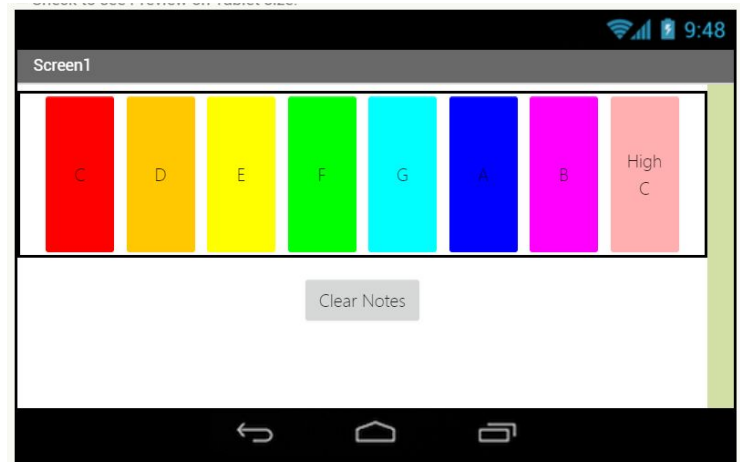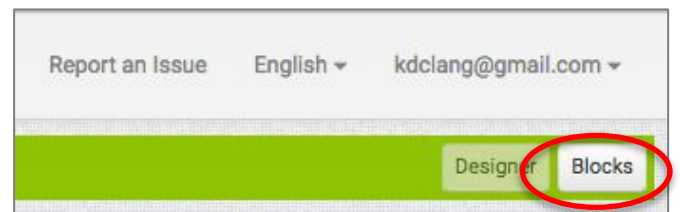# MY PIANO: PART 3

In this activity, you will learn how to use a procedure to manage the blocks of all the piano keys.

START HERE
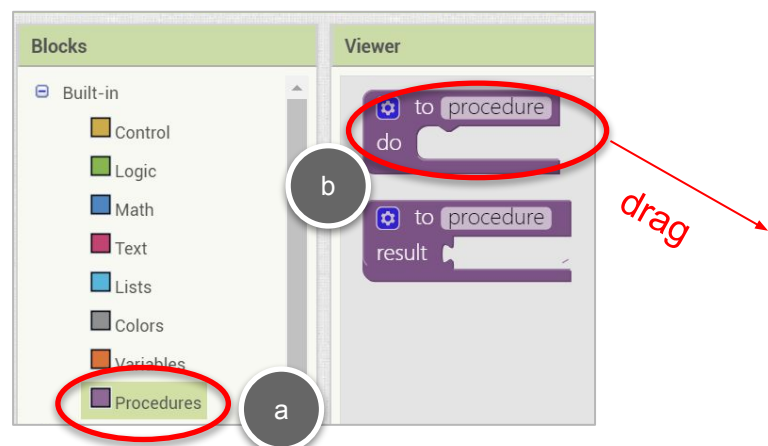
1 Go to the MIT App Inventor website (http://ai2.appinventor.mit.edu) and click the **Blocks** button to go to the Blocks Editor.
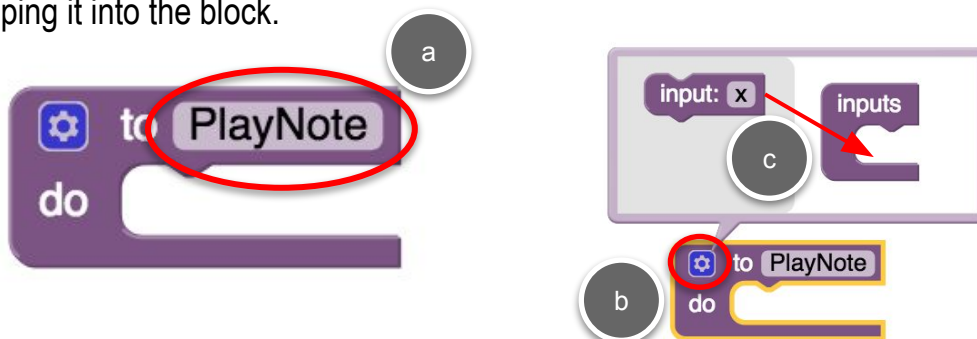
Because the code blocks for **CNote.Click** and **DNote.Click** are so similar, we are going to make a procedure to play the notes.

**2** Click on **to procedure** in the **Procedures** Drawer, then drag out a **to procedure** block.

3 Change the name to **PlayNote**. Add an input by clicking on the blue circle and snapping it into the block.
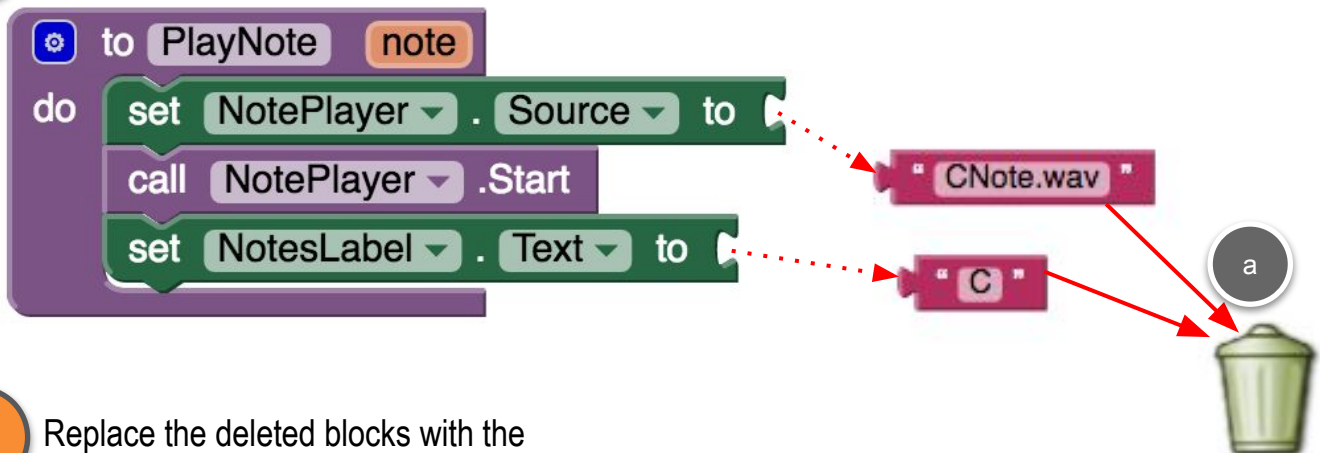
Copyright MIT 2021

WRITE A PROCEDURE

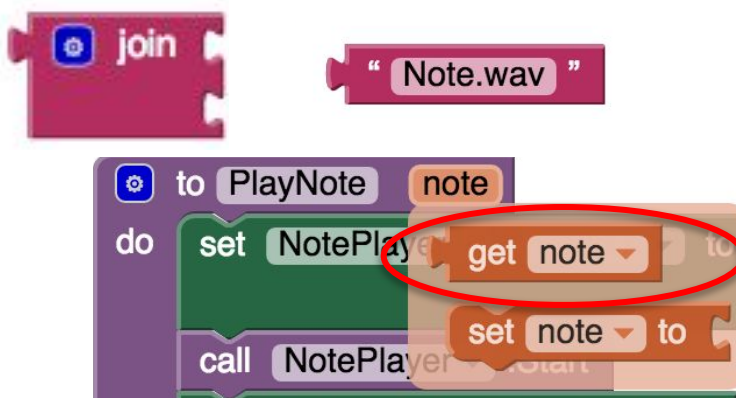**4** Then, rename the input "**note**".

5 The code for the procedure is similar to the **CNote.Click** block, so drag the code inside **CNote.Click** to the new **PlayNote** procedure Block.
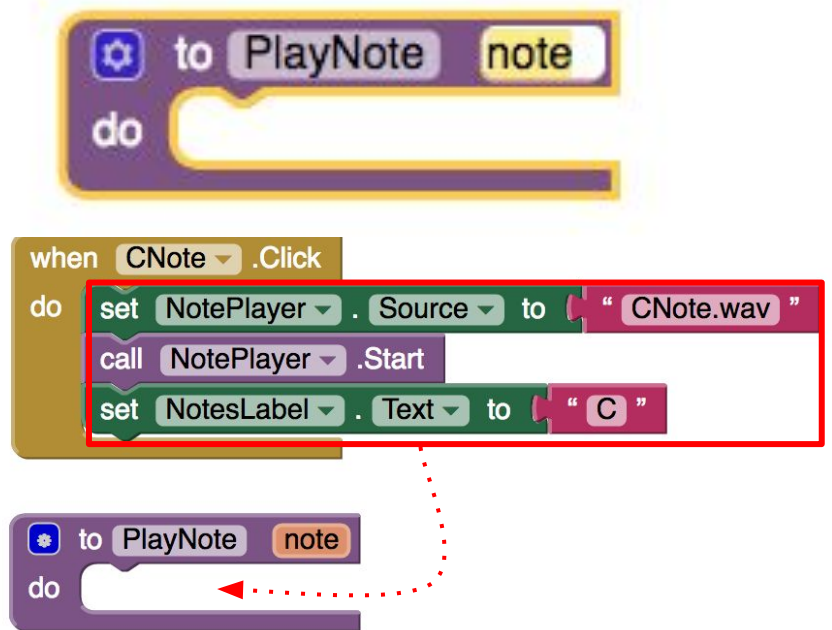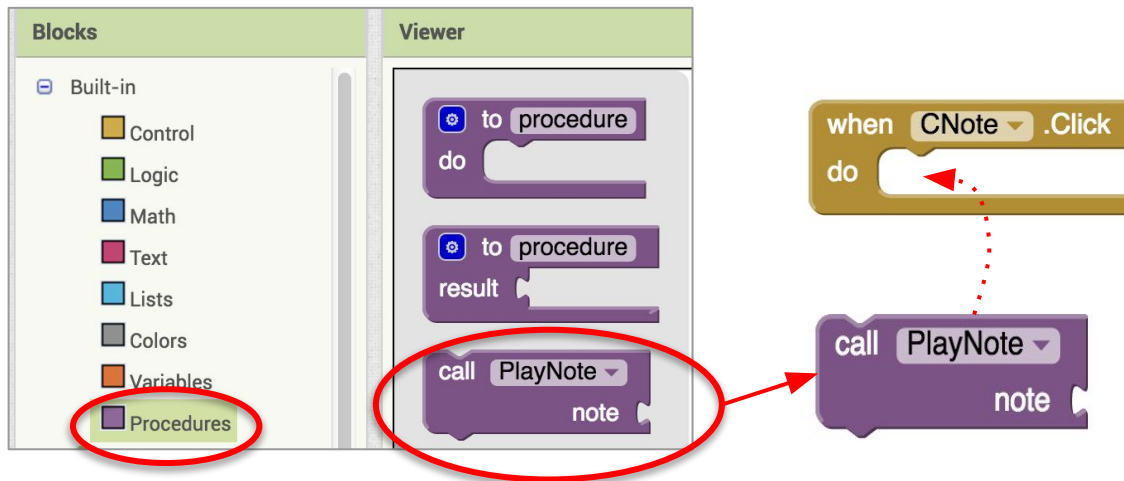
6 Delete the original Text blocks.

**7** Replace the deleted blocks with the following blocks.

If **note** is "C", joining it with "Note.wav" makes "C"+"Note.wav", or "CNote.wav", the name of the sound file.
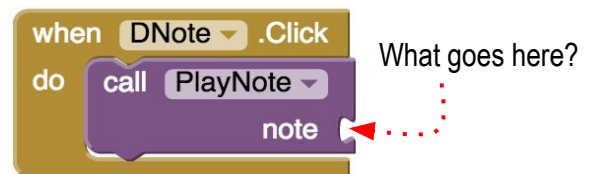
CALL PLAYNOTE

8 Drag out a **call PlayNote** block from the **Procedures** drawer and add to **CNote.Click** so that the PlayNote code runs when the C note is pressed.

8 Complete the puzzle piece and pass "**C**" as the note to **PlayNote**.

Grab this from the Text drawer.
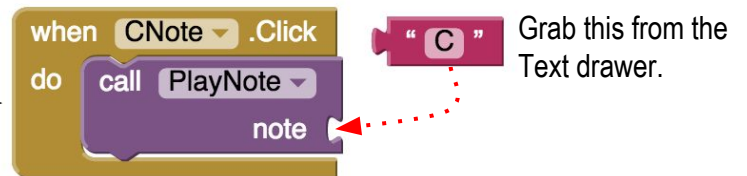
10 Do the same for **DNote.Click.** Delete the code blocks that were in **DNote.Click**, and replace it with a call to **PlayNote.**
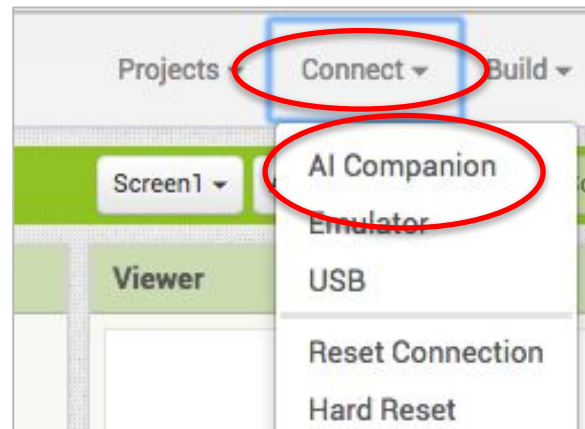
What goes here?

11 Add **.Click** event blocks for all the other note buttons, and call **PlayNote** with the correct note for each button.
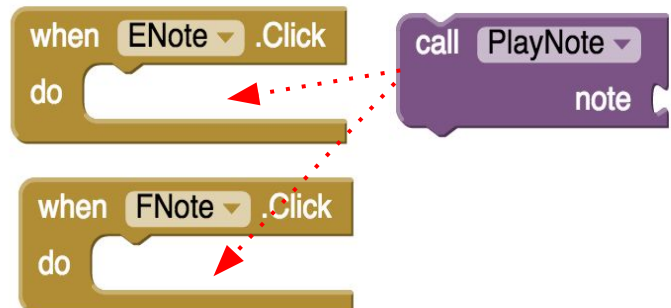
TEST THE APP

12 Test your app with the MIT AI2 Companion to make sure you can play all eight notes and see the correct notes displayed.

CHANGE THE APP

Let's make a change to our app.
Instead of just displaying the current note, let's display *all* the notes pressed in sequence , like "**A C C C D E F**" etc.

By using a procedure, you can update your app in one place, instead of having to update all the Click event blocks!

13 In the **PlayNote** procedure, update **set NotesLabel.text** so it uses a **join** block, with 3 inputs.
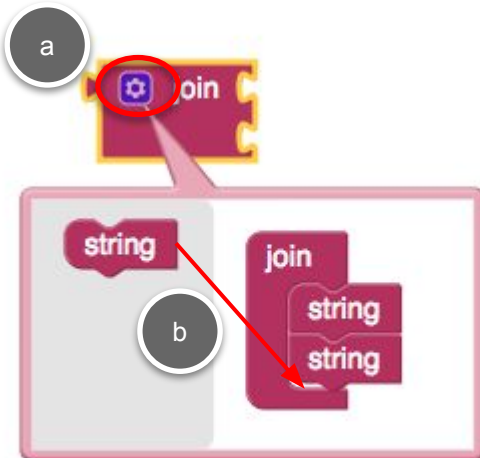
These are the other blocks you will need.

get note

A space " ", not a blank!

NotesLabel . Text

MIT
APP INVENTOR

CLEAR BUTTON

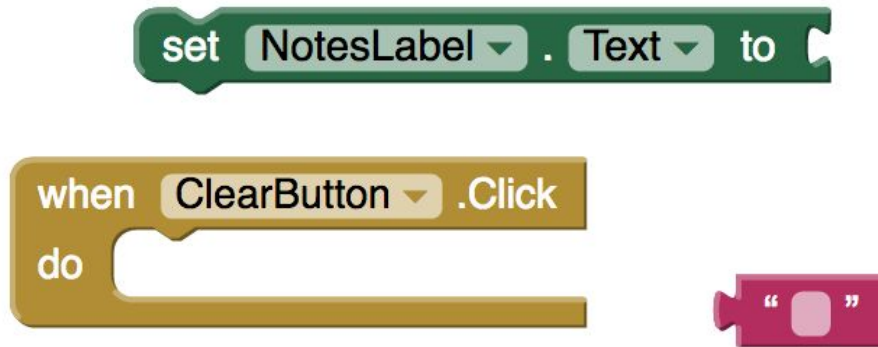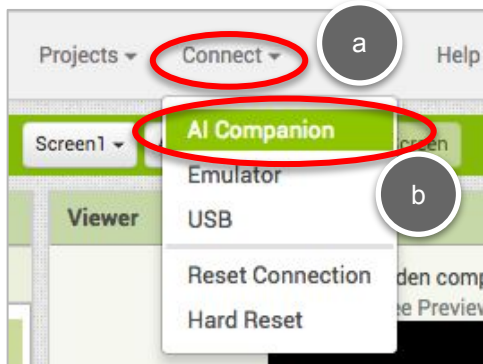**14** Sometimes the string of notes can get too long, so code the Clear button to reset the string. Use the following blocks.
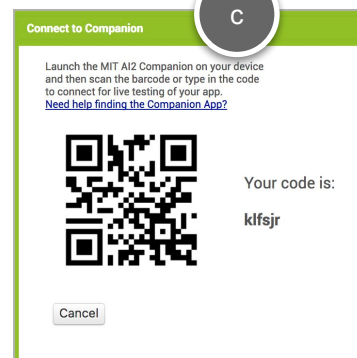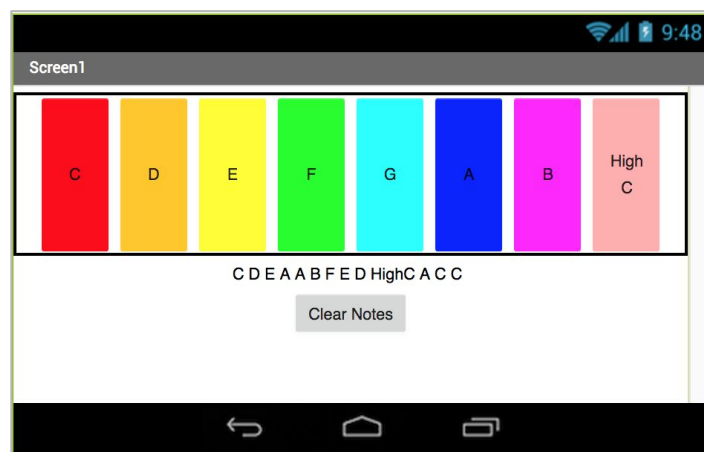
set NotesLabel . Text to

when ClearButton .Click
do

" "

TESTING!

15 Now test your app on your tablet!

Projects ▾   Connect ▾ (a)   Help

Screen1 ▾   AI Companion
             Emulator   (b)
Viewer   USB

             Reset Connection   den comp
             Hard Reset   e Preview

Scan the QR code with MIT AI2 Companion

Connect to Companion   (c)

Launch the MIT AI2 Companion on your device and then scan the barcode or type in the code to connect for live testing of your app.
Need help finding the Companion App?

Your code is:

klfsjr

Cancel

16 Play with your piano. Try to press all the "keys" and you should hear the corresponding notes and see which notes you play appear in **NotesLabel**.

Screen1

C D E F G A B High C

C D E A A B F E D HighC A C C

Clear Notes

MIT
APP INVENTOR

Extend Your App

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

Add the Sharp notes - the note files are included in Media.

Change the keys to black and white to look like a piano

Add a SoundRecorder to record the music.

What other ideas do you have?
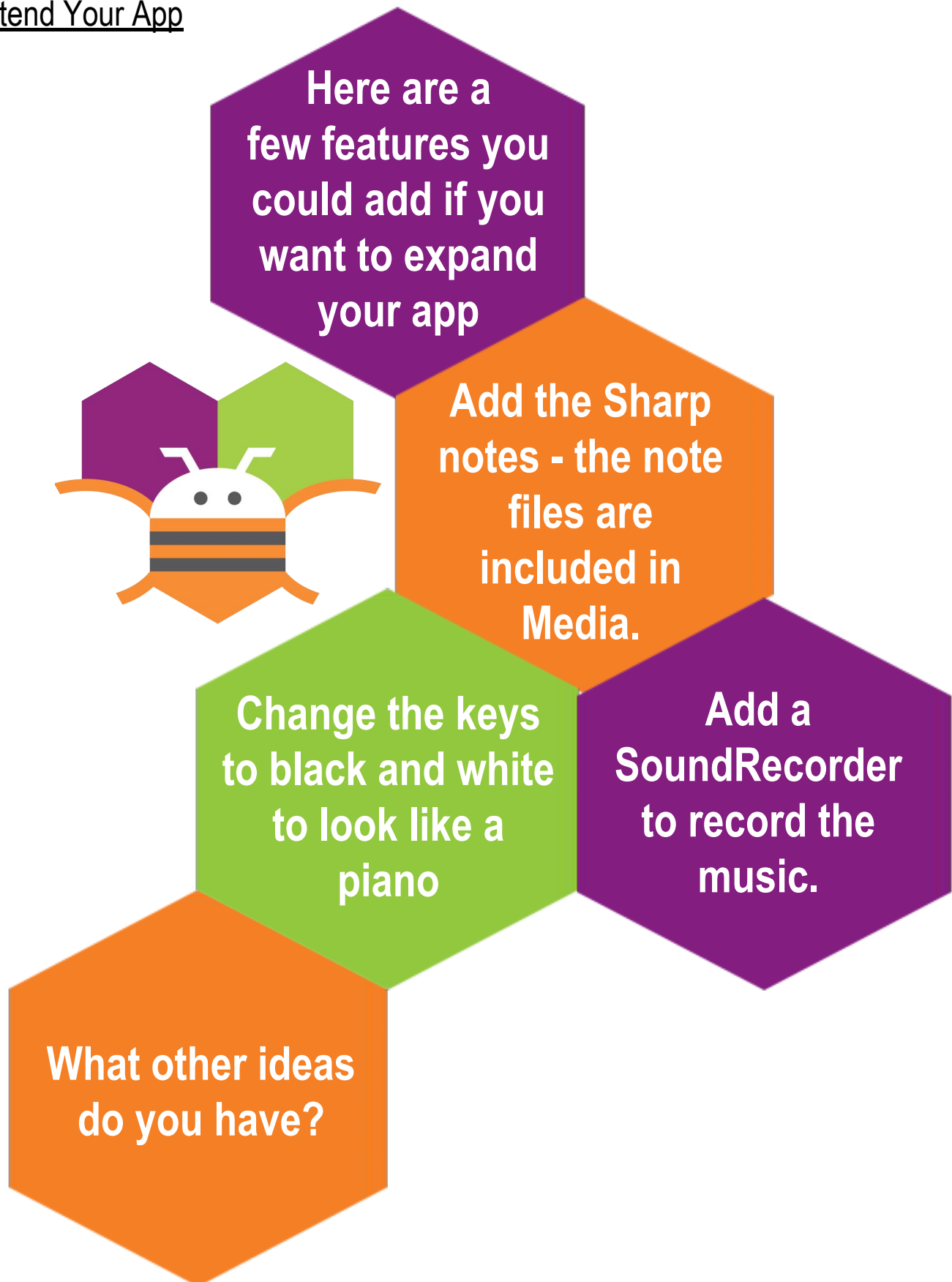
COMPUTATIONAL THINKING CONCEPTS

The following are the Computational Thinking Concepts learned in Part 3.

My Piano
1. Naming
2.   Abstraction   and   Modularization   (Procedures)