

# MY PIANO: CHALLENGE



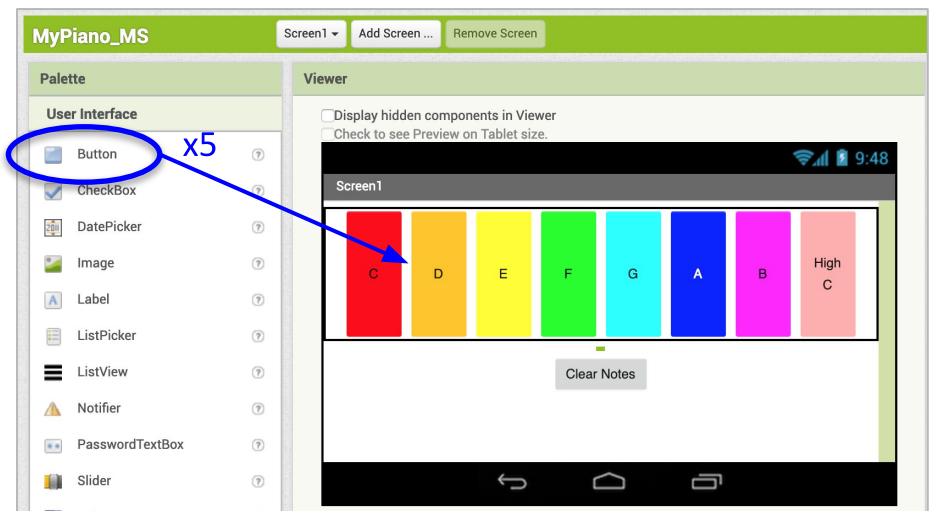
Thanks for trying some of the Challenges. Try one, two, or all of the suggestions, or come up with your own!

## ADD SHARP NOTE BUTTONS

1 Switch to the Designer.



2 Add 5 more Buttons for the 5 Sharp Notes (C, D, F, G, and A). Remember to name them CSharpButton, DSharpButton, etc) so the sound file works properly.



3 Since you added 5 new Buttons, you have a total of 13. If you want all the *Width*'s to add up to 100%, what percentage should each **Button Width** be? You can round down to the nearest whole number.

## ADD SHARP NOTES (continued)

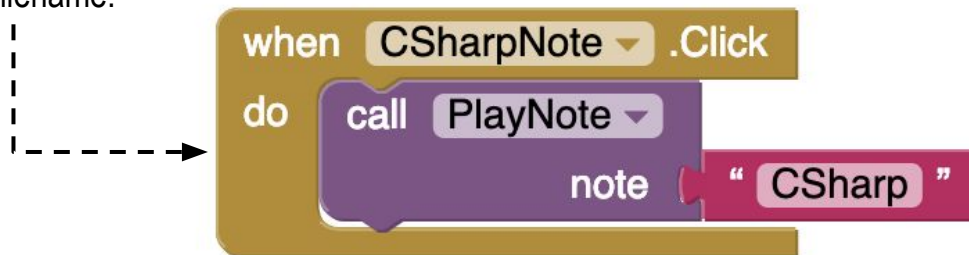
4

Switch to the Blocks Editor. ----->



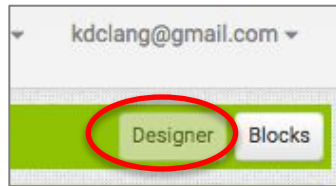
5

Add Button.Click event blocks for all you new Buttons. Remember to set the note parameter to match the Button name, since that matches the sound filename.



## MAKE THE BUTTONS LOOK LIKE A PIANO

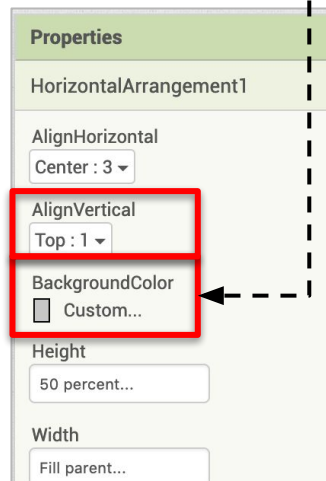
1 Switch to the Designer.



2 Changing the key color is really easy! Change all the regular buttons to a white *BackgroundColor*, and all the Sharp buttons to a black *BackgroundColor*. You will have to change the *TextColor* for the black buttons to white so they appear on the black background.

3 To make the white buttons show up, change **HorizontalArrangement1's** *BackgroundColor* to a light grey (or some other color).

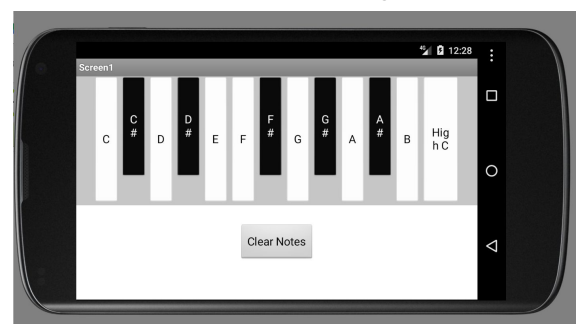
4 And change the *AlignVertical* property to "Top: 1" for **HorizontalArrangement1**.



5 You could make the Sharp Note **Buttons** not quite as tall as the regular notes. 40% is a good option, but you can try different values to see what you like.



Should look something like this!

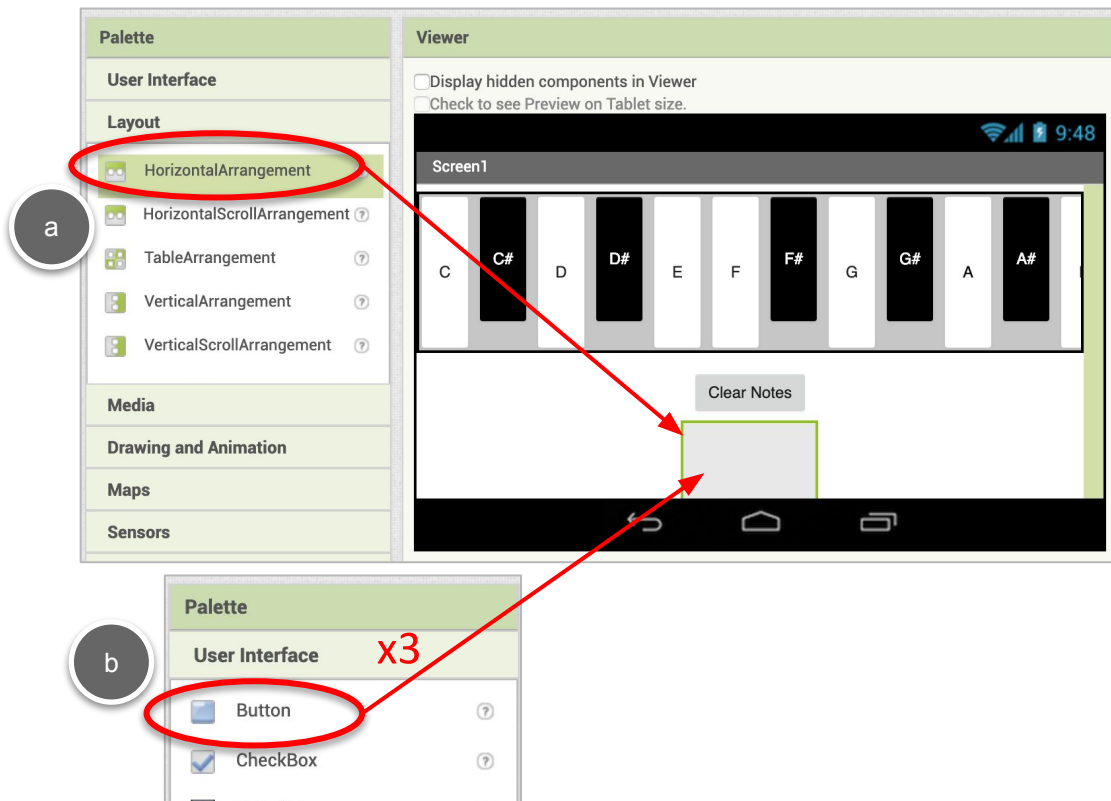


## RECORD YOUR MUSIC

1 Switch to the Designer.

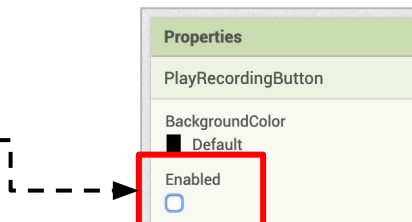


2 Add a **HorizontalArrangement** to the Viewer, and drop 3 **Buttons** into the **HorizontalArrangement**.

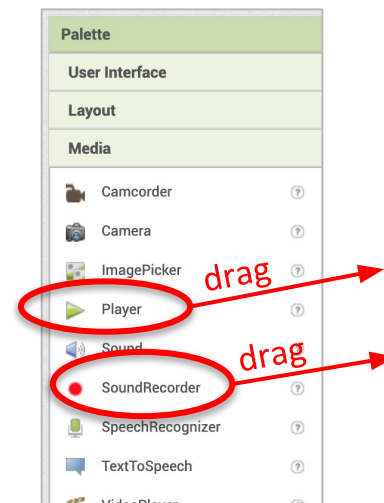


3 Name them **RecordButton**, **StopRecordingButton**, and **PlayRecordingButton**, in that order and change the *Text* property for each to its purpose.

4 Uncheck the *Enabled* box for the **StopRecordingButton** and the **PlayRecordingButton**.



5 Drag in a **SoundRecorder** component and another **Player** component from the Media drawer. Rename the Player component **RecordingPlayer**.



RECORD YOUR MUSIC (continued)

6

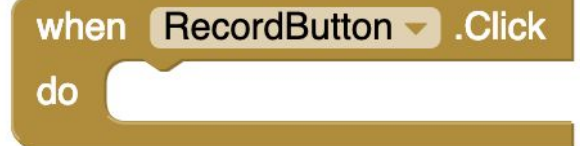
Switch to the Blocks Editor.



7

You will need to write code blocks for the 3 new Buttons:

- **RecordButton**
  - Start the **SoundRecorder**
  - Enable the **StopButton**
  - Disable **RecordButton** and **PlayRecordingButton**
- **StopRecordingButton**
  - Stop the **SoundRecorder**
  - Enable the **RecordButton**
  - Disable the **StopRecordingButton**
- **PlayRecordingButton**
  - Start the **RecordingPlayer**



8

When the **SoundRecorder** finishes recording (after it stops), it triggers the **AfterSoundRecorded** event. You will need to code this event too, setting the **RecordingPlayer.Source** to the returned sound. You should also enable the **PlayRecordingButton** in this event.



All the blocks needed are shown.

