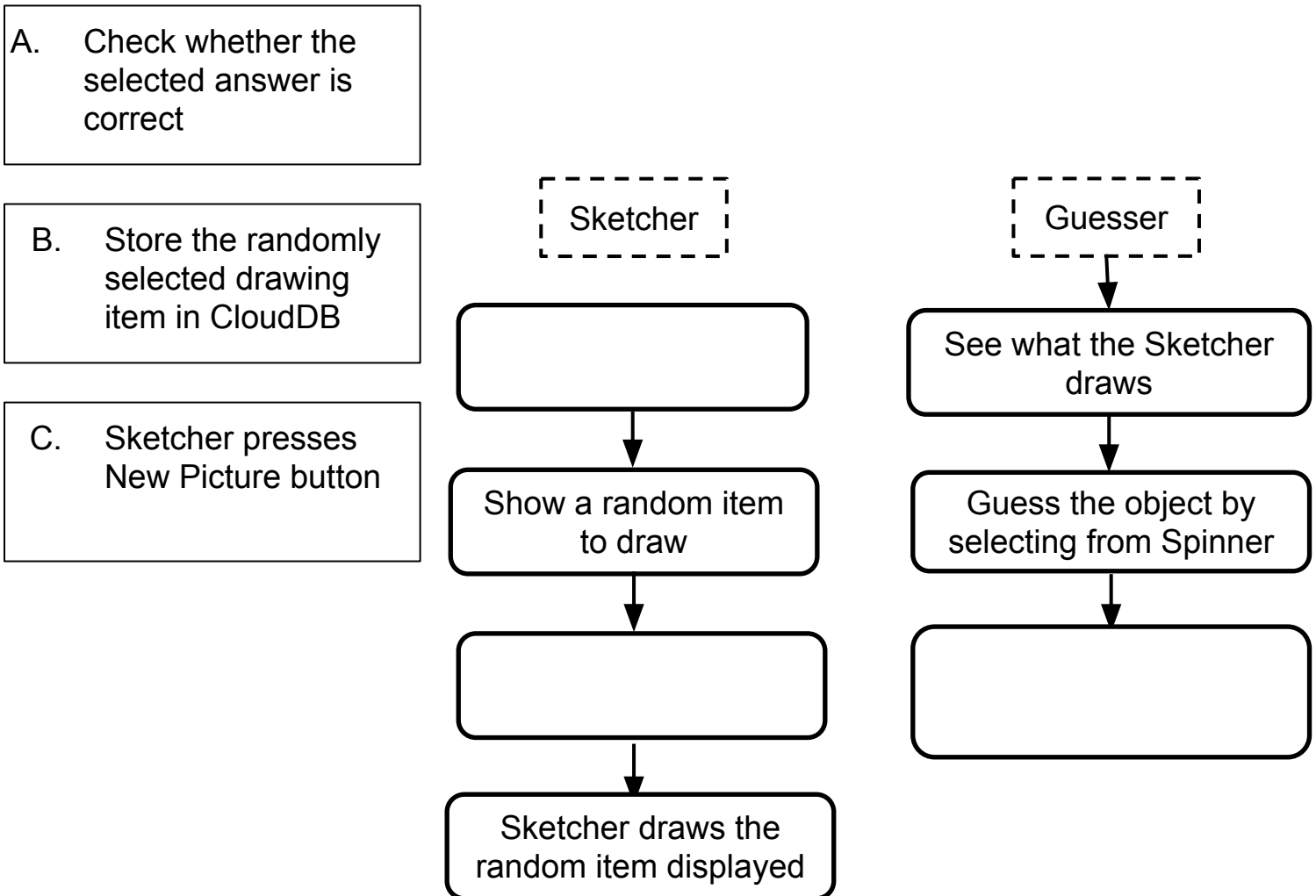# SKETCH & GUESS: PART 4

In Part 4, 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

↓

[ ]

MIT APP INVENTOR

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 Guessers 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

MIT
APP INVENTOR

MOVE VARIABLE BLOCKS USING BACKPACK

The GuesserScreen will need the **currentDrawing** and **drawingOptions** variables you made for SketcherScreen. There is an easy way to duplicate those blocks for GuesserScreen.
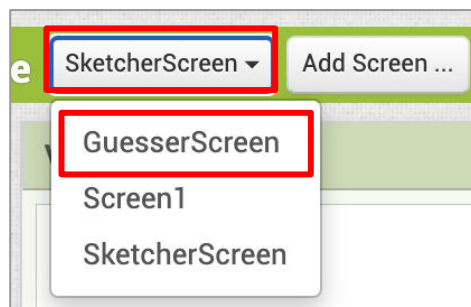
1   Make sure you are in the **SketcherScreen** and in the **Blocks Editor**.
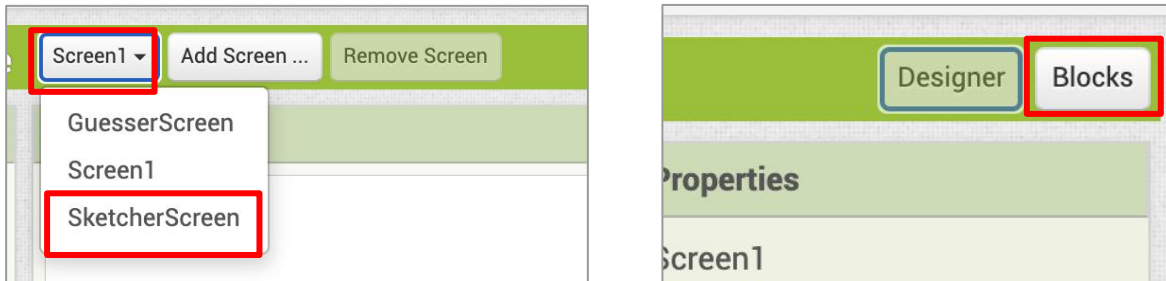
2   You want to include the **drawingOptions** list and the **currentDrawing** variable in the GuesserScreen. An easy way to do this is to drag these blocks into the Backpack in the upper right corner of the SketcherScreen.
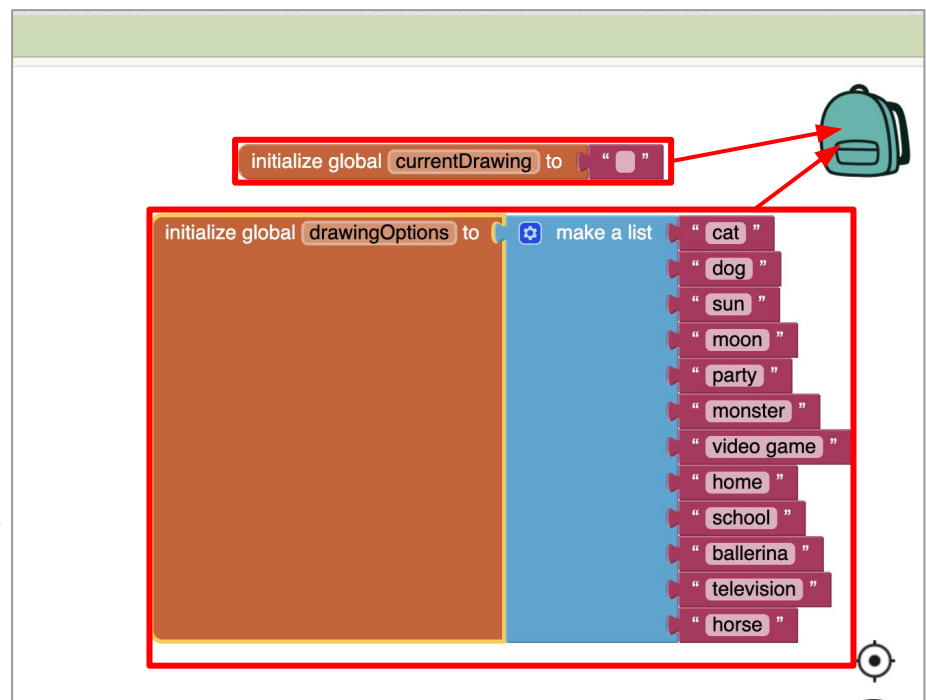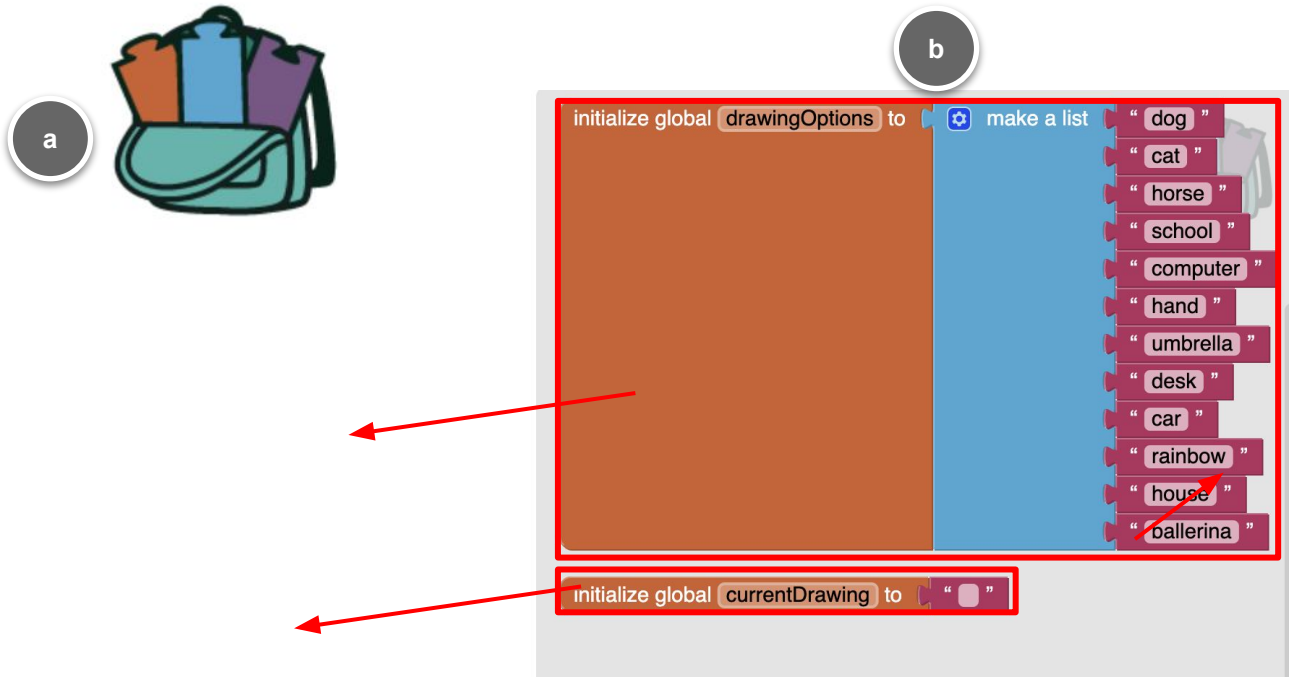
3   Switch to GuesserScreen.

MIT
APP INVENTOR

Copyright MIT 2021

MOVE BLOCKS USING BACKPACK (continued)

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

a

b

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.

te GuesserScreen ▾   Add Screen ...

GuesserScreen
Screen1
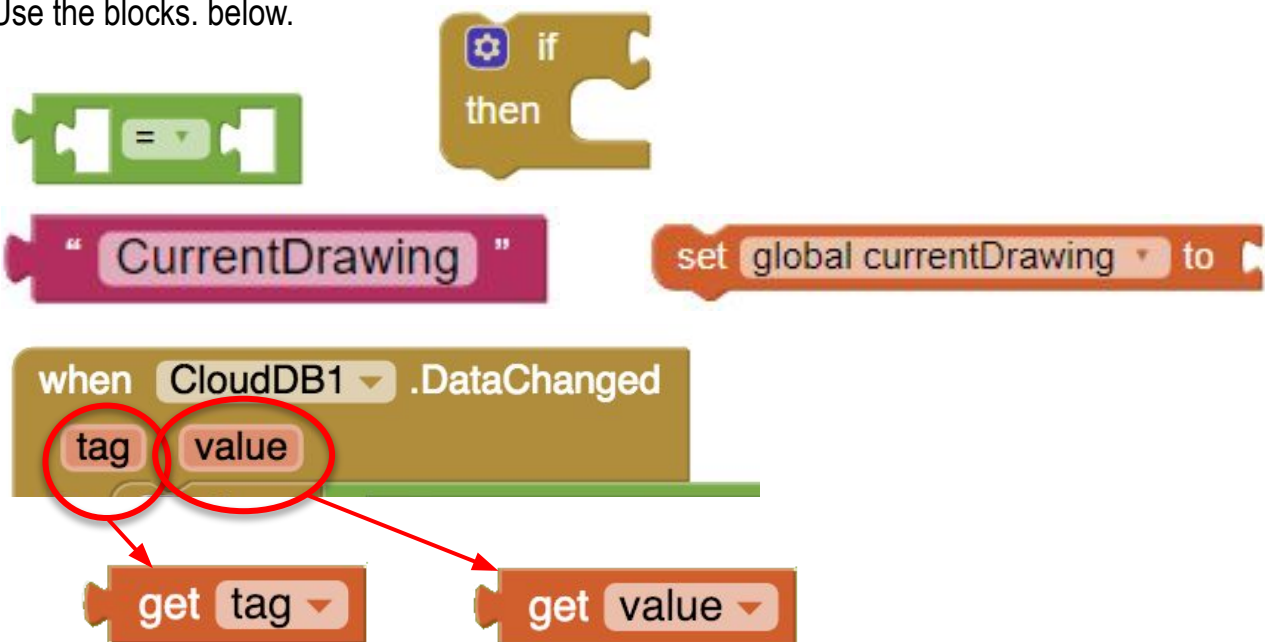SketcherScreen
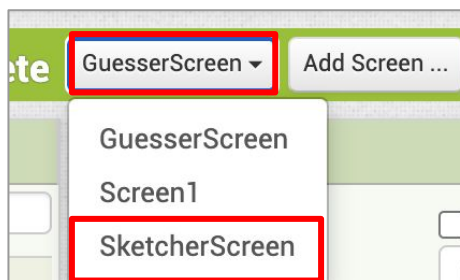
STORE AND RETRIEVE CURRENT DRAWING

When the Sketcher clicks the New Picture button, they are given a randomly choses drawing to draw. The GuesserScreen needs to know what the Sketcher is drawing, so when the Guesser makes a guess, the app can check if they are correct or not.

6 In SketcherScreen, after the random item is selected from drawingOptions, store that currentDrawing in CloudDB. Use these blocks.

```
call CloudDB1 ▾ .StoreValue
                          tag
                  valueToStore
```

```
when StartButton ▾ .Click
```

```
" CurrentDrawing "
```

```
get global currentDrawing ▾
```

7 In GuesserScreen, retrieve the currentDrawing from CloudDB, and store it in the currentDrawing variable to use later to determine if a guess is correct or not.

Use the blocks. below.

```
⬚ = ▾ ⬚
```

```
⚙ if
then
```

```
" CurrentDrawing "
```

```
set global currentDrawing ▾ to
```

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

```
get tag ▾
```

```
get value ▾
```

CHECK FOR A CORRECT GUESS

8   **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** when that screen first opens. Use the following blocks.
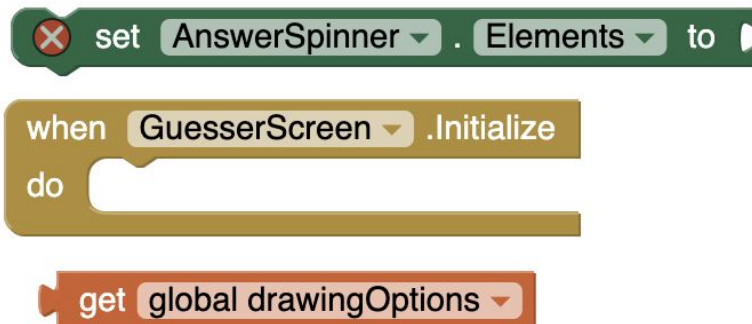
9   Check if the answer that the Guesser selects from the **AnswerSpinner** is correct, and notify the user. Use the blocks below.
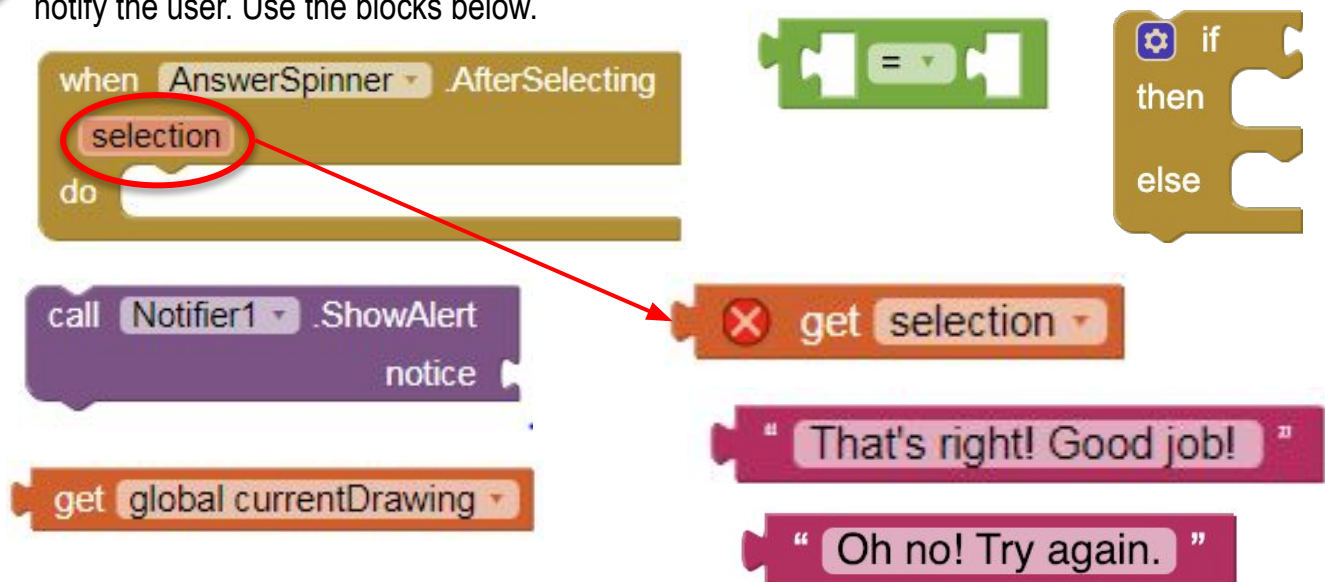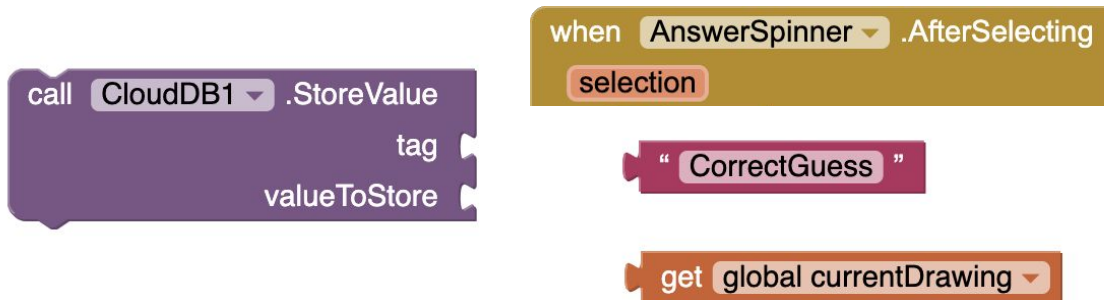

10   Test with your partner. Build the apk using the QR code option, scan the QR code and download and install the apk on your individual devices. One person sketch and the other guess.


Copyright MIT 2021

6
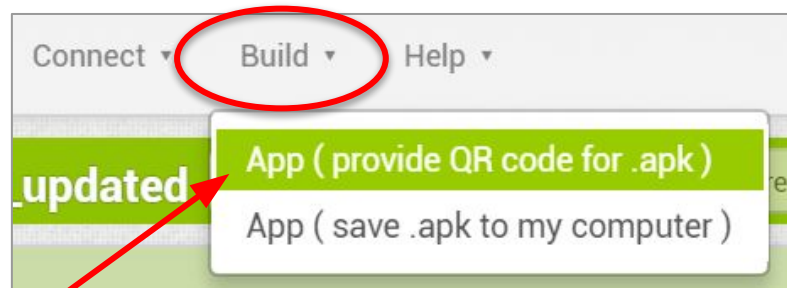
UPDATE ALL USERS WITH A CORRECT GUESS

11 The Guesser who guesses correctly should store that information in CloudDB, which in turn will send that information out to all other users. Update **AnswerSpinner.AfterSelecting** to store a correct guess in CloudDB. Use the following blocks.
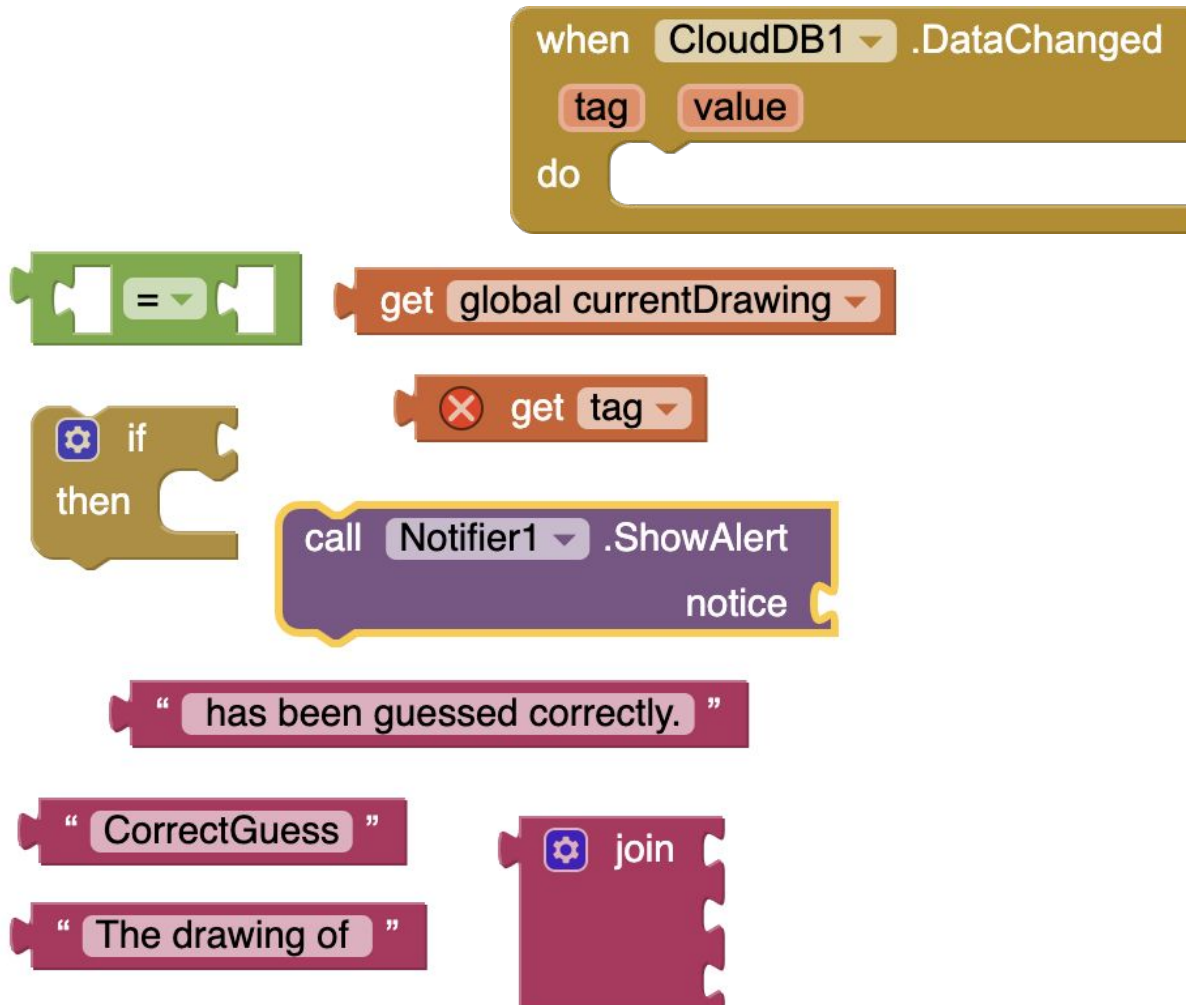

12 Add a new **if** block to the **DataChanged** event to look for a CorrectGuess tag, and if found, notify the user that the correct guess has occurred. This covers a game played with multiple guessers. Use the following blocks.
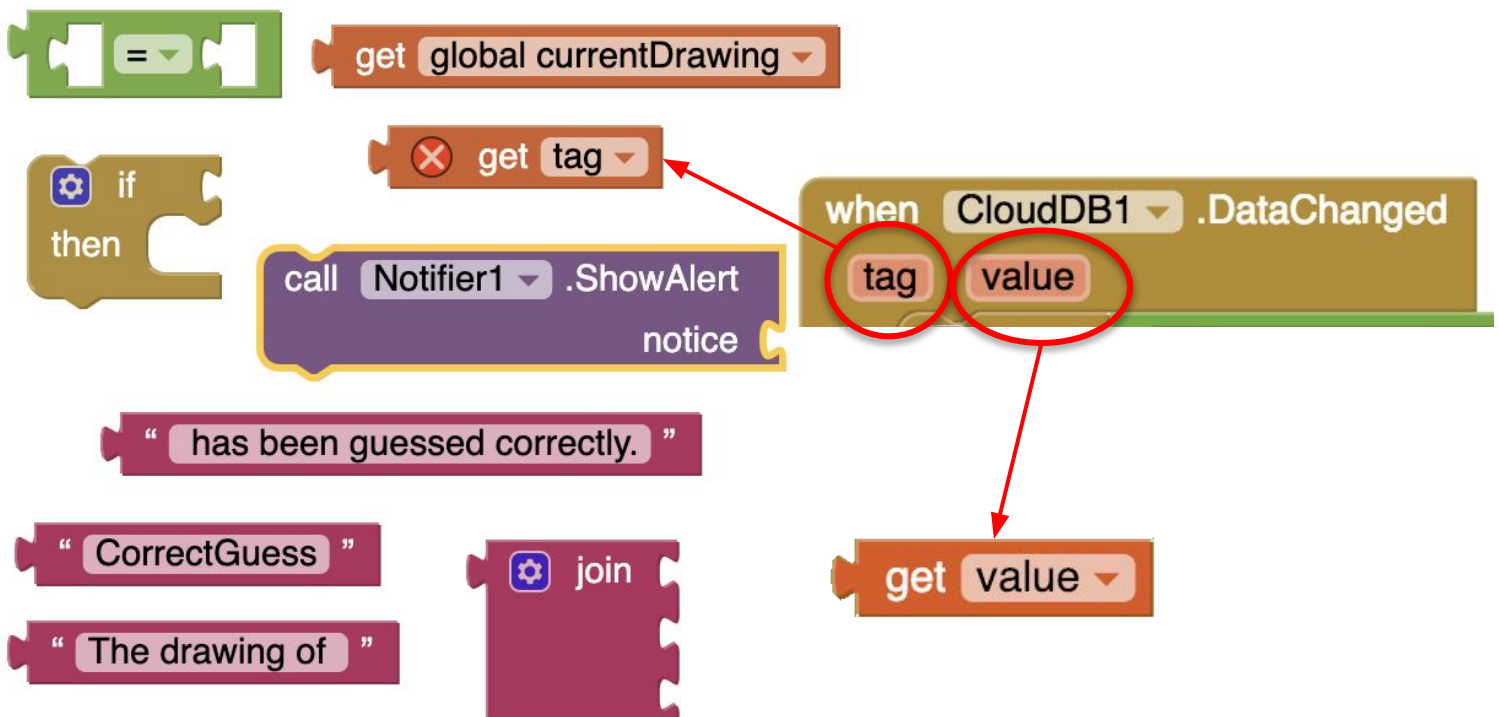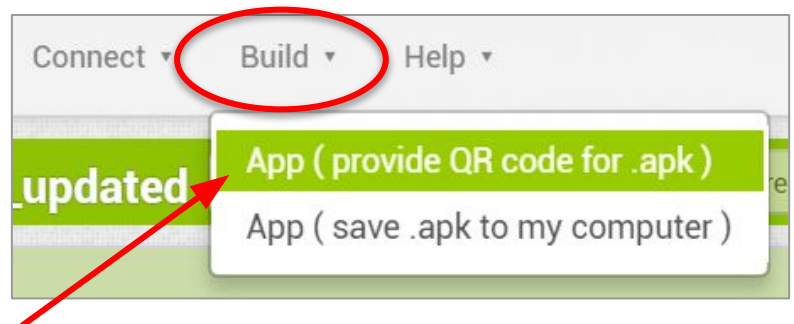
UPDATE SKETCHER

13 The Sketcher also needs to know someone guessed the drawing correctly, so add a **CloudDB1.DataChanged** event to SketcherScreen, and create the same code as you just did for GuesserScreen. Use these blocks.

when **CloudDB1** .DataChanged
tag   value
do

= ▾   get **global currentDrawing** ▾

⚙ if   ⊗ get **tag** ▾
then
          call **Notifier1** ▾ .ShowAlert
                              notice

" **has been guessed correctly.** "

" **CorrectGuess** "   ⚙ join

" **The drawing of** "

14 Test again with your partner. Build the apk using the QR code option, scan the QR code and download and install the apk on your individual devices. One person sketch and the other guess. Check that all users get a notification upon a correct guess.

Connect ▾   Build ▾   Help ▾

updated

App (provide QR code for .apk)

App (save .apk to my computer)

MIT
APP INVENTOR

Choose Ways to Extend Your App

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

Add TextToSpeech to speak what is to be drawn

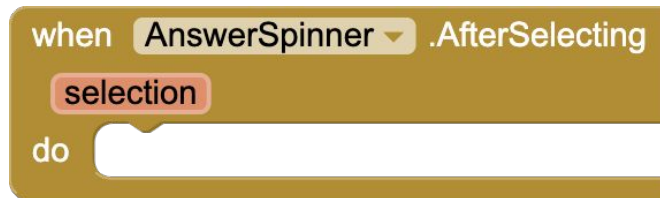Send users back to Screen1 on a correct guess to restart the game

Keep score! Each player can keep track of their correct guesses!
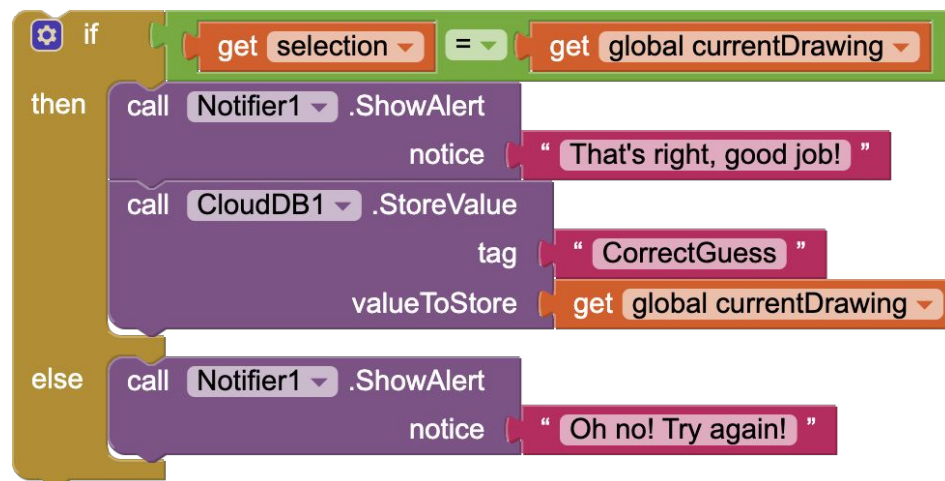
What other ideas do you have?

MIT
APP INVENTOR

COMPUTATIONAL THINKING CONCEPTS

Sketch And Guess Part 4
1. Events
2. Conditionals
3. Naming / variables
4. Manipulation of Data and Elementary Data Structures

COMPUTATIONAL THINKING CONCEPTS

Sketch And Guess Part 4
5. Sequences