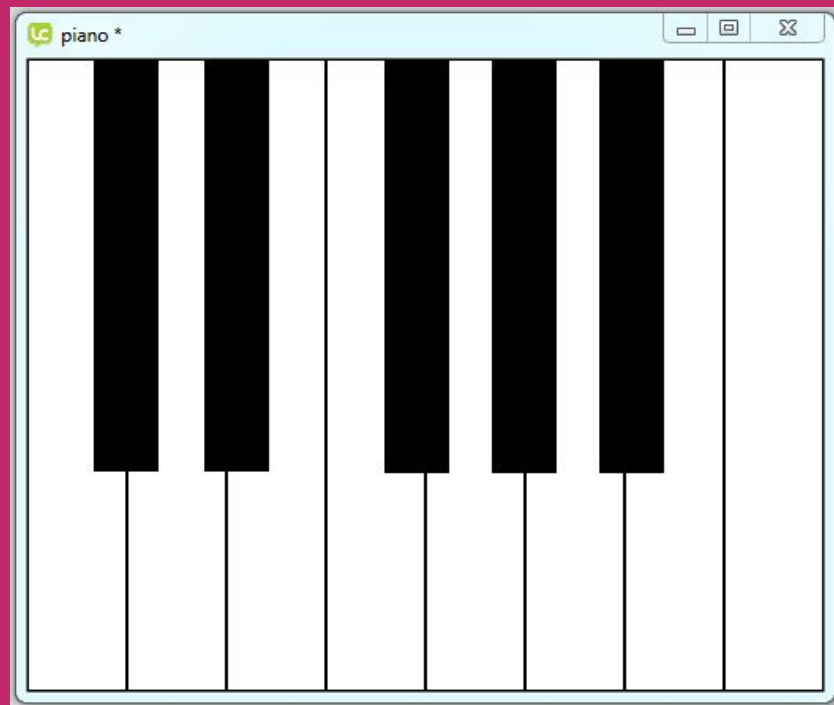


Playing the piano

Now that we have the User Interface for the piano app set up, we want to be able to play some notes.

The first thing we need to do is to make sure we have all our piano sounds in a place that the piano app can find - just like we did with the Jungle Soundboard. If you haven't downloaded the sounds and put them in the right folder, go back to the first lesson and follow the steps there now.



Setting the default folder

To make sure the app can find the sound files, we need to set the **defaultFolder**. The **defaultFolder** tells the app where to look for files it wants to use. These might be sound files, videos, PDFs, etc.

Just like in the Jungle soundboard, we will add some code to set the **defaultFolder** when the stack is opened.



Setting the default folder

We will add code to the *Stack Script* to do this.

Open the *Stack Script* from the *Object Menu* and add the **preOpenStack** handler, which sets the **defaultFolder** property when the stack is opened.

```
on preOpenStack
    set the itemDel to "/"
    set the defaultFolder to (item 1 to -2 of the filename of me)
end preOpenStack
```

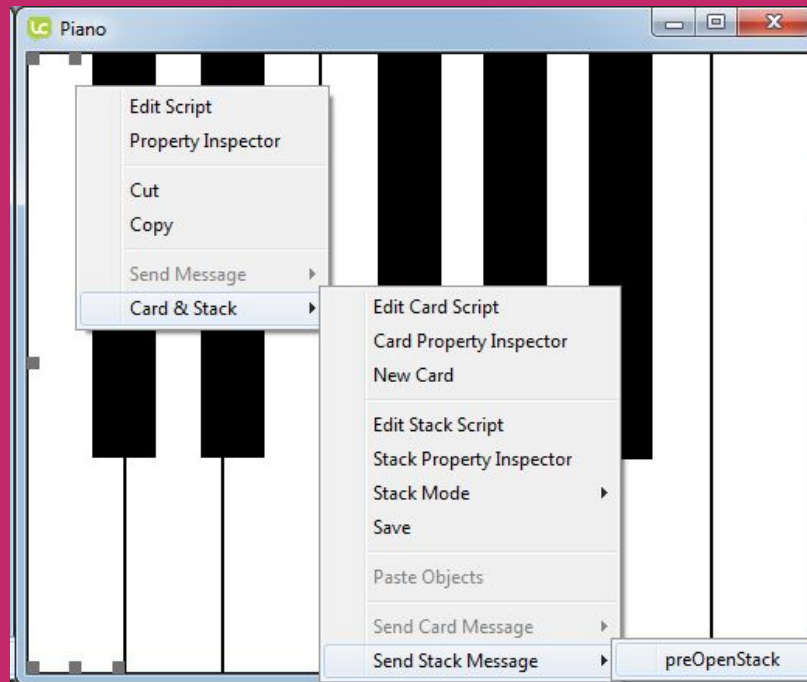
Remember to click *Apply* to save your code!

Setting the default folder

Now we want the code we just wrote to be executed. We can tell LiveCode to send the `preOpenStack` message whenever we want:

- Make sure you are in **Edit mode**.
- Right (or control) click anywhere.
- Choose *Card and Stack* -> *Send Stack Message* -> `preOpenStack` from the menu.

This will send the message and the code will be executed.

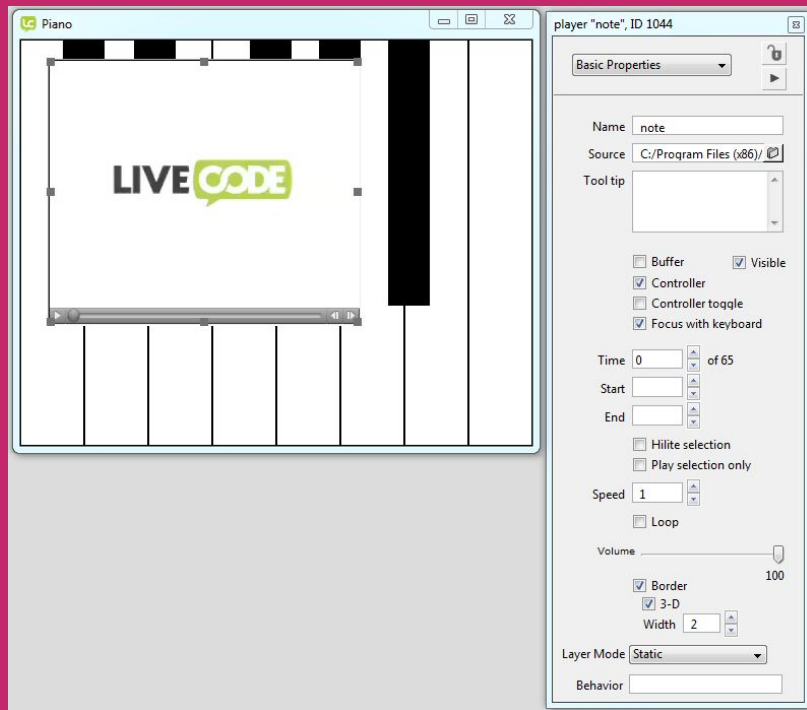


Add a player object

In order to play sounds, you need to add a **player** to your app.

Drag out a **player object** from the tools palette and drop it anywhere on the app.

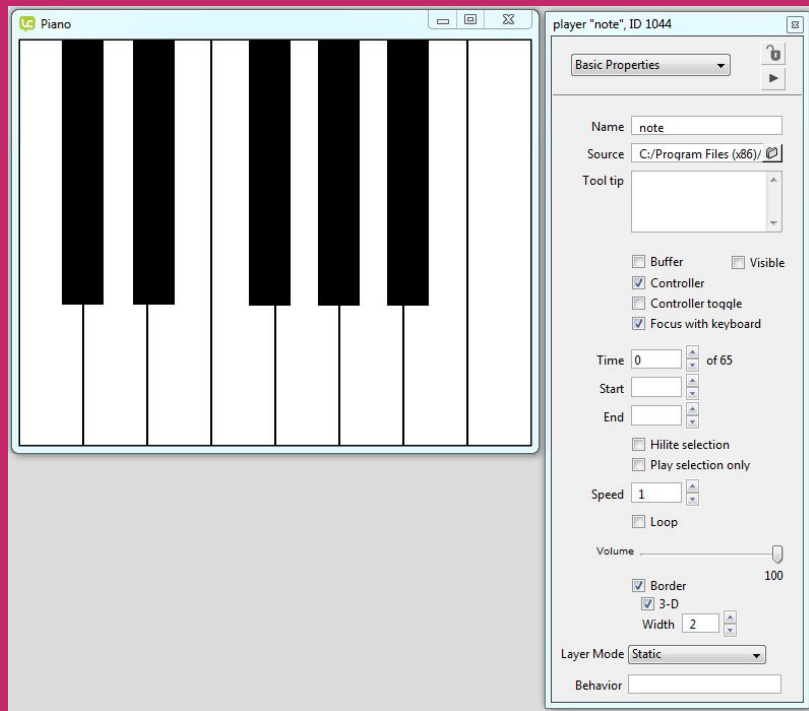
Open the *Object Inspector* from the *Object* menu and set the **Name** property of the player to **note**.



Add a player object

You do not want the user to be able to interact with the player, so you want to hide it.

In the **Basic Properties** pane, uncheck the box next to **Visible**. The player should now be hidden!



*Well done, you are halfway
through this step. Swap pairs
now.*



Pairs Swap

Grouping the key buttons

We have 13 buttons that will all do pretty much the same thing

1. Get the path to the right sound file.
2. Set the **Filename** property of the player.
3. Start the player.

In the Jungle Soundboard app, we added code to each button but we have a lot more buttons this time and we don't want to have to type in the code each time.

What we can do this time is to group all our buttons together and then add the code to the group of buttons.

Creating a group

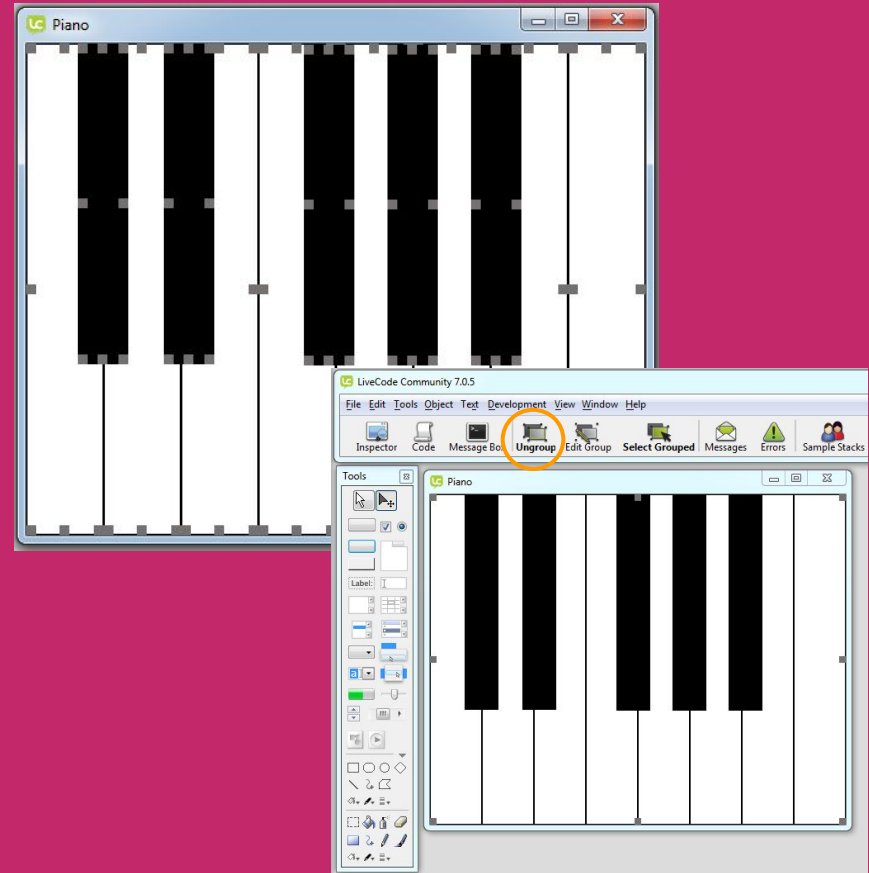
A group is a single object that holds a set of objects. Once you've created the group, it becomes an object in its own right. You can select, copy, move, and resize the group and all the objects in the group come with it. A group has its own properties and its own script.



Creating a group

To create a group, we select all the buttons. You can do this by choosing *Select All* from the *Edit* menu or using the keyboard shortcut. You can also select multiple objects using the mouse. Make sure you are in **Edit mode**, hold down the shift key, and click on objects to select them.

Select all the buttons and then click the **Group** button in the *Menubar* to group them together.



Adding code to the group

Now that we have a group, we can add a `mouseDown` handler to the group and it will be triggered when any object in the group gets a `mouseDown` message.

When the group gets the `mouseDown` message, we can check which button was actually clicked and set up the player to play the right sound.

Notice that in this app we are using the `mouseDown` message rather than the `mouseUp` message, which we used in the Jungle Soundboard. This is because the piano should make a sound when the user presses down on the key, so we want the app to respond to the `mouseDown` message.



Adding code to the group

Now we will add code to the group:

- Make sure you are in **Edit mode**.
- Select the group.
- Open the *Object Script* from the *Object* menu.
- Add the code on the next slide.
- Remember to *Apply* your code.



Adding code to the group

1. Find out which button was actually pressed and store it in a variable.
2. Set the **filename** property of the player to the relevant file. LiveCode will look for this in the default folder.
3. Start the player.

```
on mouseDown
```

1. put the short name of the target into tTarget
 2. set the filename of player "note" to ("Resources/sounds/" & tTarget & ".mp3")
 3. start player "note"
- ```
end mouseDown
```

the target: tells us which one of our buttons was actually clicked on

the short name: give us the **Name** property

so

the short name of the target: gives us "C" or "F#", the **Name** property of the button that was clicked.

# *What is a variable?*

Sometimes you want to store a piece of information that you will need later. We do this using a **variable**.

A **variable** is a place to store data. This data can be a number of things, such as an integer or a piece of text. If you think of a variable as some sort of container - like a box or a jar - you can put your data into the container to store it. You can also take stored data out of the container and put it into another one.

For example, your name is a piece of data that you can store in a name tag. Any time you want to get the name, you look at the name tag.



# *What is a variable?*

To create a variable in LiveCode, we just put the value we want to store into it. We give the variable a name or label so that we can look up the value in it whenever we need it.

The line:

```
put the short name of the target into tTarget
```

Will create a variable called **tTarget** and put the short name of the clicked button into the variable so we can look it up again later.



# *Test the Code*

Switch into **Run** mode in the tools palette and click on one of the piano keys.

You should hear a note! Try some more - maybe play a tune!





# *Your App so Far*

Congratulations, you have now finished your piano soundboard app!

You can use it because you have LiveCode installed, but what if you want to share it with people who don't have LiveCode?

In the next step, you will be saving your app so that you can share it with anyone.





**Well Done**



# Section Complete



*Time Up?*



*Save Your Work*



*Pawns Swap*