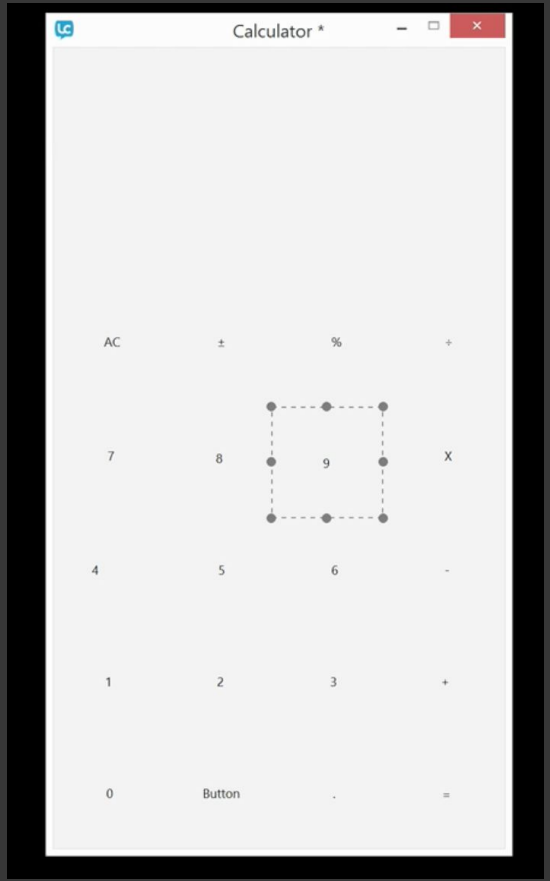
Setting More Button Properties

We want the buttons to have a plain, flat appearance in the iOS style. Select all the buttons and set the following properties, following the UI specification:

Pane	Property	Value
Icons & Border	Three D	False
Icons & Border	Border	False
Icons & Border	Hilite Border	False
Icons & Border	Border Width	0

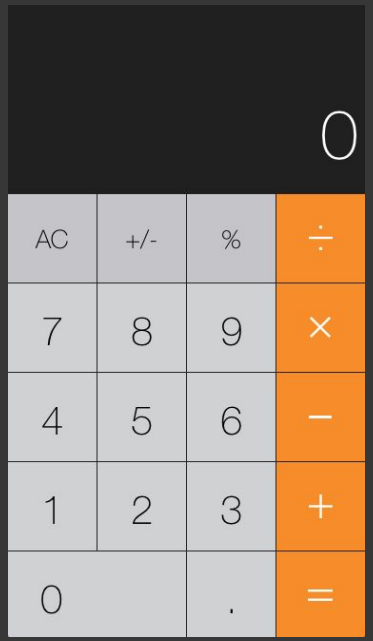


Setting the button labels

The buttons do not all have the same labels or colors, so we need to set the name, label, fill color and hilited text color of each of the buttons individually. First we set the names and labels for each button. Set the name and label properties in the Basic Properties pane of the Property Inspector of each button.

Name	Label	Name	Label
clear	AC	divide	÷
toggle	+/-	multiply	x
percent	%	plus	+
equals	=	minus	-

For the buttons that display numbers, including the decimal point, set the name of the button to the number that it displays. Do not set the label.



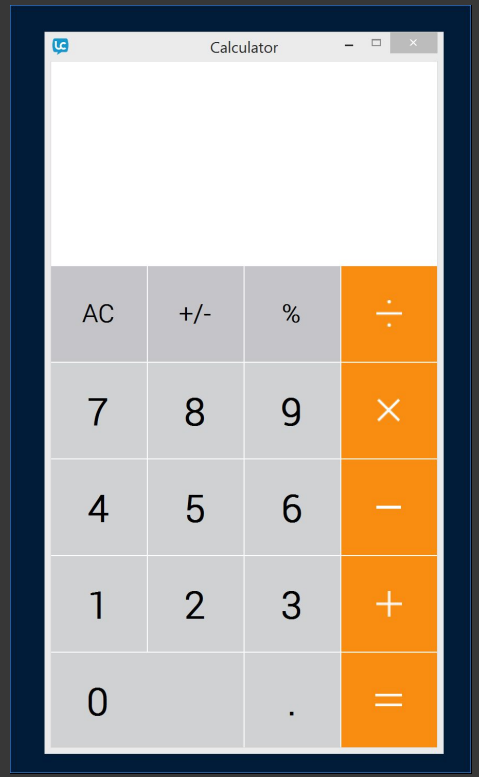
Extending the 0 button

We now have a “spare” button next to the 0 button. We will delete it and extend the 0 button to fill the space.

- Select the spare button
- Delete it using the Delete key
- Select the 0 button
- Drag it wider to fill the space
- Open the Property Inspector
- Set the Text Align to Left

To get the 0 to align with the 1 we need to set the margins of the button. The margins specify how much empty space is left on each side between an object's edges and its contents. If a single integer is specified, all four margins of the object are set to that number of pixels. If four integers are provided, the object's top, left, bottom, and right margins are set to each one respectively:

- Set the margins to 38,4,4,4



Button Names and Labels

You may be wondering why we have given the buttons names AND labels…

Names are descriptive and are used to refer to the object in code. **Labels** appear on the button and can be changed without changing any references in your code.

- The **Name** property is used as an identifier to refer to a button in code
- The **Label** property is displayed on a button

If a button does not have a **Label** its **Name** is shown, if the **Show Name** property is true.

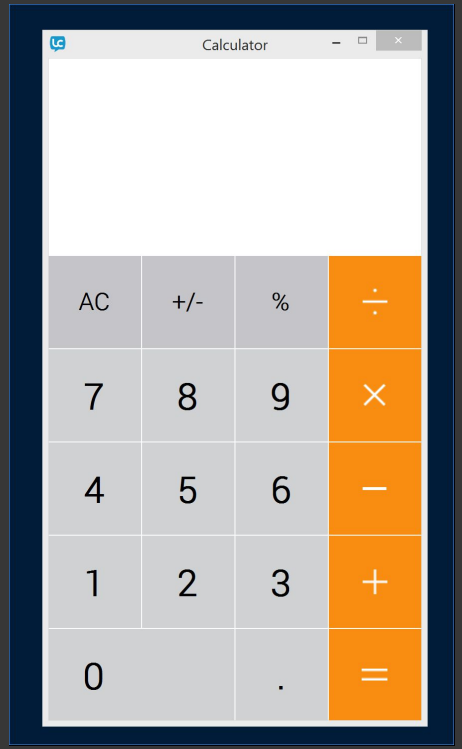
For example, you might have an app where the user can choose the language. You will therefore want to change the **Label** of the button so that it is in the user's chosen language.

Setting the Button Properties

Next we want to set the color and text properties of each of the buttons. We want to set the **Fill color** property and the **Hilite text color** property. Both can be found in the “Colors & Patterns” pane in the Property Inspector.

We also want to set the **Text Size** which is in the Text Formatting pane.

Button Name	Text color	Fill color	Hilite text color	Text Size
clear, toggle and percent	-	196, 196, 201	187, 187, 193	27
equals, divide, multiply, plus and minus	white	249, 141, 17	236, 126, 6	42
0 - 9 and .	-	208, 209, 211	199, 200, 203	42



User Interaction

User Interface is also about User Interaction. When you press any of the buttons, the color will change because we set the **Hilited text color** to a slightly darker color. This gives feedback to the user that the button was pressed.

Switch to **Run** mode in the tools palette and try it now. The images below show the color change when the minus button is pressed.