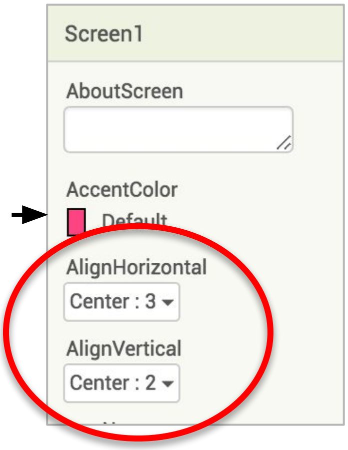
TWO-BUTTON GAME: PART 1

In this lesson, you will remake the two button game from Unit 1. Users click the buttons as fast as they can!

START HERE

1 Open a new project in MIT App Inventor and name the project “TwoButtonGame”.

2 Set *AlignHorizontal* and *AlignVertical* for **Screen1** to **Center**.



Screen1

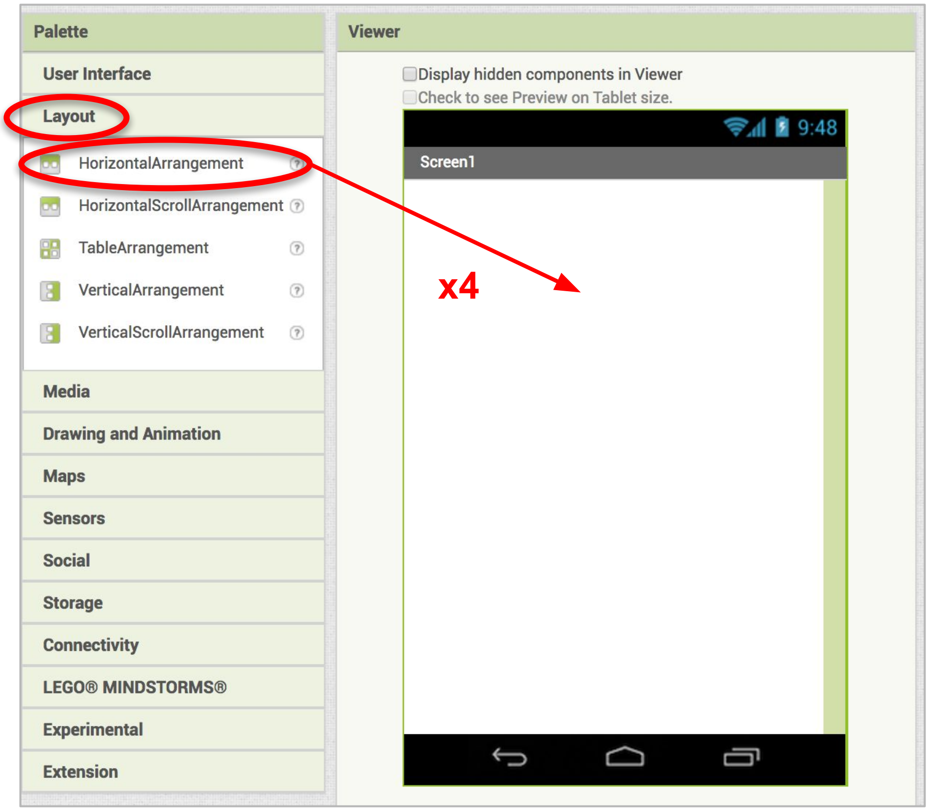
AboutScreen

AccentColor: Default

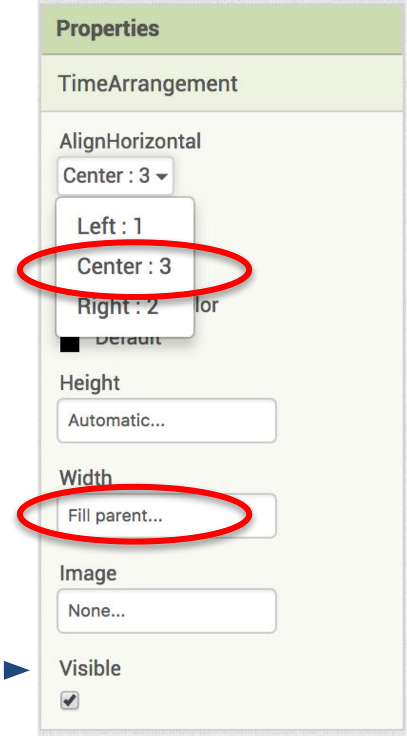
AlignHorizontal: Center : 3

AlignVertical: Center : 2

3 Drag in 4 **HorizontalArrangements**. Name them *TimeArrangement, ButtonArrangement, ScoreArrangement,* and *StartButtonArrangement*.



4 For each **HorizontalArrangement**, change its *AlignHorizontal* property to “**Center**” and its *Width* property to “**Fill Parent**”.



Properties

TimeArrangement

AlignHorizontal: Center : 3

Height: Automatic...

Width: Fill parent...

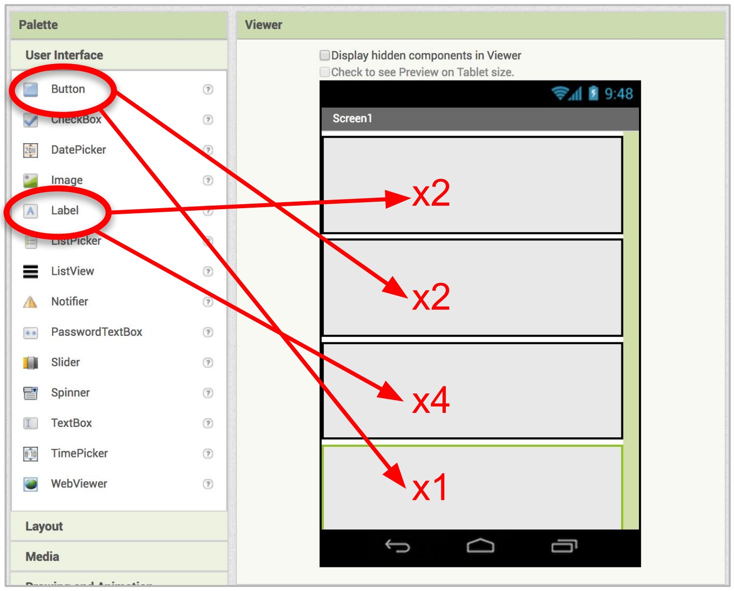
Image: None...

Visible: ☑

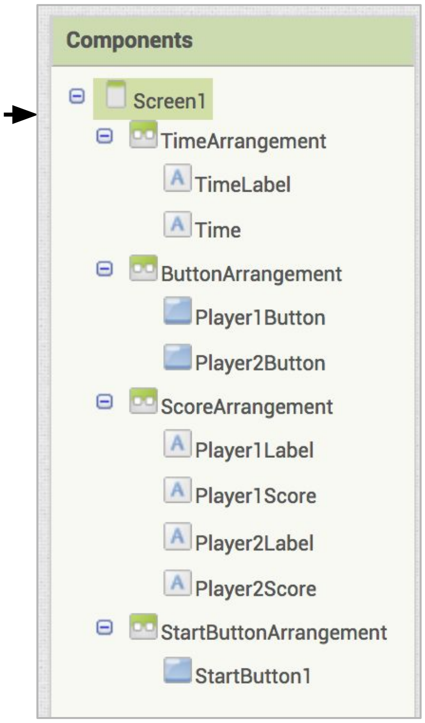
ADDING COMPONENTS

5 Drag in the following components to the corresponding HorizontalArrangement:

- TimerArrangement - 2 labels
- ButtonArrangement - 2 buttons
- ScoreArrangement - 4 labels
- StartButtonArrangement - 1 button



6 Name each new component as shown below.

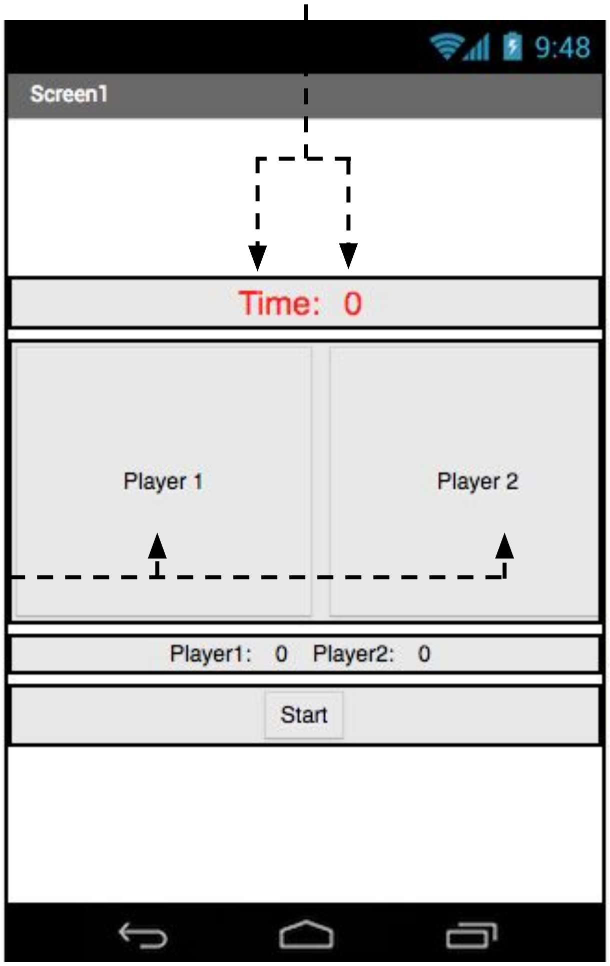


Components

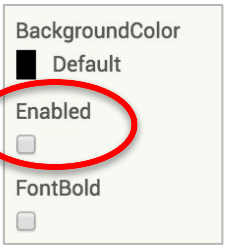
- Screen1
  - TimeArrangement
    - TimeLabel
    - Time
  - ButtonArrangement
    - Player1Button
    - Player2Button
  - ScoreArrangement
    - Player1Label
    - Player1Score
    - Player2Label
    - Player2Score
  - StartButtonArrangement
    - StartButton1

7 Change the *TextColor* for these labels to **Red** and the *FontSize* to **20**.

8 Set the *Text* property of each component so the final User Interface looks like this.



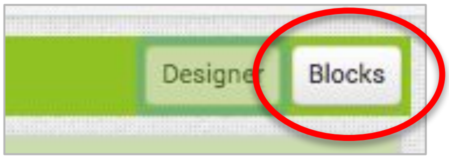
9 Set the *Height* property for these two buttons to **150 pixels**, and the *Width* property to **50 percent**. Then uncheck *Enabled* for both.



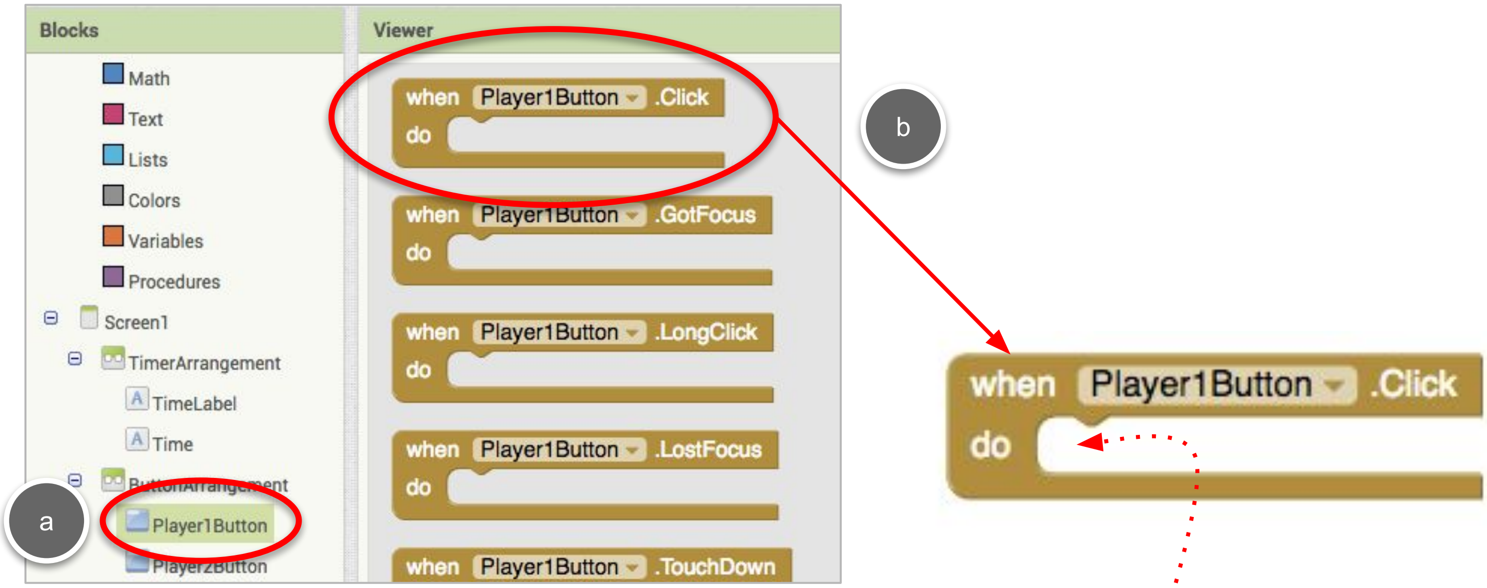
BackgroundColor
■ Default
Enabled
□
FontBold
□

BLOCKS

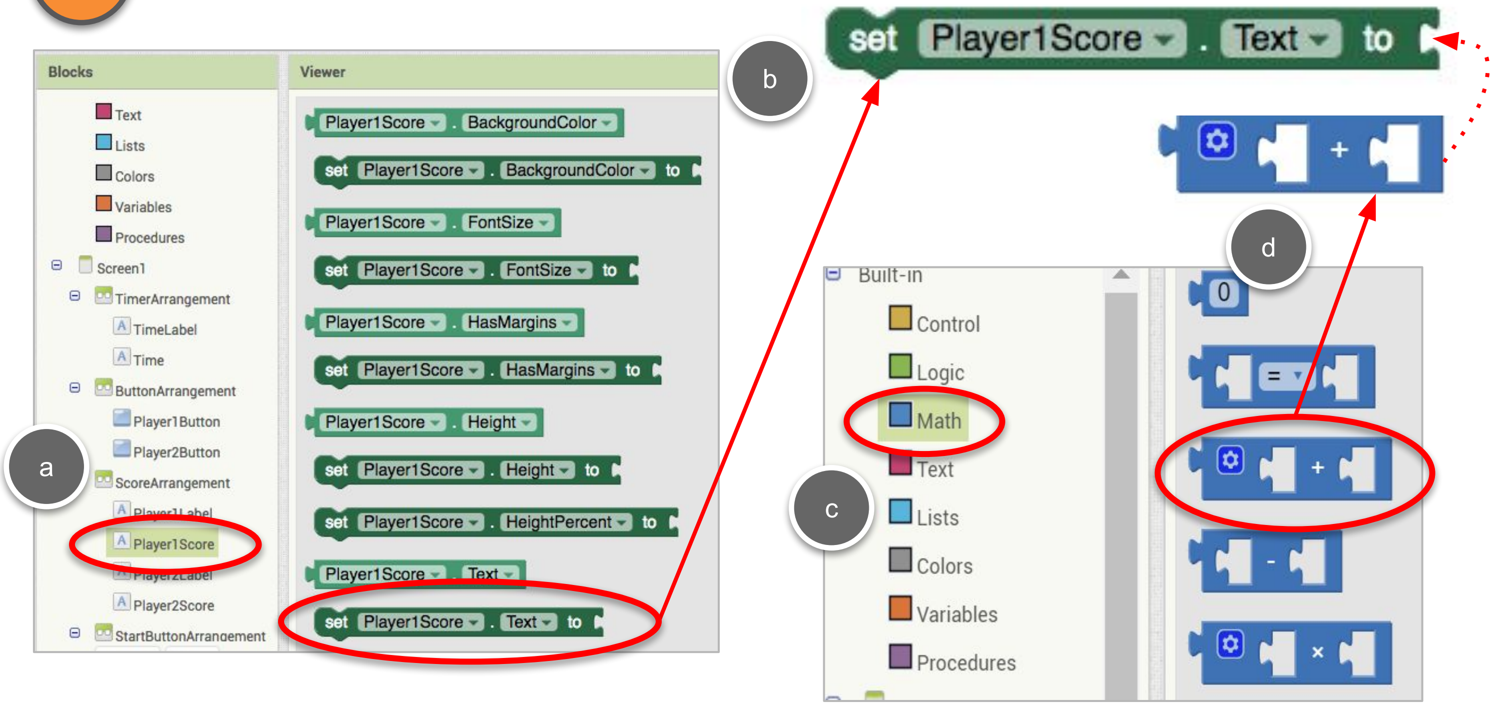
10 Switch to the Blocks editor from the Designer.



11 Drag out a **Player1Button.Click** event block.

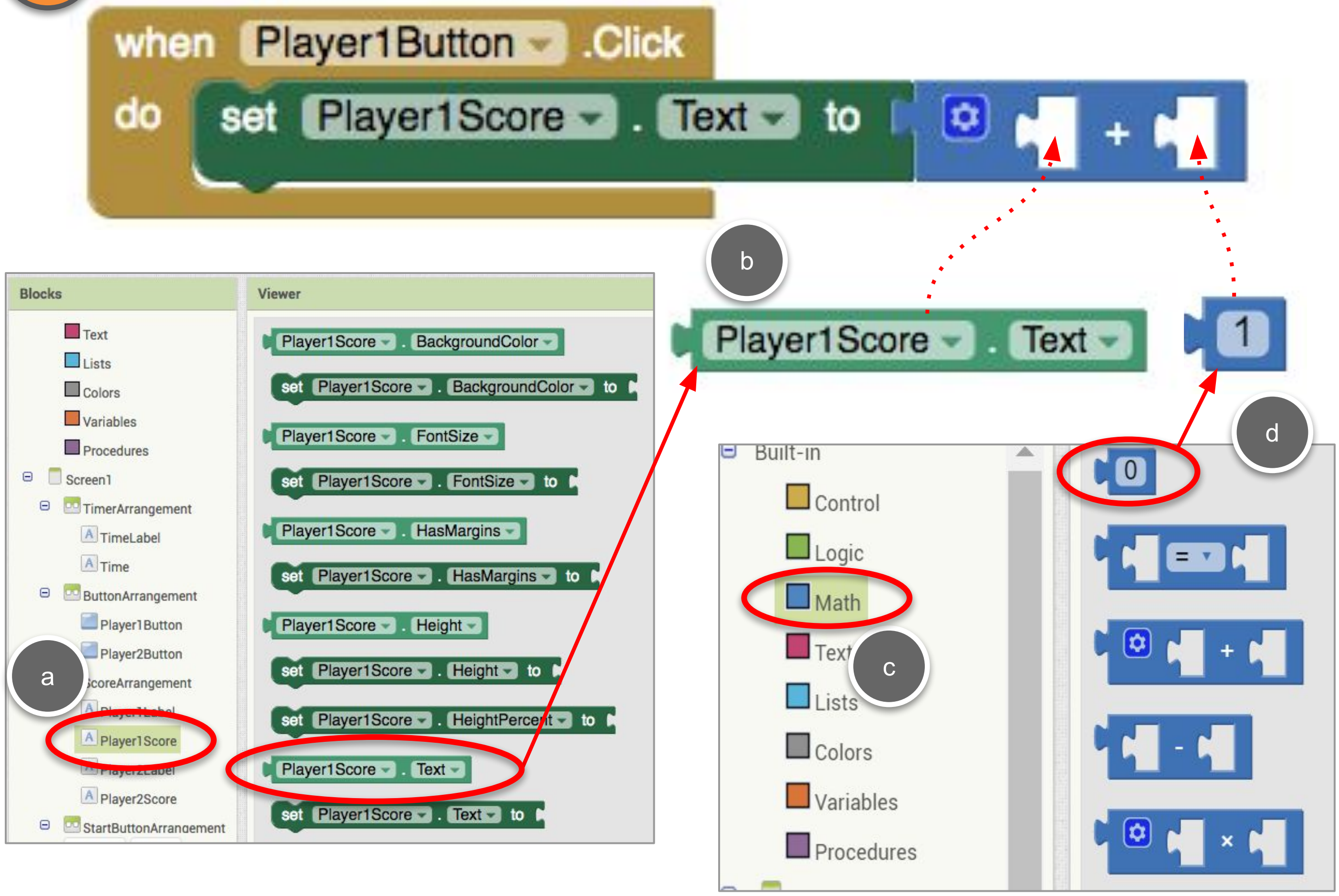

Blocks
Math
Text
Lists
Colors
Variables
Procedures
Screen1
TimerArrangement
TimeLabel
Time
ButtonArrangement
Player1Button
Player2Button
Viewer
when Player1Button .Click
do
when Player1Button .GotFocus
do
when Player1Button .LongClick
do
when Player1Button .LostFocus
do
when Player1Button .TouchDown
a
b
when Player1Button .Click
do

12 Increase Player1's score by 1.



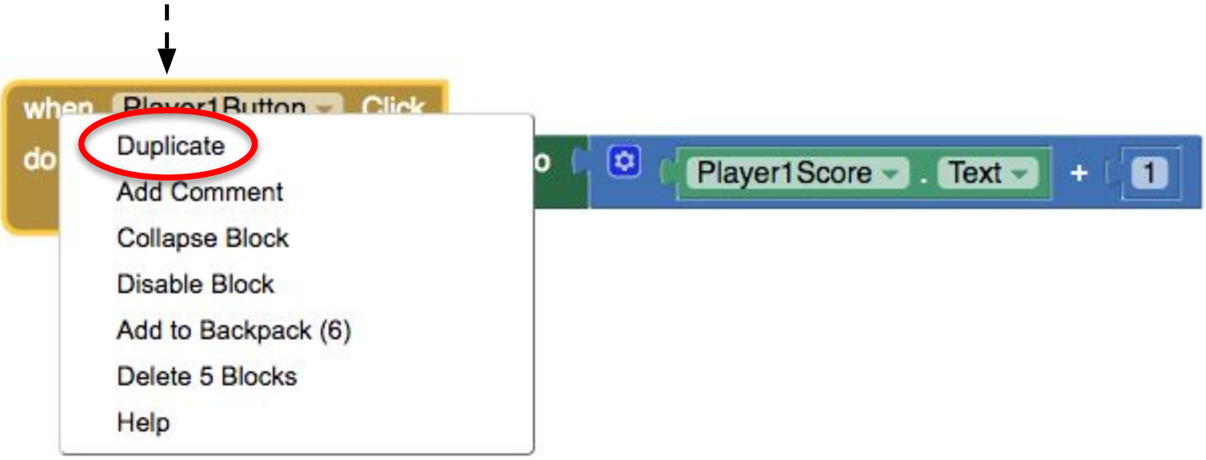
INCREASE SCORE

13 Fill in the addition block.



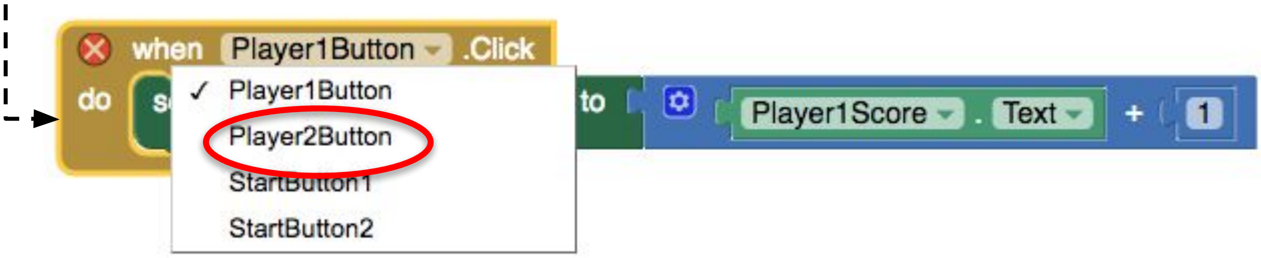
Now, code the **Player2Button** by duplicating the **Player1Button.Click** event and changing it slightly.

14 Right-click on the **when Player1Button.Click** event block, and click **Duplicate.**

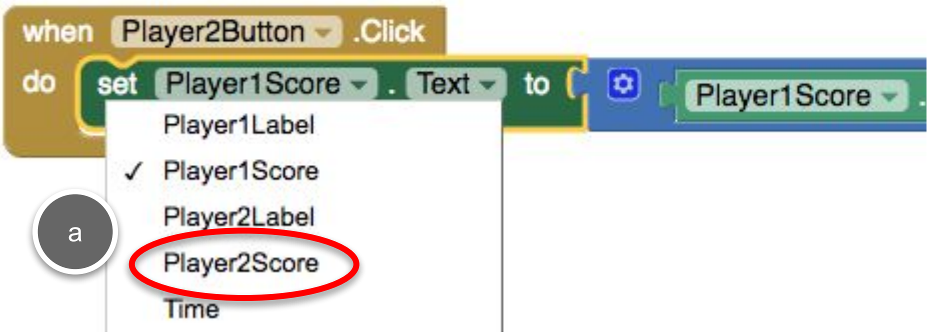


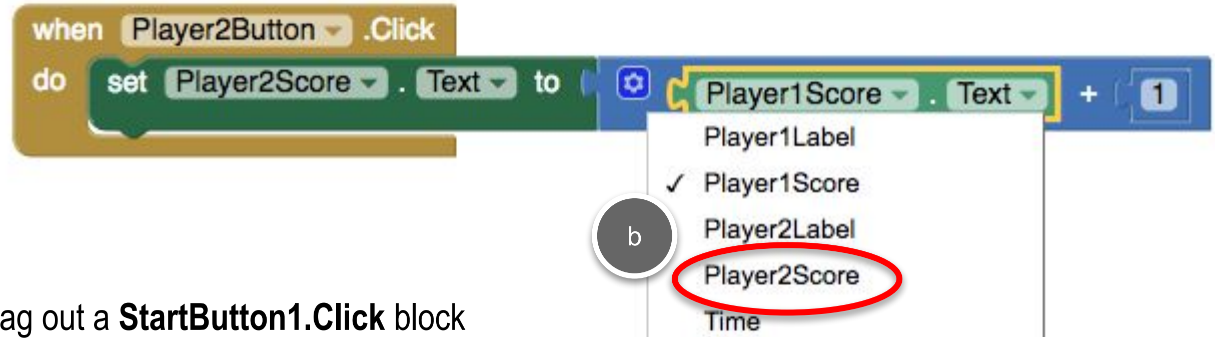
PLAYER2 BUTTON

15 Click the dropdown menu of **Player1Button**, and select **Player2Button.**

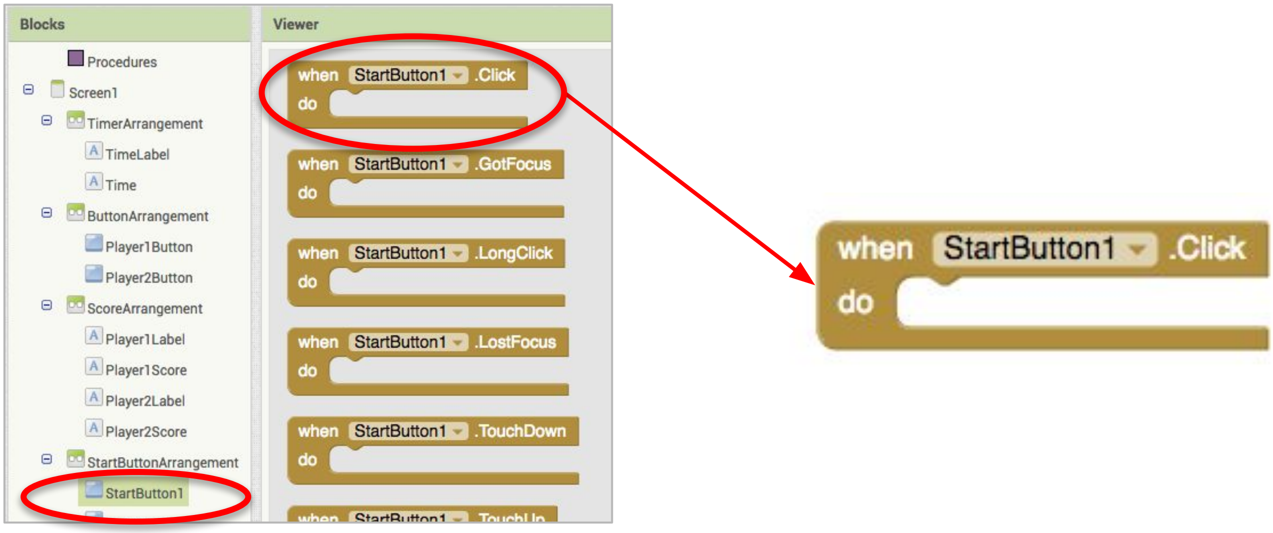


16 Click the dropdown menu of **Player1Score**, and click **Player2Score.**



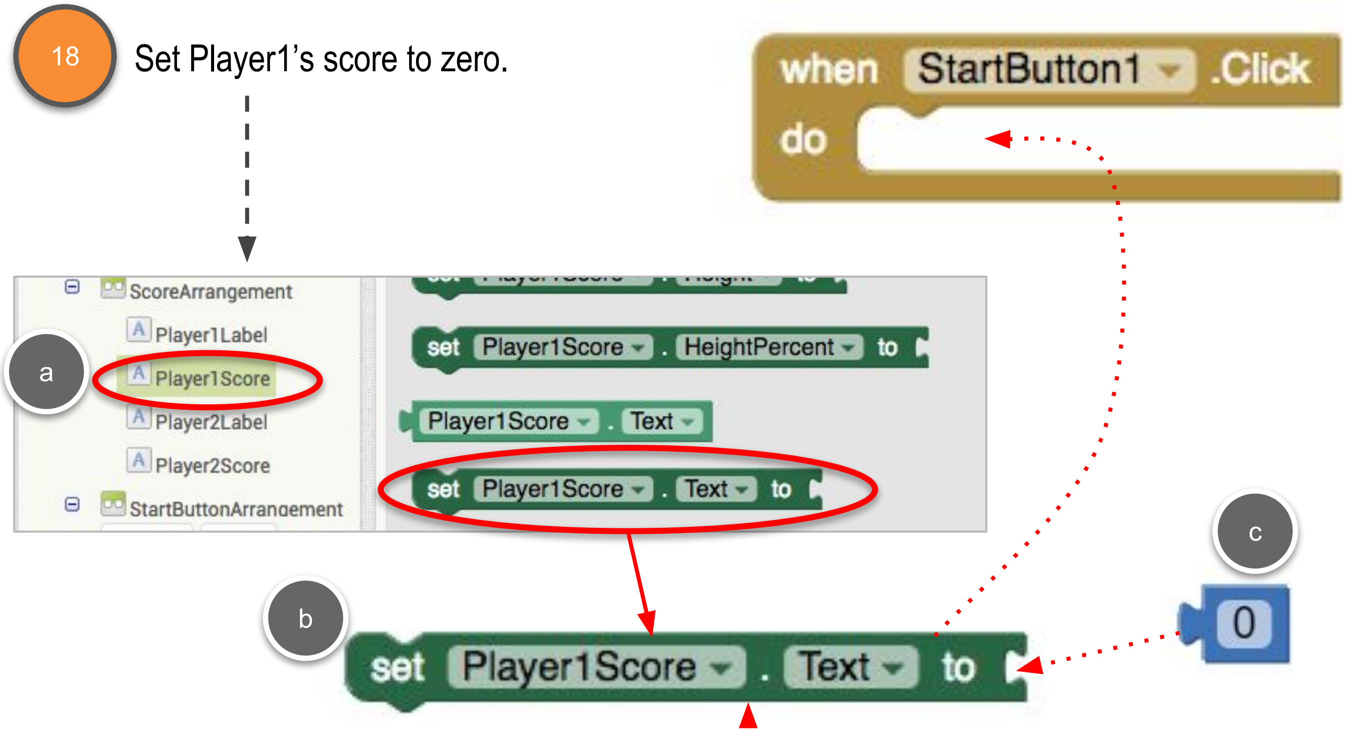


17 Drag out a **StartButton1.Click** block

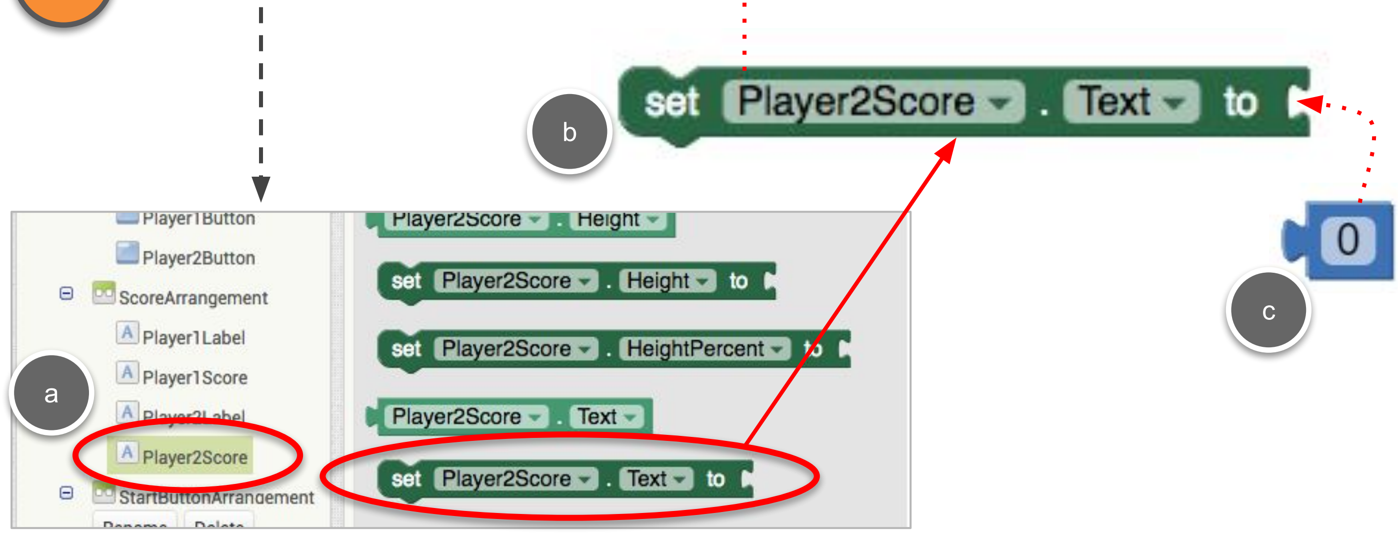


START BUTTON

18 Set Player1's score to zero.



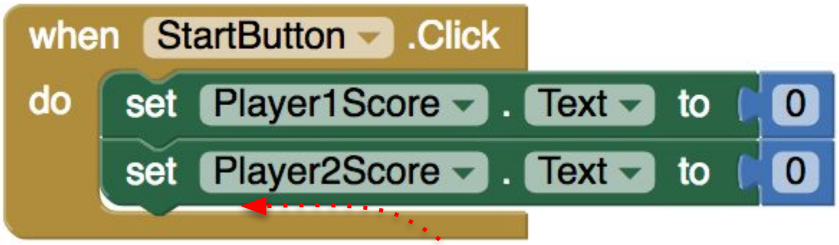
19 Do the same for Player2.



MIT APP INVENTOR

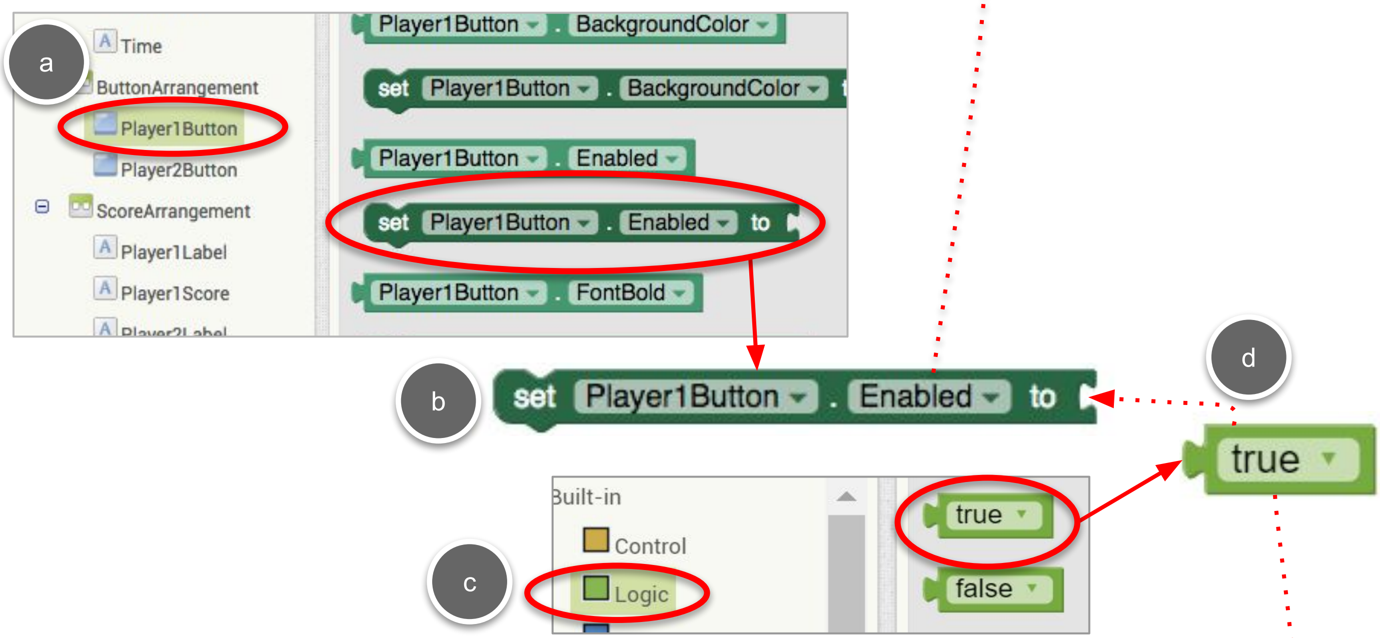
Copyright MIT 2019, Review Copy, Not for Distribution

START BUTTON (continued)

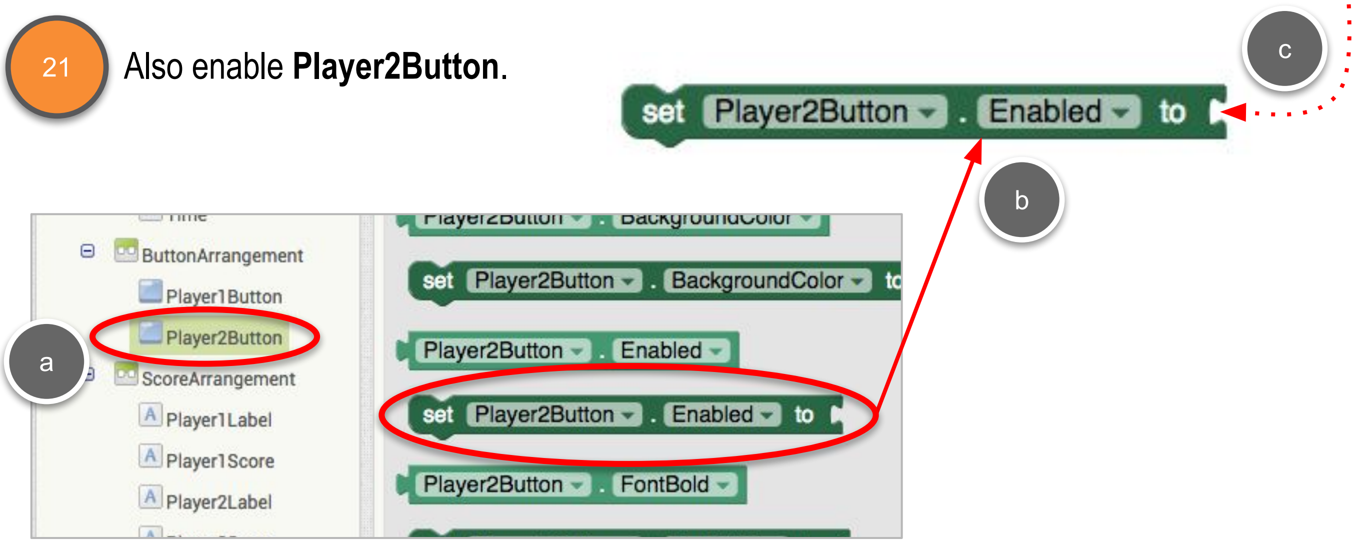


```
when StartButton .Click
do  set Player1Score . Text to 0
    set Player2Score . Text to 0
```

20 Now enable **Player1Button** so users can click on it.

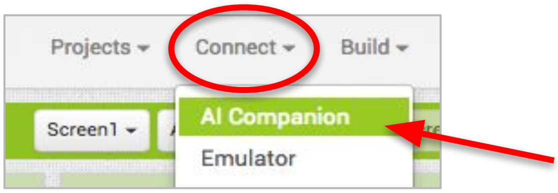


21 Also enable **Player2Button**.



TESTING!

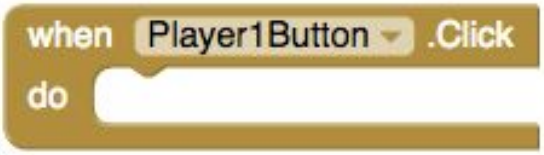
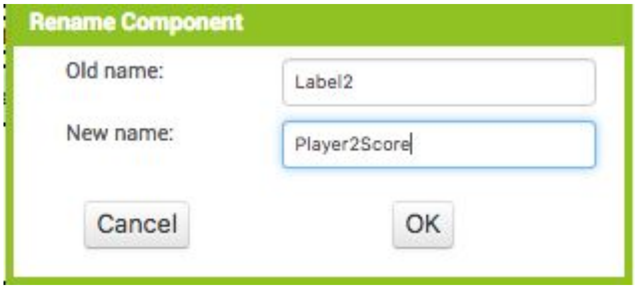
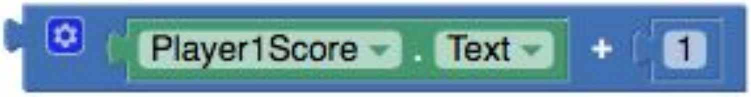
22 Now test your app by connecting to the MIT AI2 Companion. Start the game and press both buttons. Do the scores update correctly?



TWO-BUTTON GAME: PART 1

COMPUTATIONAL THINKING CONCEPTS

The following are the Computational Thinking Concepts learned in this lesson.

Two-Button Game	
1. Events:	 when Player1Button .Click do
3. Naming:	 Rename Component Old name: Label2 New name: Player2Score
4. Operators:	 Player1Score . Text + 1