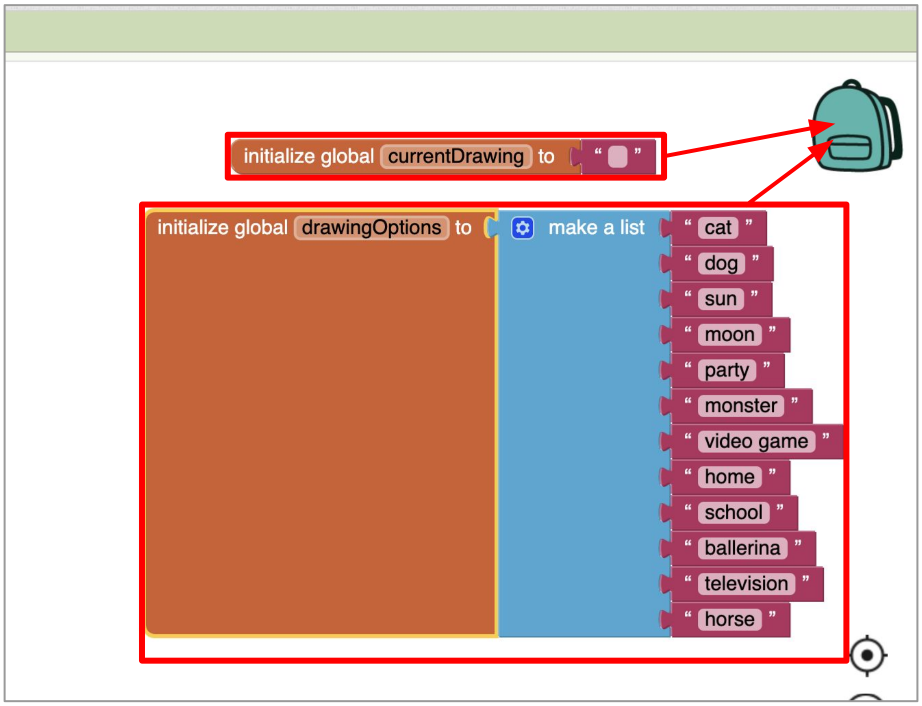
MOVE BLOCKS USING BACKPACK

1 Make sure you are in the **SketcherScreen** and in the **Blocks Editor**.



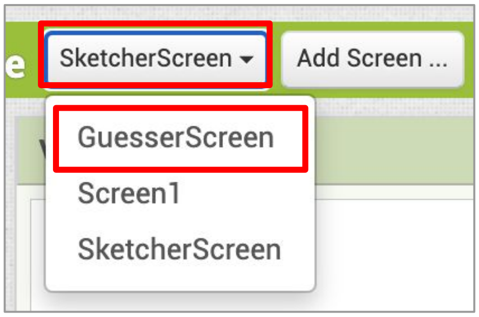


2 You want to include the **drawingOptions** list and the **currentDrawing** variable in the GuesserScreen. An easy way to do this is to drag these blocks into the Backpack in the upper right corner of the SketcherScreen.


initialize global currentDrawing to " "

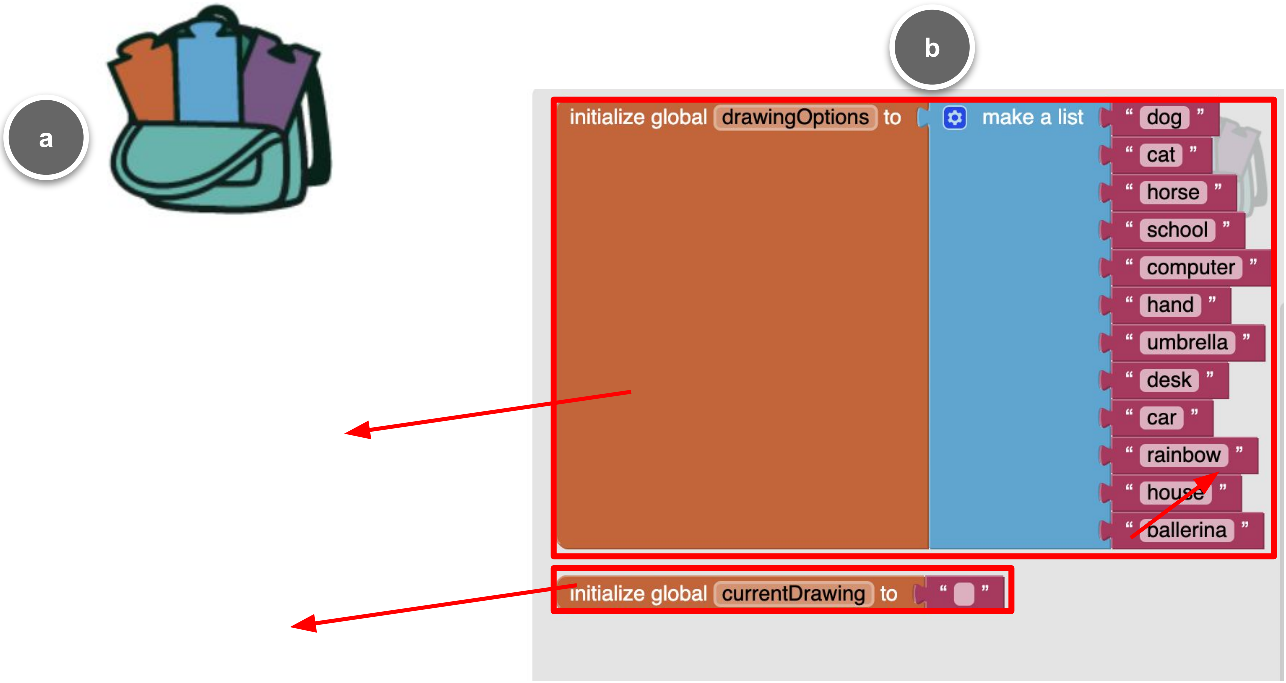
initialize global drawingOptions to make a list "cat" "dog" "sun" "moon" "party" "monster" "video game" "home" "school" "ballerina" "television" "horse"

3 Switch to GuesserScreen.



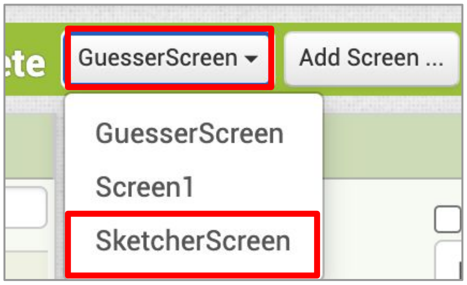
MOVE BLOCKS USING BACKPACK (continued)

4 Click on the full backpack, and then drag out the blocks.



```
initialize global drawingOptions to  make a list  " dog "
                                                  " cat "
                                                  " horse "
                                                  " school "
                                                  " computer "
                                                  " hand "
                                                  " umbrella "
                                                  " desk "
                                                  " car "
                                                  " rainbow "
                                                  " house "
                                                  " ballerina "
initialize global currentDrawing to " "
```

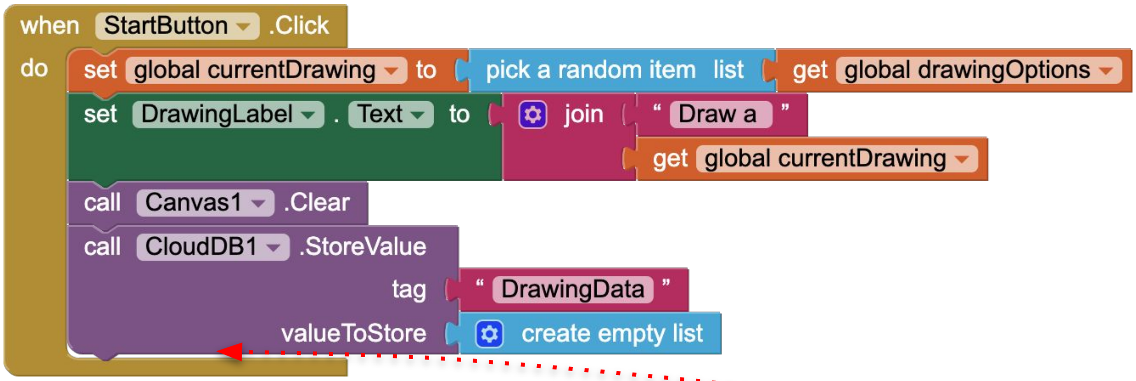
5 Switch back to SketcherScreen.



MIT APP INVENTOR

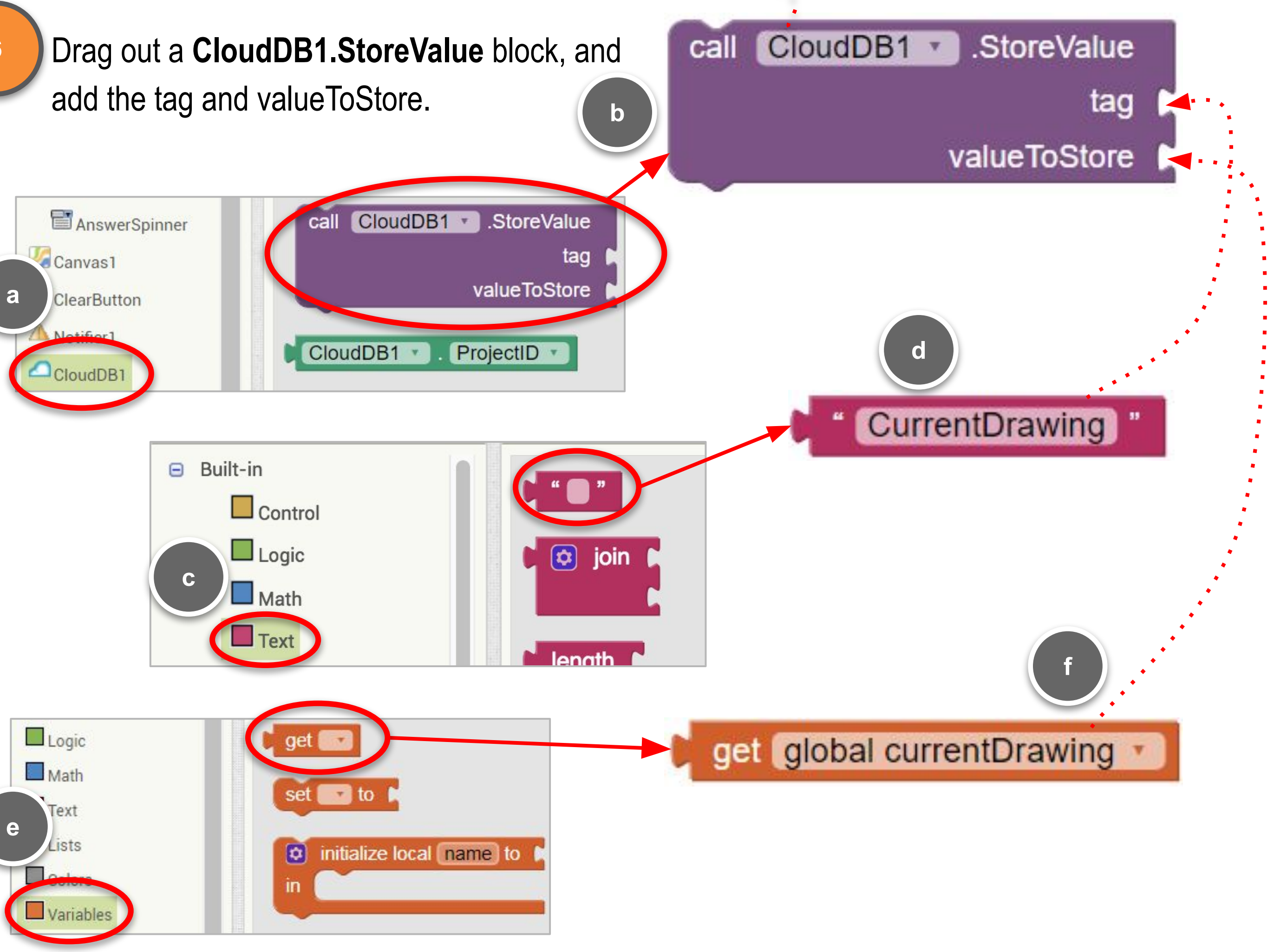
STORE CURRENT DRAWING

The Sketcher will store the current drawing in CloudDB, so it can be passed to the the Guesser. You'll use the tag, "CurrentDrawing". That information will be passed to the Guesser so the app can check for a correct answer.



```
when StartButton .Click
do  set global currentDrawing to  pick a random item list  get global drawingOptions
    set DrawingLabel . Text to  join  " Draw a "
                                      get global currentDrawing
    call Canvas1 .Clear
    call CloudDB1 .StoreValue
                        tag  " DrawingData "
               valueToStore  create empty list
```

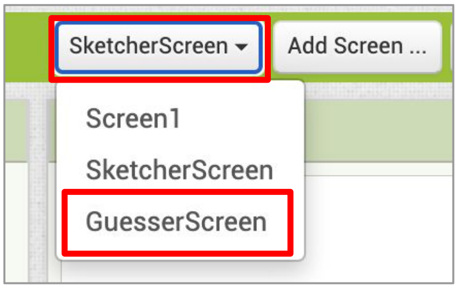
6 Drag out a **CloudDB1.StoreValue** block, and add the tag and valueToStore.



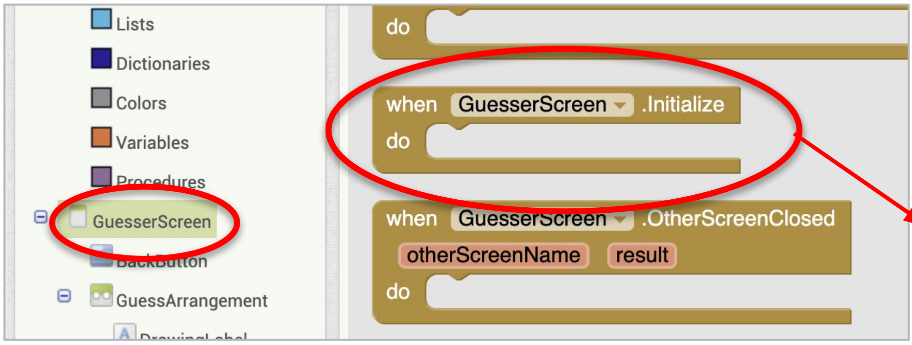
INITIALIZE GUESSER SCREEN

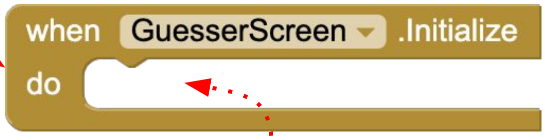
AnswerSpinner is a component that will list the possible objects being drawn, so the Guesser can make a guess when the Sketcher draws something. You need to set the **AnswerSpinner's** Elements to the list of drawingOptions.

7 Switch to **GuesserScreen**.

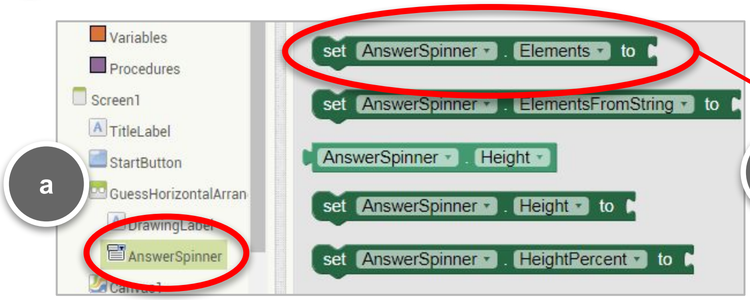


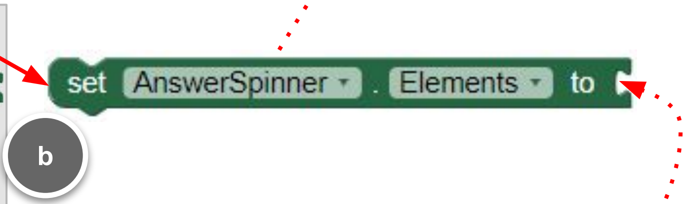
8 Pull out a **GuesserScreen.Initialize** block.

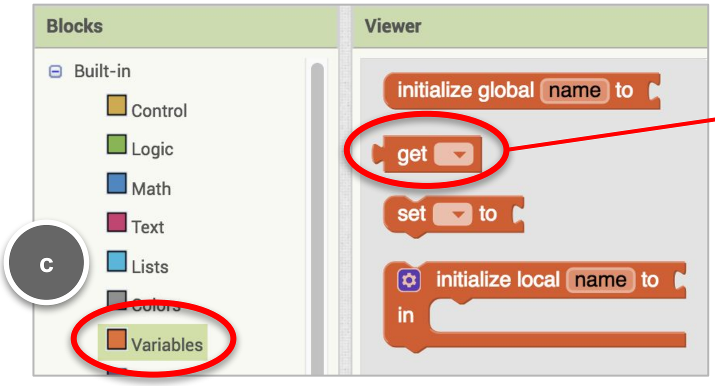


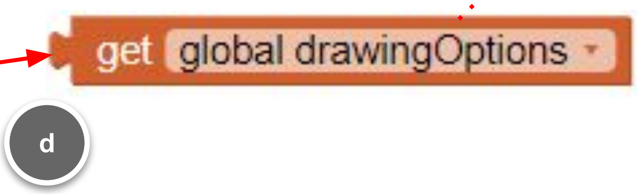


9 Set the **AnswerSpinner.Elements** to the drawingOptions list.


a
Variables
Procedures
Screen1
TitleLabel
StartButton
GuessHorizontalArran
DrawingLabel
AnswerSpinner
Canvas1
set AnswerSpinner . Elements to
set AnswerSpinner . ElementsFromString to
AnswerSpinner . Height
set AnswerSpinner . Height to
set AnswerSpinner . HeightPercent to

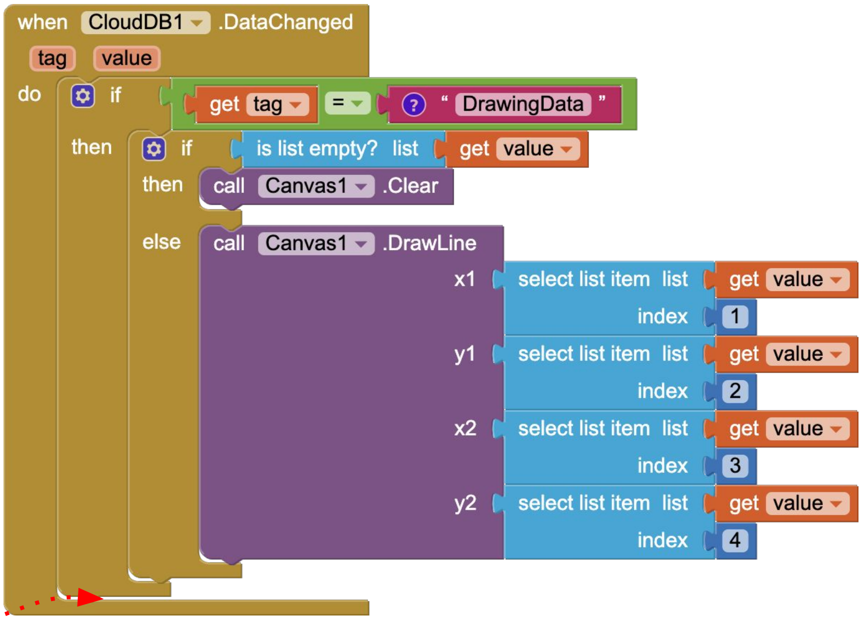




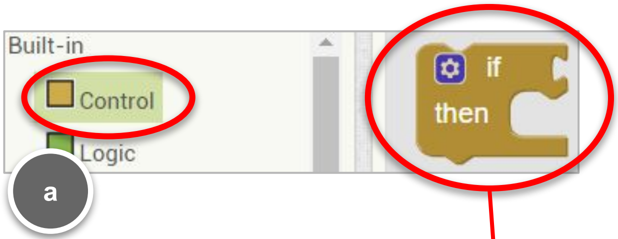


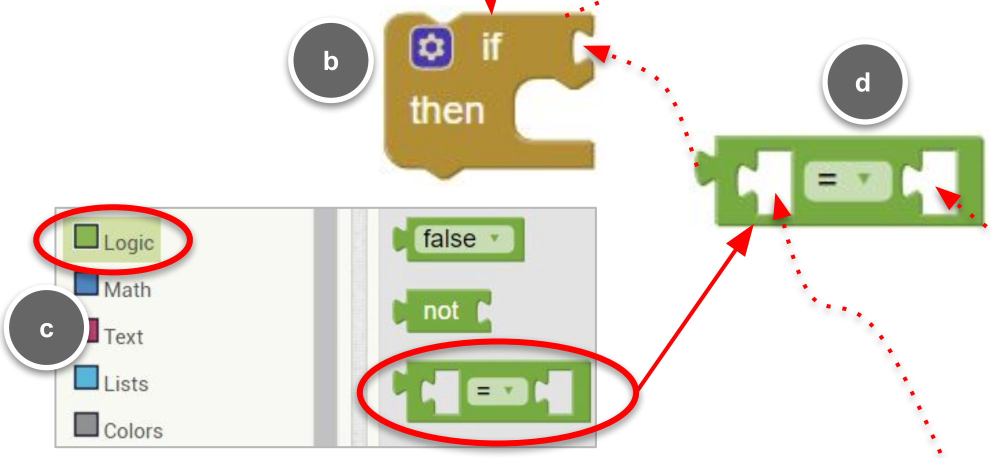
GET CURRENTDRAWING FOR THE GUESSER

10 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.

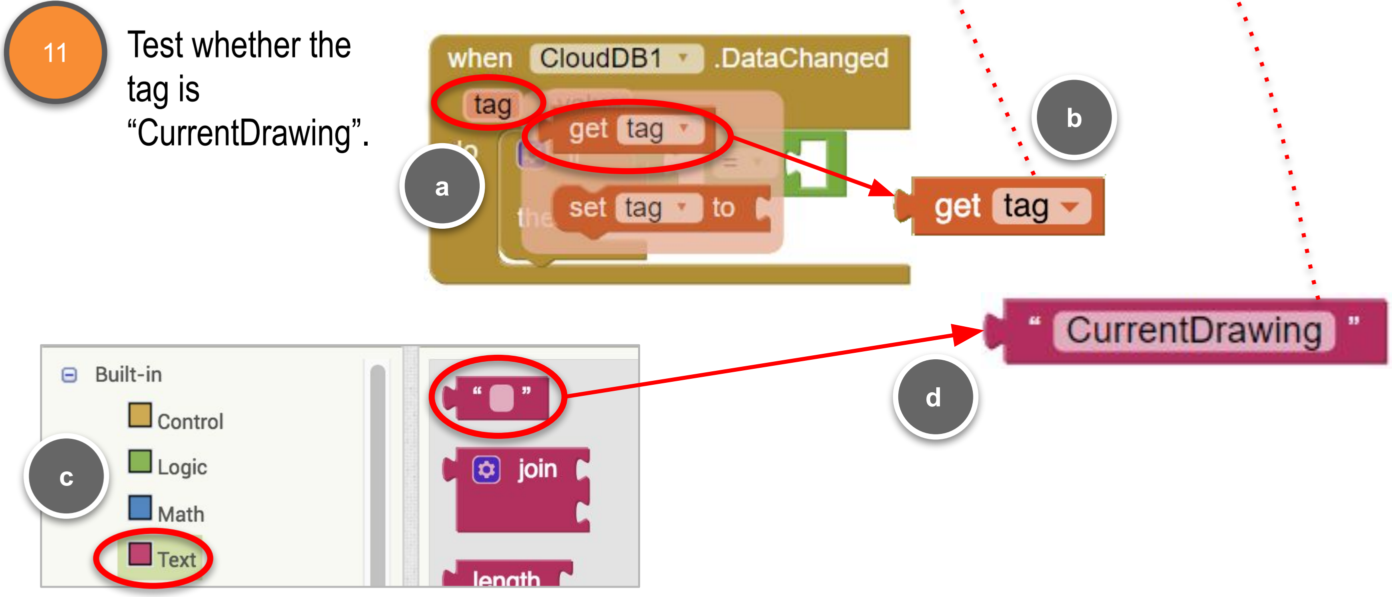


```
when CloudDB1 .DataChanged
  tag  value
do  if  get tag = " DrawingData "
    then  if  is list empty? list  get value
          then  call Canvas1 .Clear
          else  call Canvas1 .DrawLine
                  x1  select list item  list  get value
                                        index  1
                  y1  select list item  list  get value
                                        index  2
                  x2  select list item  list  get value
                                        index  3
                  y2  select list item  list  get value
                                        index  4
```



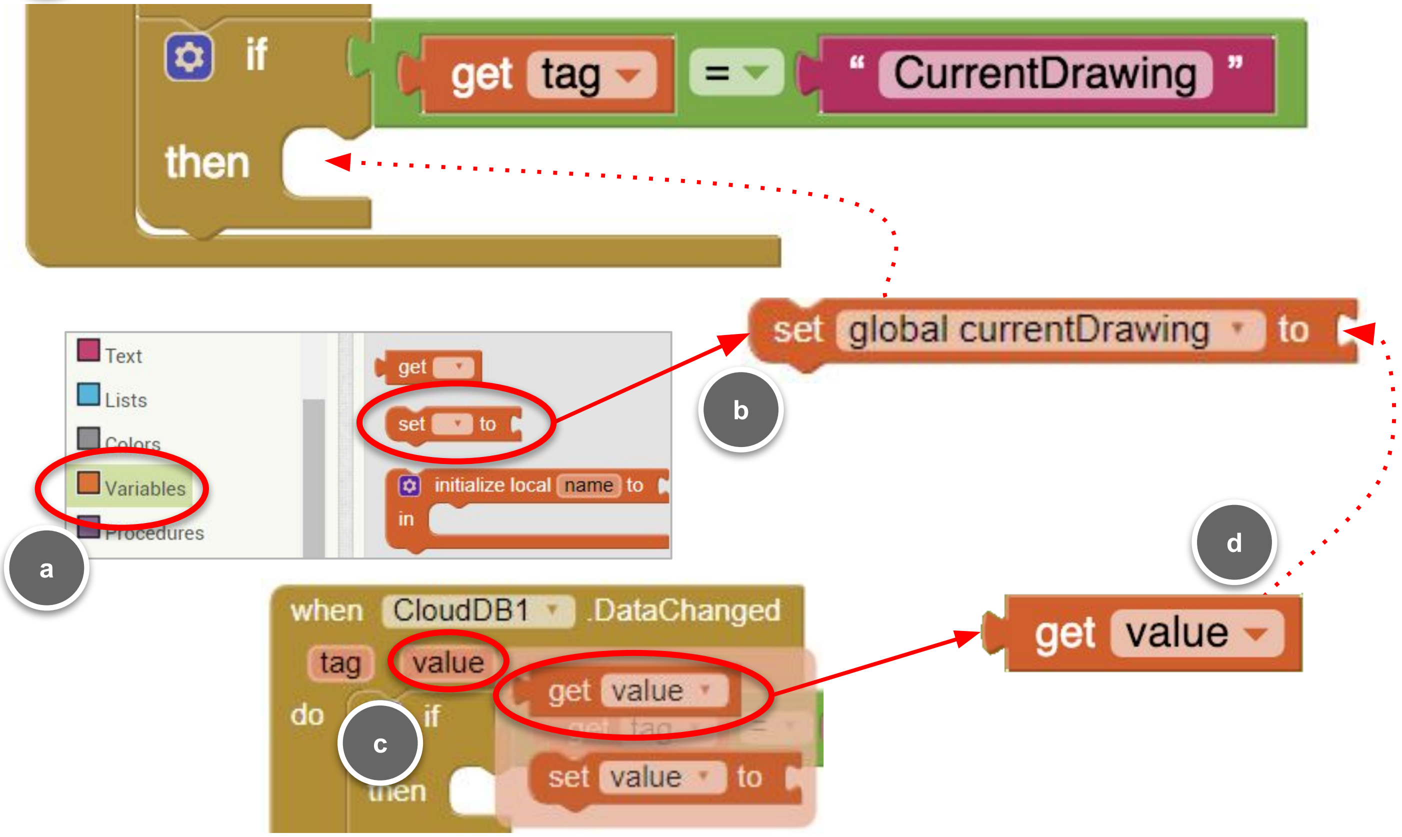


11 Test whether the tag is “CurrentDrawing”.



SET THE ANSWER FOR CHECKING

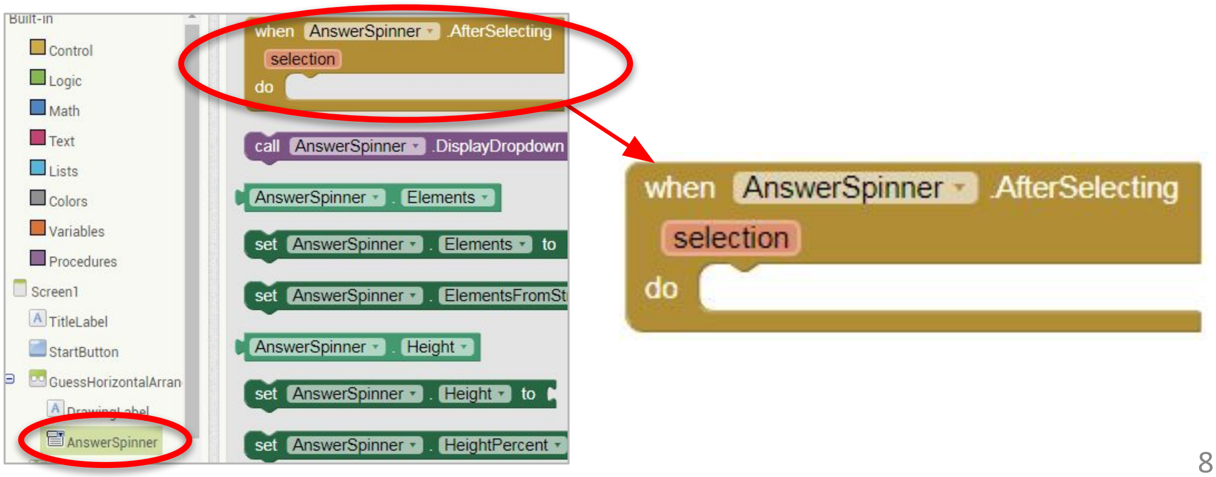
12 Set the **currentDrawing** variable to the value received from CloudDB.



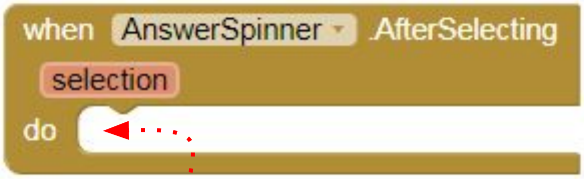
CHECK ANSWER

The Guesser makes a guess by selecting from the **AnswerSpinner**. When they do, check if the guess is correct.

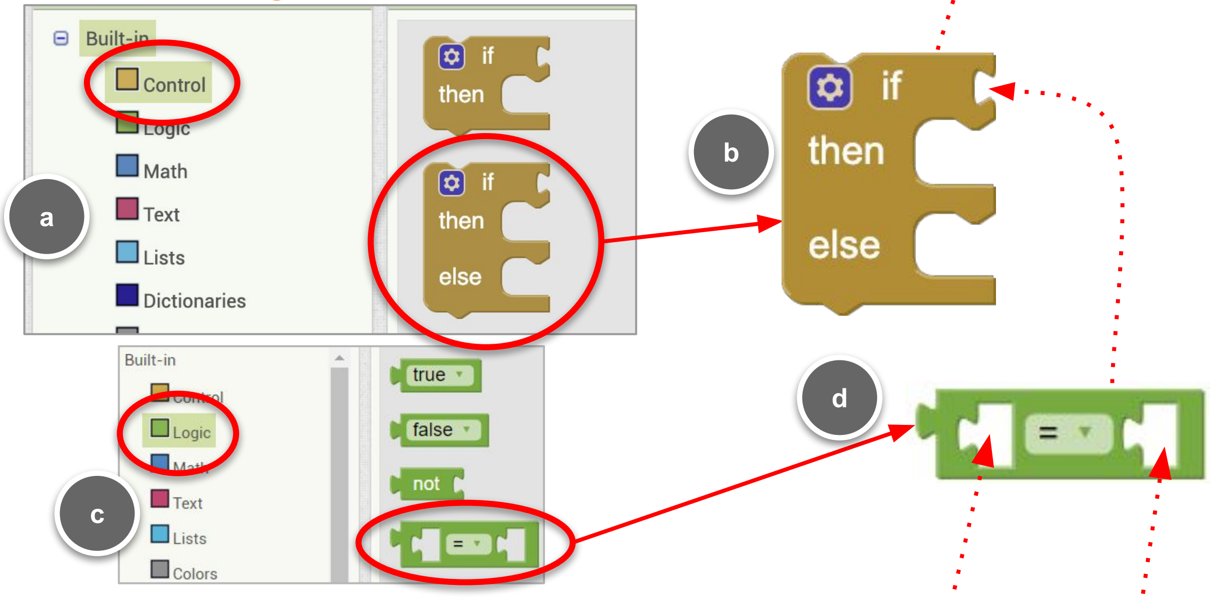
13 Check the answer in the **AnswerSpinner.AfterSelecting** block.


Built-in
Control
Logic
Math
Text
Lists
Colors
Variables
Procedures
Screen1
TitleLabel
StartButton
GuessHorizontalArran
DrawingLabel
AnswerSpinner
when AnswerSpinner .AfterSelecting
selection
do
call AnswerSpinner .DisplayDropdown
AnswerSpinner . Elements
set AnswerSpinner . Elements to
set AnswerSpinner . ElementsFromSt
AnswerSpinner . Height
set AnswerSpinner . Height to
set AnswerSpinner . HeightPercent
when AnswerSpinner .AfterSelecting
selection
do

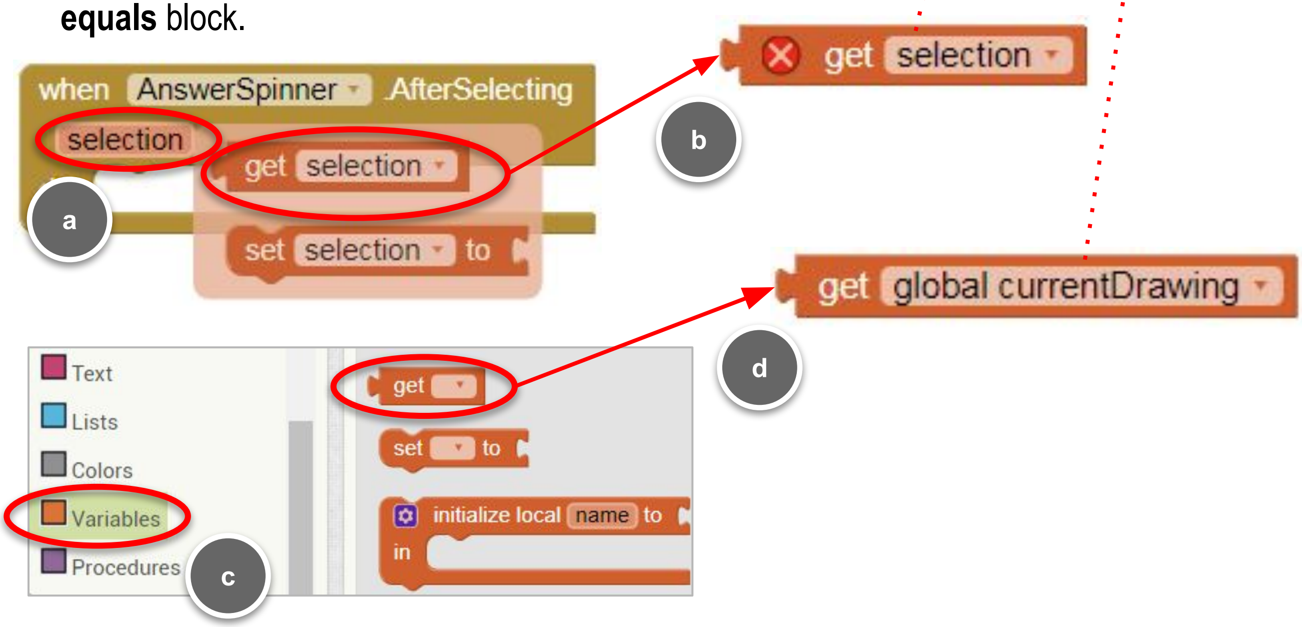
CHECK ANSWER (continued)



14 Test if the user's selection matches the **currentDrawing**.

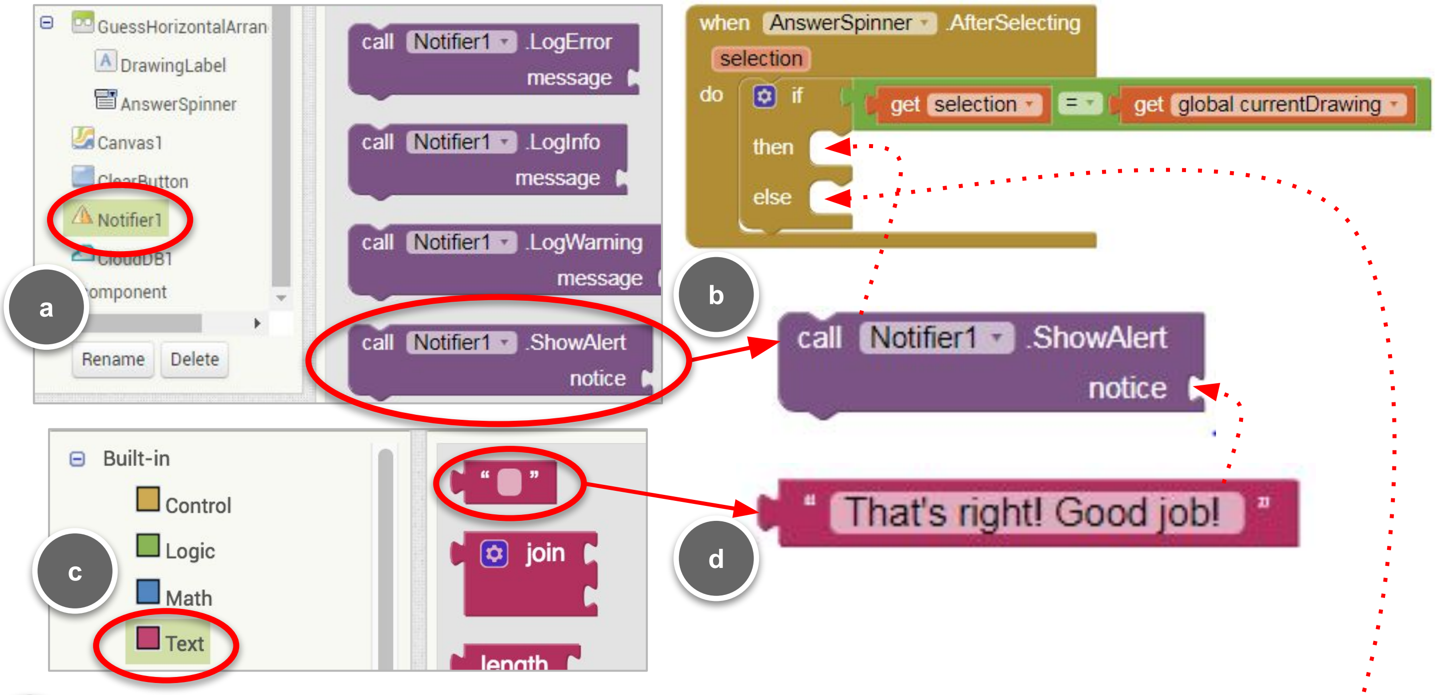


15 Snap a **get selection** and a **get global current Drawing** block into the **equals** block.

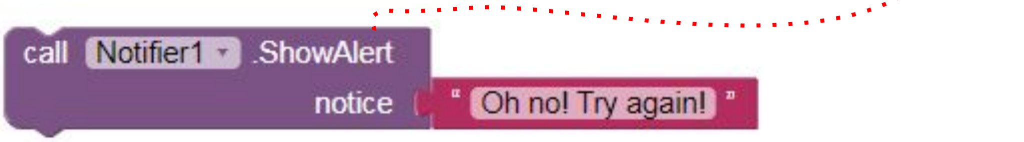


CHECK ANSWER (continued)

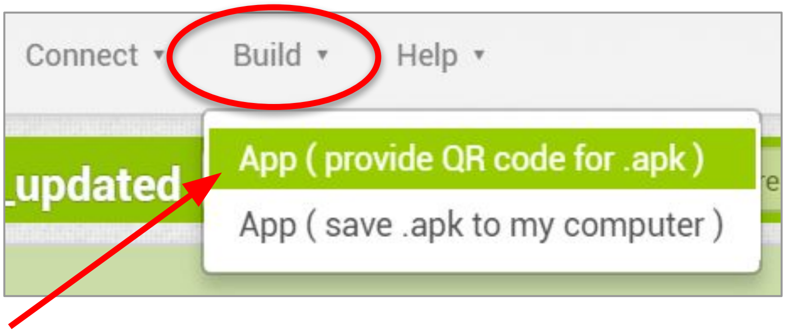
16 Drag out a **Notifier1.ShowAlert** block and set the messages. If the user's selection in the **AnswerSpinner** matches **currentDrawing**, notify the user that they are correct: *"That's right! Good job!"*; otherwise, notify user *"Oh no! Try again!"*.



17 Duplicate the **Notifier1.ShowAlert** block, change the text block to "Oh no! Try again!, and add it as the **else** clause of the **if-then-else** block.



18 Test with your partner. Check for correct and incorrect guesses on the Guesser's device, and make sure the correct notification is displayed.

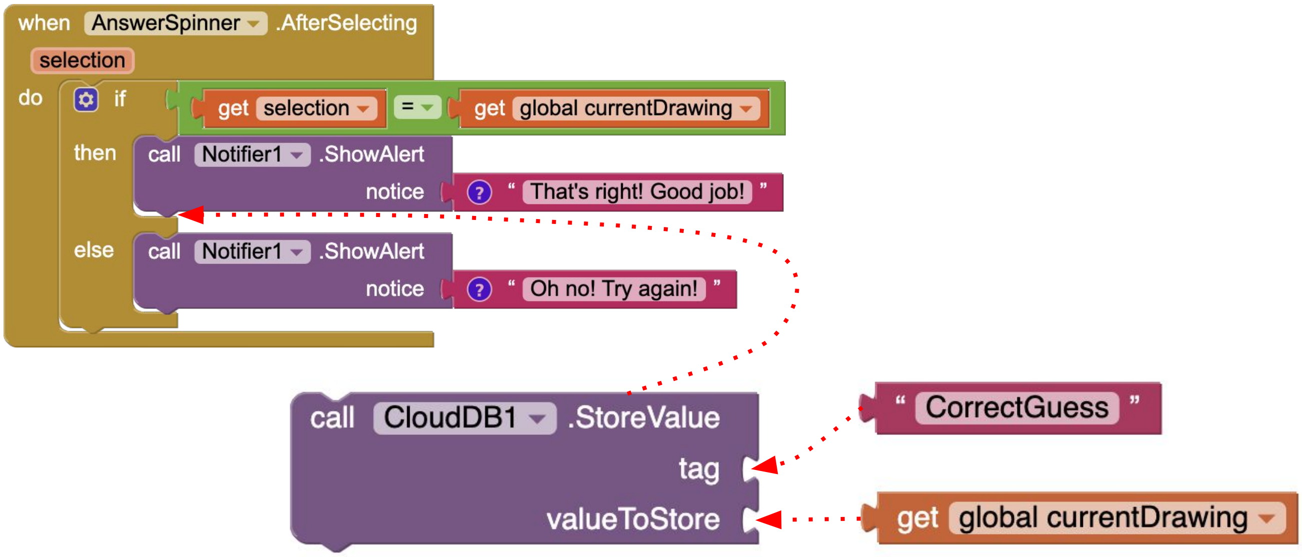

Connect
Build
Help
updated
App (provide QR code for .apk)
App (save .apk to my computer)

UPDATE USERS WITH CORRECT GUESS

The last thing to do is to notify all other players, including the Sketcher, when someone has guessed the correct answer.

The Guesser who guesses correctly should store that information in CloudDB, which in turn will send that information out to all other users.

19 In the **if-then-else** block in **AnswerSpinner.AfterSelecting**, add a **CloudDB1.StoreValue** block. For a tag, use "**CorrectGuess**", and store as the value the variable **currentDrawing**.

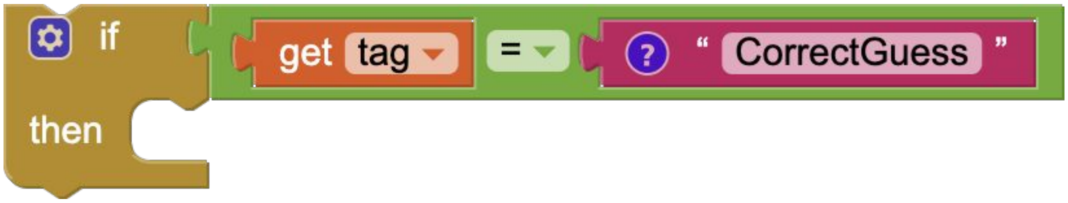


```
when AnswerSpinner .AfterSelecting
  selection
do  if   get selection = get global currentDrawing
    then call Notifier1 .ShowAlert
                 notice  " That's right! Good job! "
    else call Notifier1 .ShowAlert
                 notice  " Oh no! Try again! "

call CloudDB1 .StoreValue
                     tag  " CorrectGuess "
            valueToStore  get global currentDrawing
```

All users should be notified of a correct guess, the Sketcher and all Guessers.

20 Add another **if-then** block to the **CloudDB1.DataChanged** event, and test if the **tag** = "**CorrectGuess**".

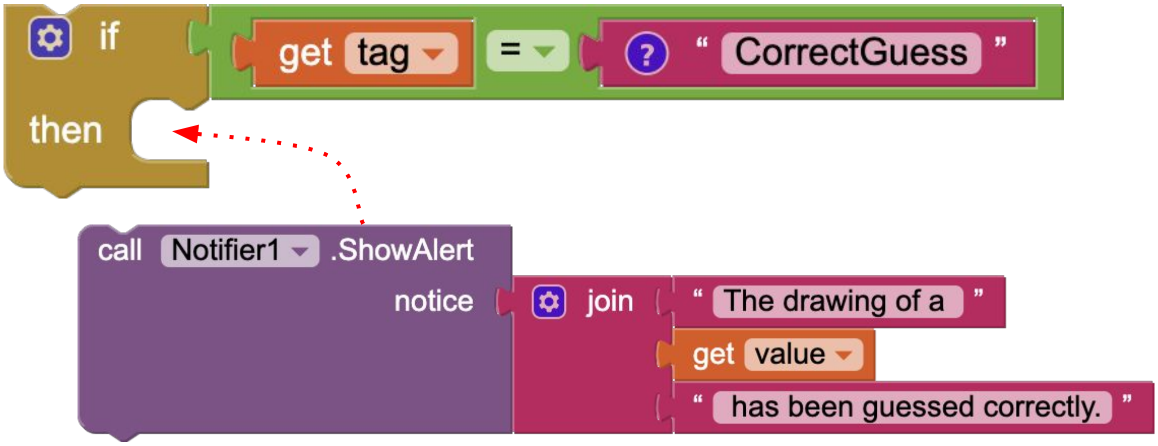


```
if   get tag = " CorrectGuess "
then
```

NOTIFY USERS

For the Guessers, it is enough to notify them that someone made the correct guess.

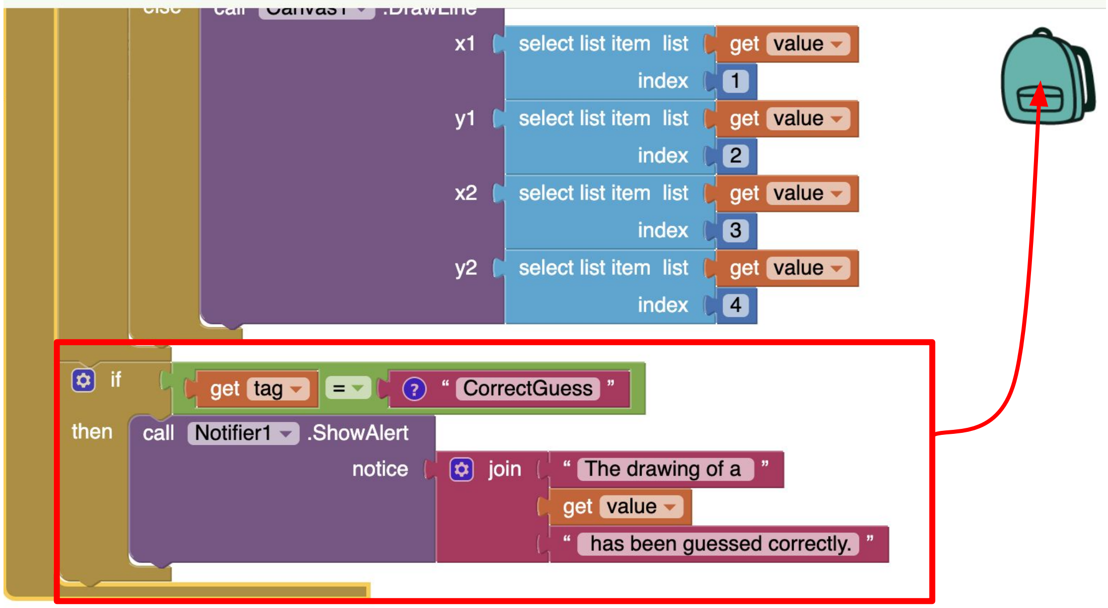
21 Add a **Notifier1.ShowAlert** block and use a join block to tell the user the correct guess has been made. Remember that value is the **currentDrawing**, so you can also tell others what the correct guess was.



```
if get tag = "CorrectGuess"
then
call Notifier1.ShowAlert
    notice join "The drawing of a "
                get value
                " has been guessed correctly."
```

The Sketcher should be notified too, and you will use the same blocks. Use the Backpack to copy blocks between Screens.

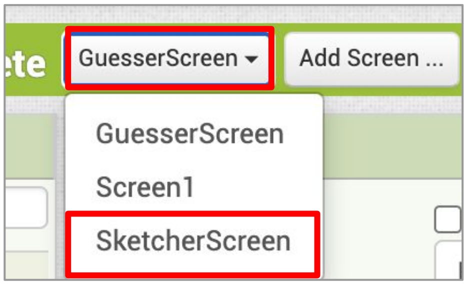
22 Drag the **if** block you just added to the backpack in the upper right corner of the screen. It will move the block out of **CloudDB1.DataChanged**, but you can just snap it back in.



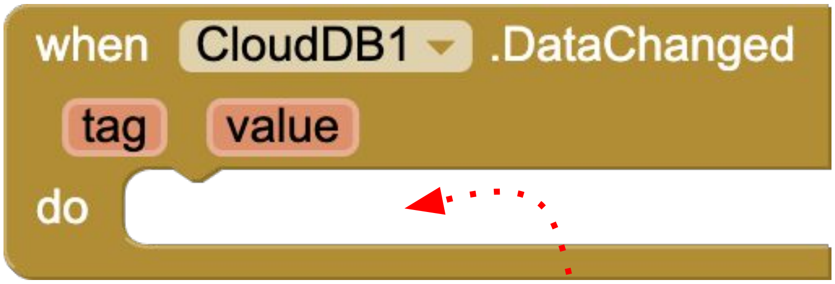
```
call Canvas1.DrawLine
    x1 select list item list get value
                    index 1
    y1 select list item list get value
                    index 2
    x2 select list item list get value
                    index 3
    y2 select list item list get value
                    index 4
if get tag = "CorrectGuess"
then call Notifier1.ShowAlert
        notice join "The drawing of a "
                    get value
                    " has been guessed correctly."
```

NOTIFY USERS (continued)

23 Switch to **SketcherScreen**.

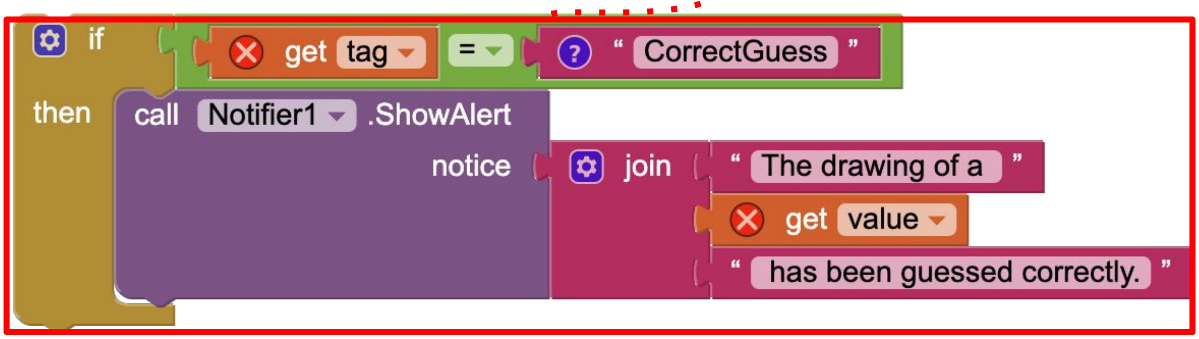


24 This screen does not have a **CloudDB1.DataChanged** event yet, so drag out that block.



```
when CloudDB1 ▾ .DataChanged
  tag  value
do
```

25 Click on the backpack in the upper right corner of the screen, and drag out the **if-then** block you just stored there, and snap it into **CloudDB1.DataChanged**.



```
if   get tag ▾ = ▾ " CorrectGuess "
then call Notifier1 ▾ .ShowAlert
                        notice  join  " The drawing of a "
                                      get value ▾
                                      " has been guessed correctly. "
```

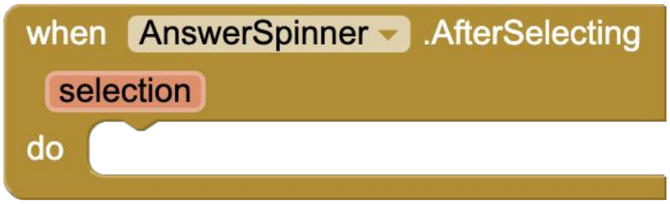
26 Final test with MIT AI2 Companion! Test that everyone is notified when a correct guess is made.

COMPUTATIONAL THINKING CONCEPTS

The following are the Computational Thinking Concepts learned in Part 4.

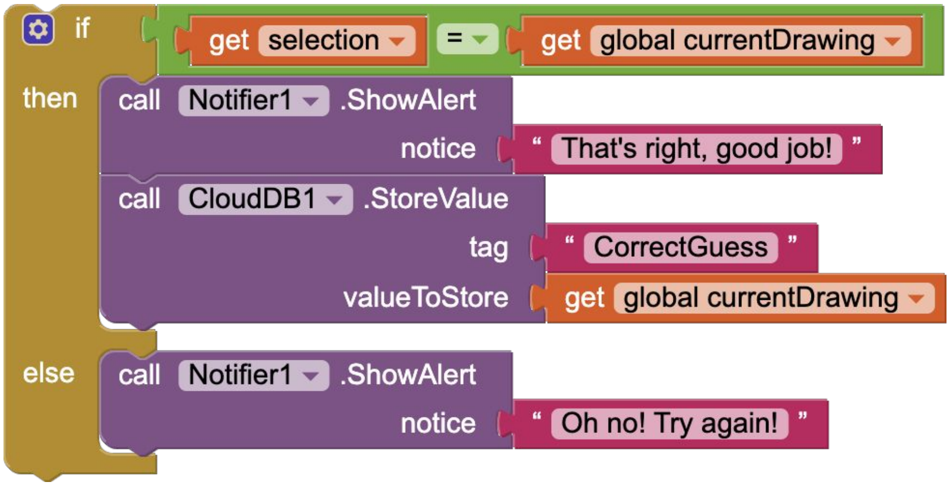
Sketch And Guess Part 4

1. Events



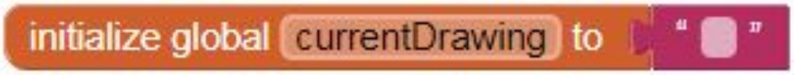
```
when AnswerSpinner .AfterSelecting
  selection
do
```

2. Conditionals



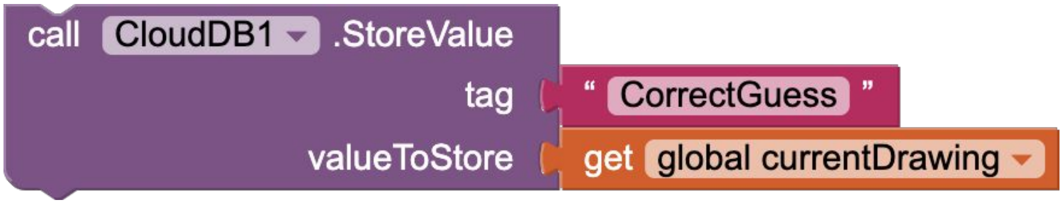
```
if   get selection = get global currentDrawing
then call Notifier1 .ShowAlert
                          notice " That's right, good job! "
     call CloudDB1 .StoreValue
                              tag " CorrectGuess "
                     valueToStore get global currentDrawing
else call Notifier1 .ShowAlert
                          notice " Oh no! Try again! "
```

3. Naming / variables



```
initialize global currentDrawing to " "
```

4. Manipulation of Data and Elementary Data Structures



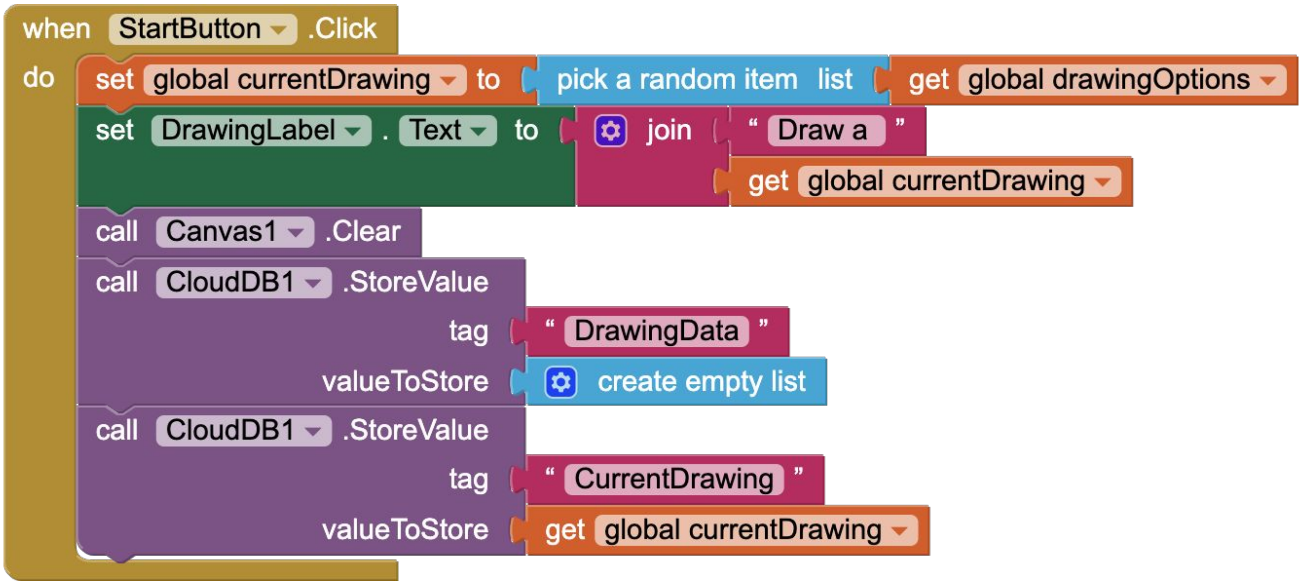
```
call CloudDB1 .StoreValue
                      tag " CorrectGuess "
             valueToStore get global currentDrawing
```

COMPUTATIONAL THINKING CONCEPTS

The following are the Computational Thinking Concepts learned in Part 4.

Sketch And Guess Part 4

5. Sequences



```
when StartButton .Click
do  set global currentDrawing to  pick a random item list  get global drawingOptions
    set DrawingLabel . Text to  join  " Draw a "
                                      get global currentDrawing
    call Canvas1 .Clear
    call CloudDB1 .StoreValue
                          tag  " DrawingData "
                 valueToStore  create empty list
    call CloudDB1 .StoreValue
                          tag  " CurrentDrawing "
                 valueToStore  get global currentDrawing
```