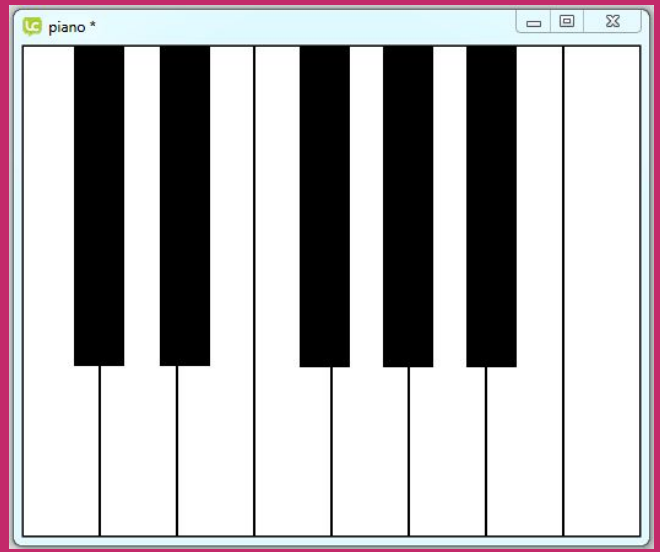
Creating the piano keys

Next we are going to create the piano keys:

- 8 white keys
- 5 black keys

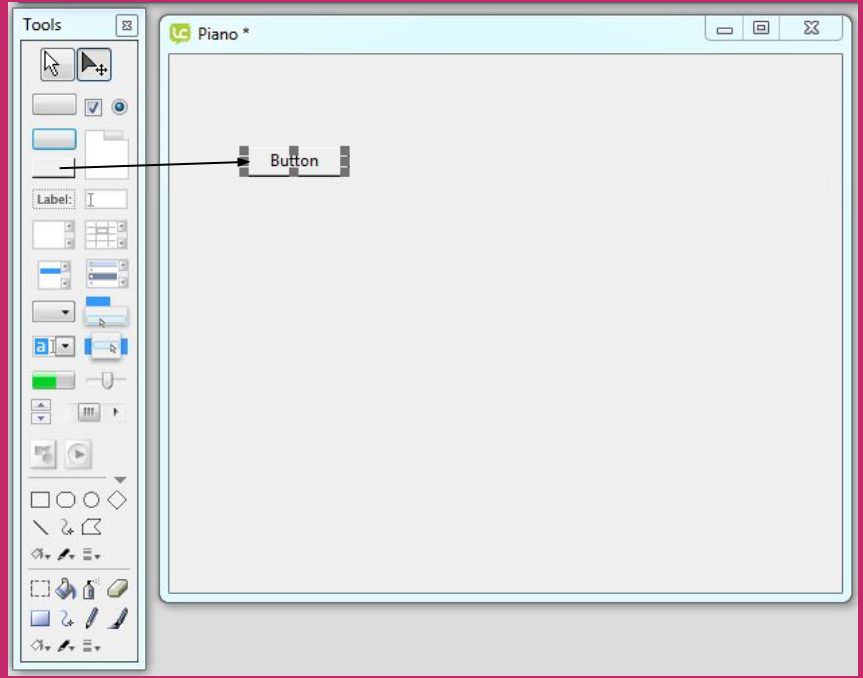
In this app, we'll be creating all the objects in LiveCode, so we don't need any images this time.



Creating a white key

Each of our keys will be a button, so we will start by creating a button:

- Make sure you are in **Edit mode.**
- Drag a **Rectangle button** from the Tools Palette onto your stack.



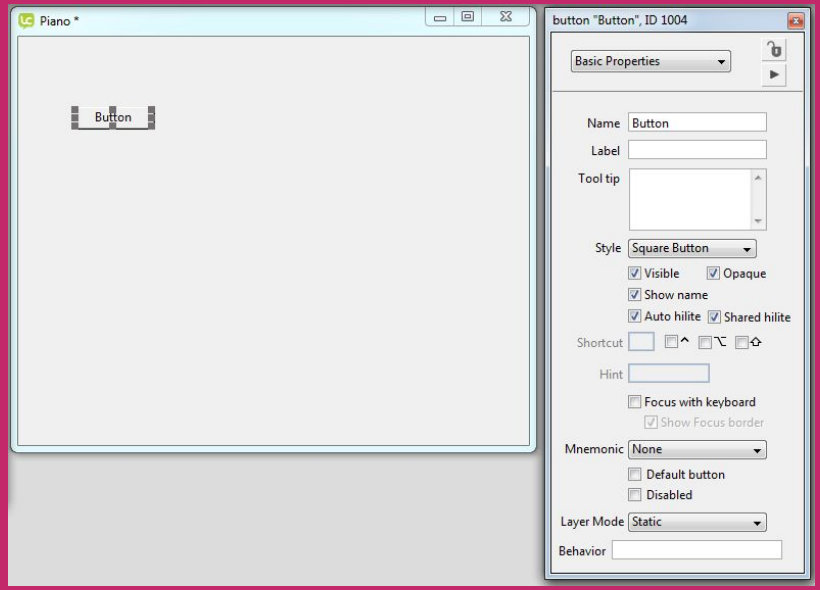
Setting the button properties

Next, we will set some properties of the button to change how it looks.

We want a plain, 2D, white rectangle with a black border and no text on it.

We will get this look by setting properties of the button.

Select the button and open its *Object Inspector* from the *Object* menu.

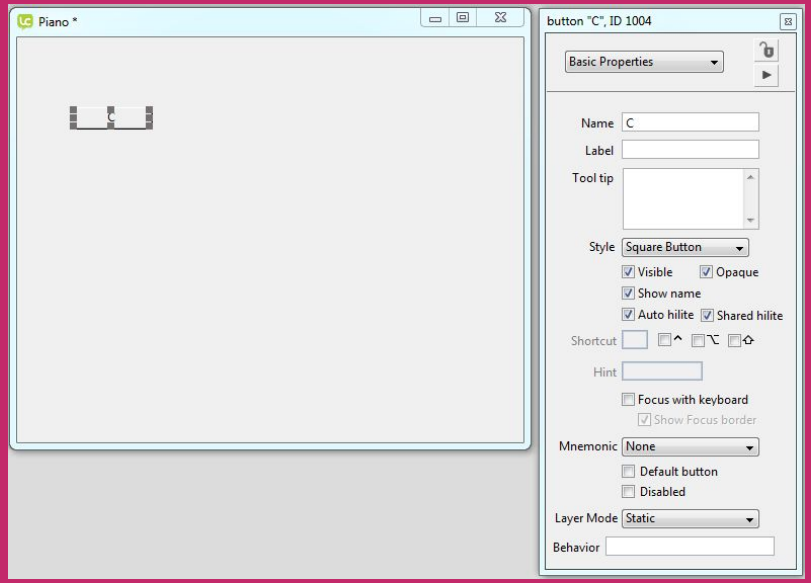


Setting the Name of the button

We want each of our keys to have unique, sensible names.

Because each key will be a note, we will set the **Name** property of the key to show which note it represents.

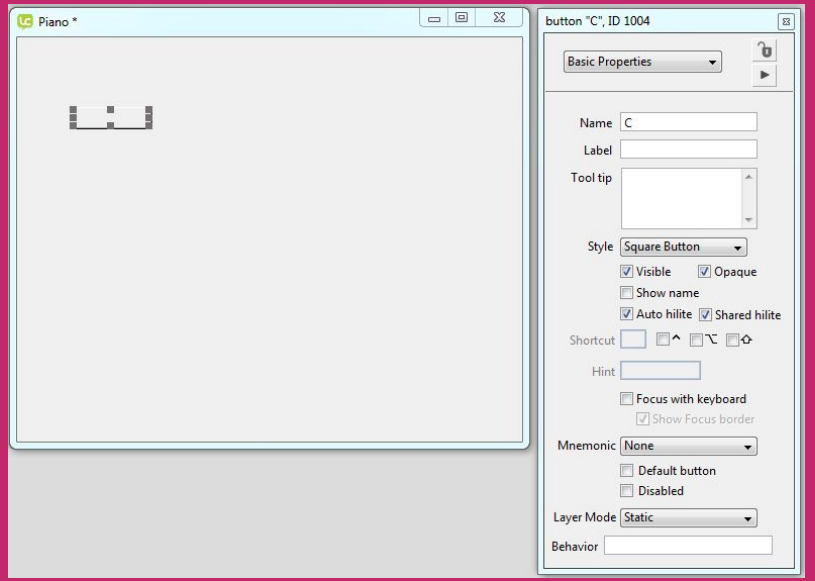
Set the **Name** of the button to **C**.



Showing the name of the button

We don’t want the name of the button to be displayed on it.

Turn off the **Show name** property of the button by unticking the **Show name** box..

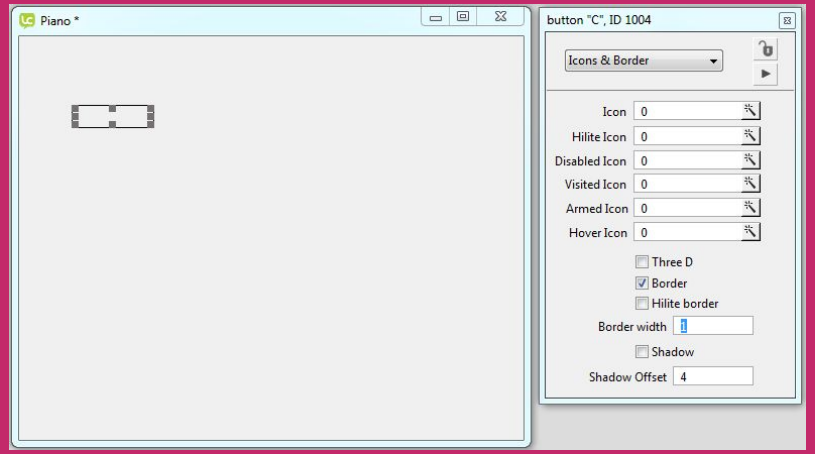


Setting the border of the button

Next, we want to make the button 2 dimensional and set the size of the border.

Switch to the **Icons & Borders** page of the **Property Inspector**.

- turn off **Three D** (untick)
- turn off **Hilite border** (untick)
- set the **Border width** to 1



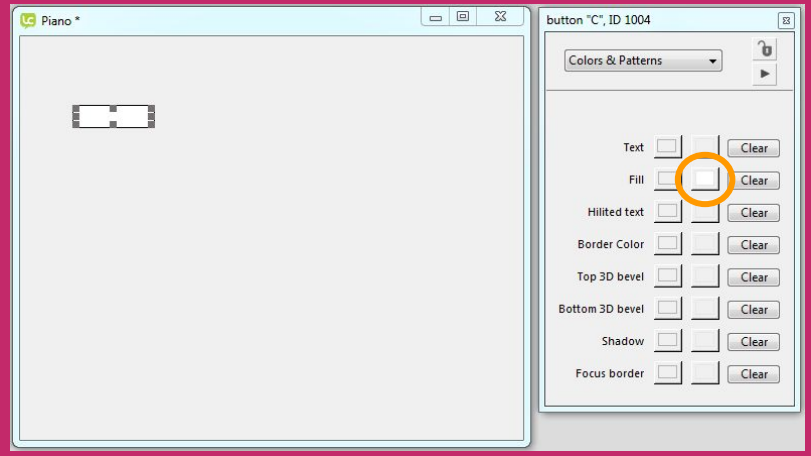
Setting the color of the button

Next, we want to make the button white.

Switch to the **Colors & Patterns** page of the **Property Inspector**.

- Click the **Fill** colour. This will show a color picker.
- Choose white.
- Click *OK*.

You should now have a white button.



Setting the size of the button

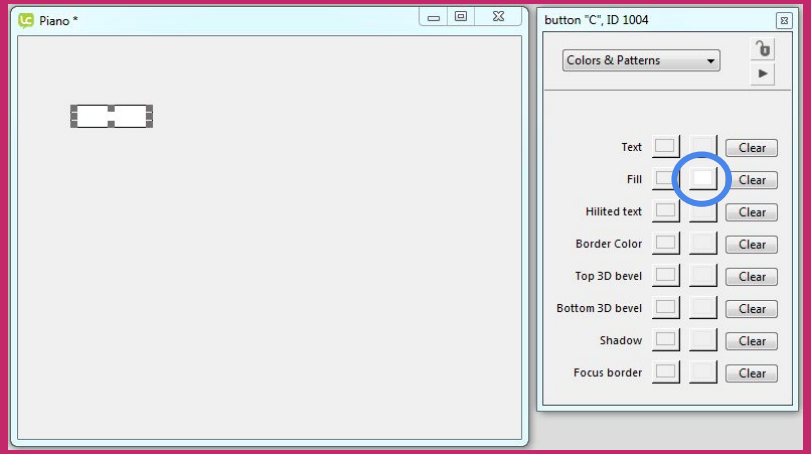
Next, we want to make the button the right size. It will be the same height as the stack and we will have 8 of them.

You can either work out the width of buttons:

Width = width of the stack / 8

or

Drag the button to be the size you want it to be. You can then change the size of the stack once all the buttons are on it.



Setting the size of the button using properties (1)

If you want to work out the exact size of the buttons, open up the *Stack Inspector* from the *Object Menu*.

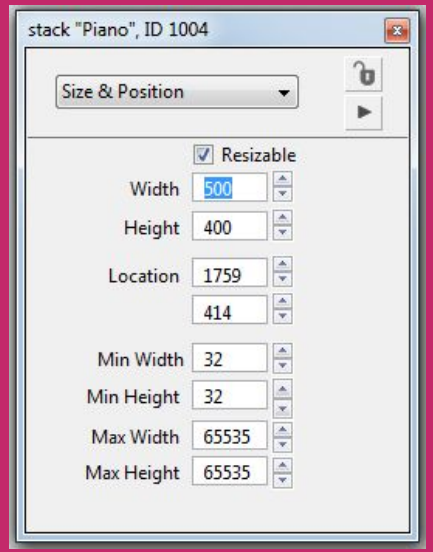
Switch to the **Size & Position** page.

Take a note of the **Height.**

Work out the **Width** divided by 8 and take a note of it.

This will be the height and width of your button.

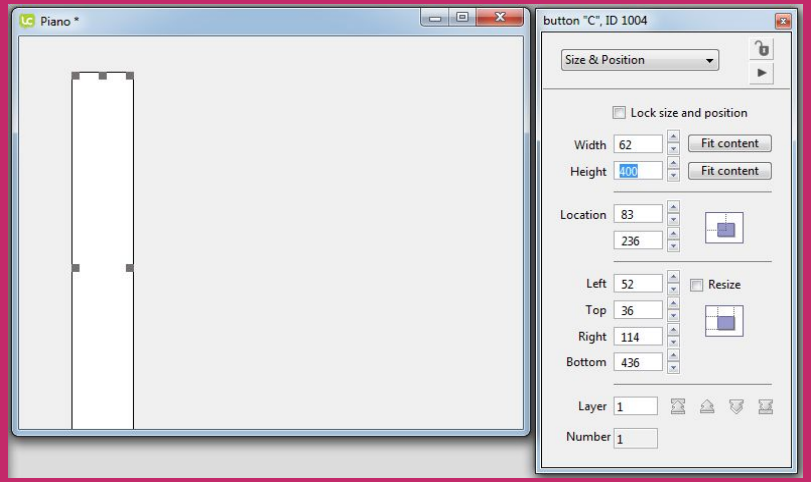
My button's **Height** will be 400 and its **Width** will be 62, but yours might be different.



Setting the size of the button using properties (2)

Now that you have the height and width for the button you need to set them.

- Select the button.
- Open the *Object Inspector* from the *Object Menu*.
- Switch to the **Size & Position** pane.
- Set the **Width.**
- Set the **Height.**

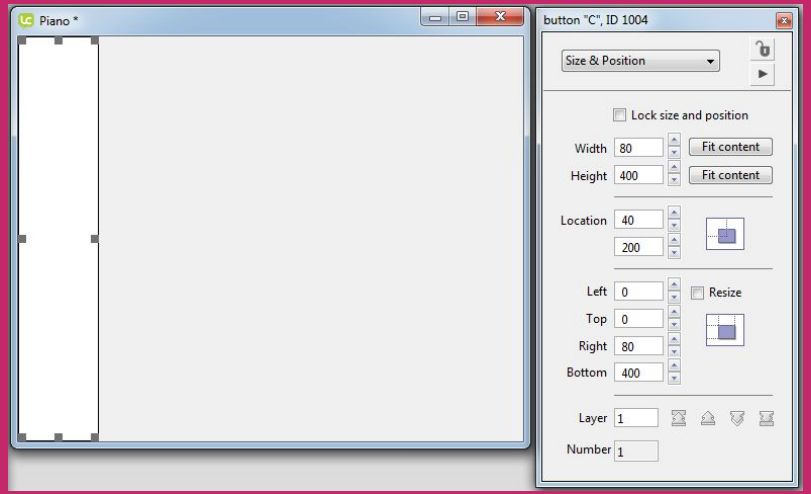


Setting the size of the button using properties (3)

Finally, we need to position our button. We want the top left corner of the button to be in the top left corner of the stack. We do this by setting the **Left** and **Top** properties of the button.

In LiveCode, the 0,0 coordinate is the top left of the stack. Larger x coordinates are further right and larger y coordinates are further down.

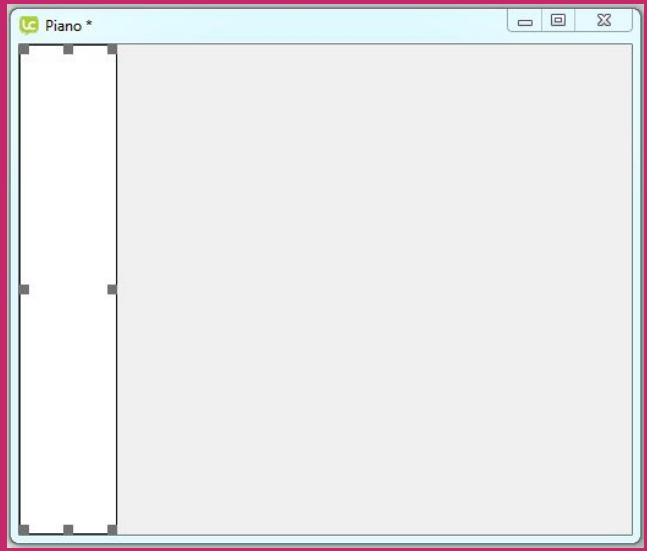
Set the **Left** to 0.
Set the **Top** to 0.



Setting the size of the button directly

If you'd rather build your app freehand, you can change the size and position of your controls by selecting them and dragging them around - just like in a drawing program. You want the button to be the height of the stack and about the right width to have 8 buttons side by side.

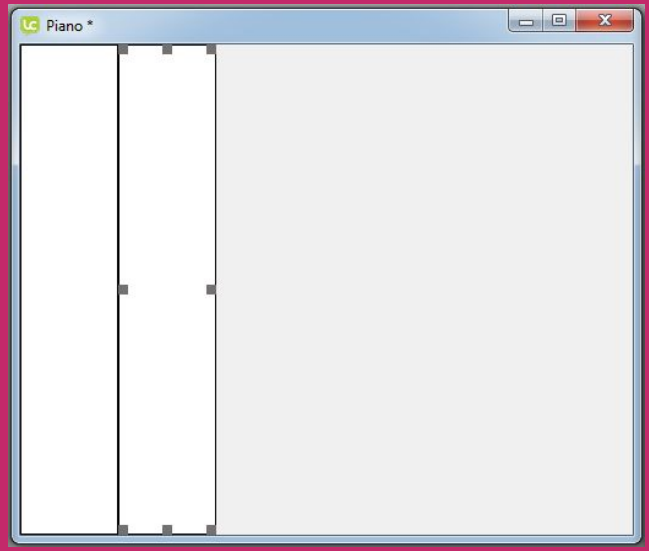
- Ensure you are in **Edit mode.**
- Select the button.
- Change the size of the button by dragging the selection handles.
- Move the button into position by dragging, or using the arrow keys.
- Put the button into the top left corner of the stack.



Creating the white keys

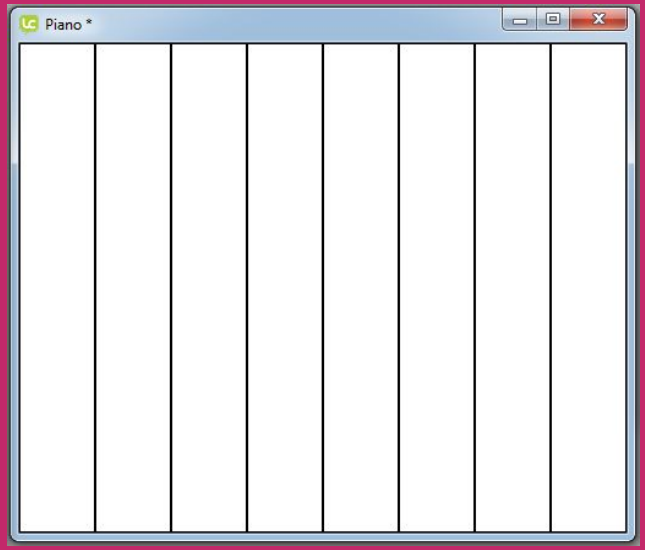
Now we have 1 white key we need 7 more. We could create them all each time but instead lets save time by copying and pasting them.

- Select the button.
- Copy it using *Copy Objects* in the *Edit* menu, or the keyboard shortcut.
- Paste is using the *Edit* menu or keyboard shortcut.
- Position the new button next to the first one using the mouse or arrow keys.



Creating the other white keys

Now repeat this 6 more times until you have 8 buttons.

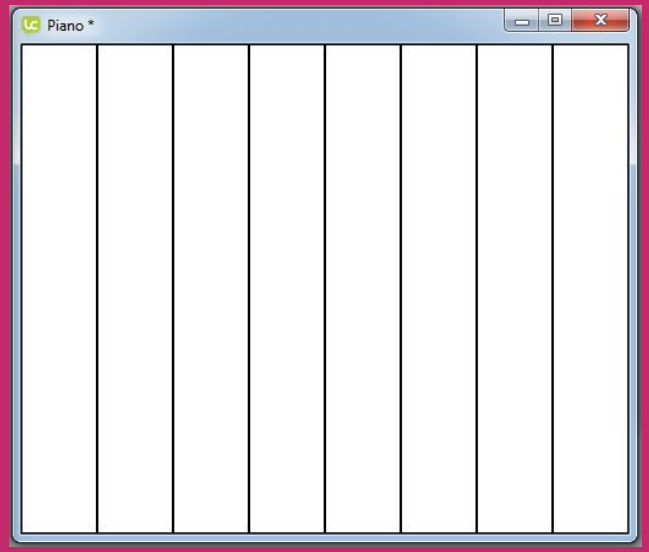


Setting the button names

Next, we need to make sure each of our buttons has a unique, descriptive name:

- Select the second button.
- Open the *Object Inspector* from the *Object* menu.
- Set the **name** of the button to **d**.

Repeat this for the next 6 buttons, setting their **name** properties to **e**, **f**, **g**, **a**, **b** and **c2**.



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

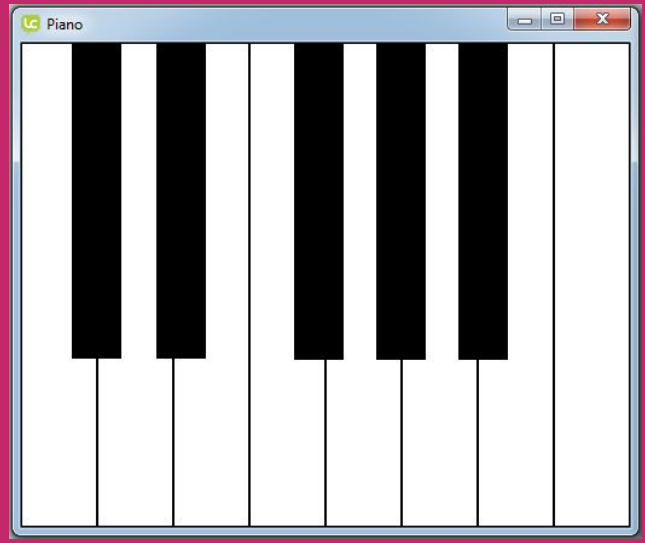
Pains Swap

Creating the black keys

Next, we want to create the black keys.

We will start by selecting one of the white keys, copying and pasting it, and changing some of the properties to create a black key.

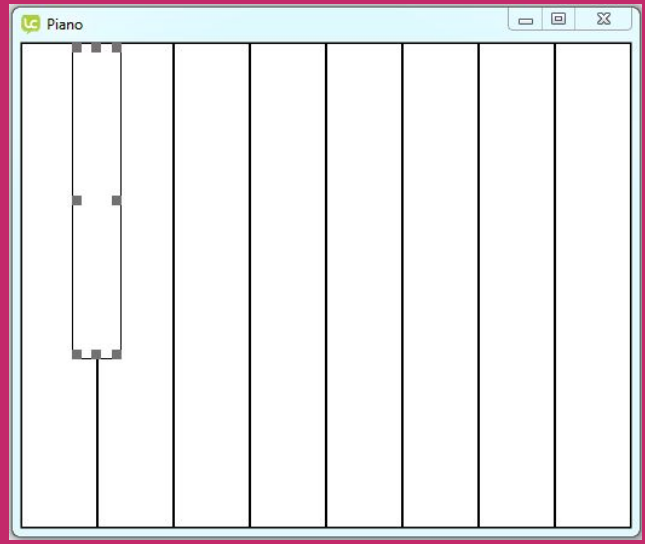
We will then create the other 4 black keys.



Creating a black key

- Select the first white key.
- Copy and paste the key.
- Move the copy to be between the first and second keys.
- Make the new button slightly narrower and shorter.
- Place the button at the top of the stack.

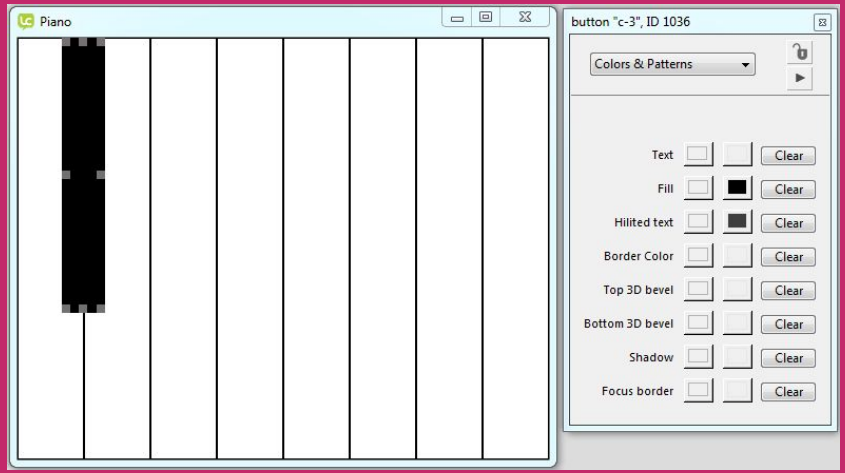
My button has **width**=40 and **height**=260, but yours can be any size you think looks right.



Setting the color of the black key

Now we want to make the black key black:

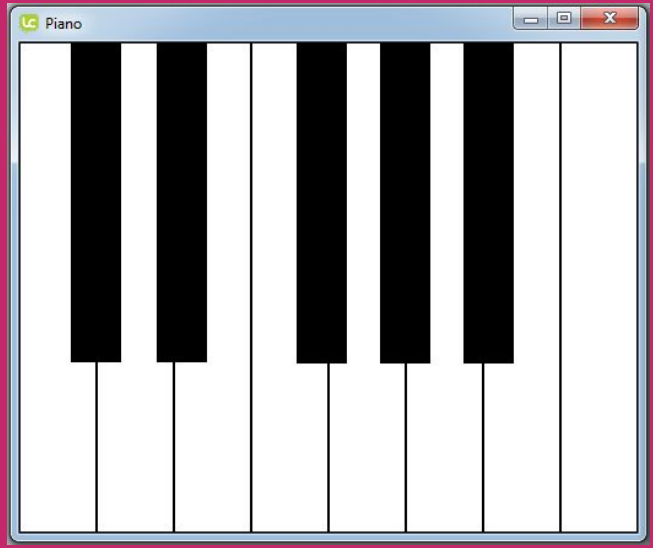
- Check the button is selected.
- Open the *Object Inspector* from the *Object* menu.
- Go to the **Colors & Patterns** pane.
- Set the **Fill Color** to black.



Creating the other 4 keys

Copy and paste the black key 4 times and place the buttons on the stack as shown in the image.

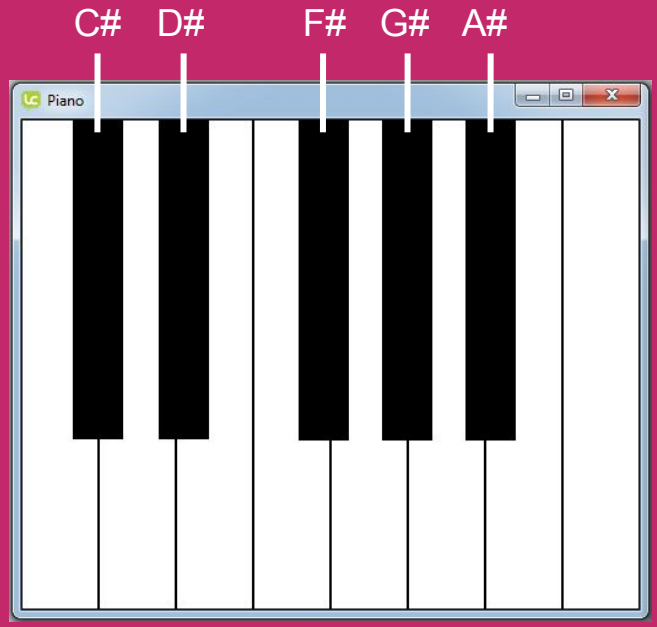
You can move the buttons using the mouse or the arrow keys.



Setting the button names

As with the white keys, we want the **name** property of each of the black keys to match the note it represents.

Select each button in turn and use the **Object Inspector** to set the **name** property of each button as shown in the diagram.



Your App so Far

Congratulations, you have now finished setting up the User Interface (UI) of your piano soundboard app!

Switch into **Run** mode in the tools palette and try clicking on the buttons.

The next step is to add code so that your app actually does something!

Section Complete

Time Up?

Save Your Work

Pairs Swap