

Flex in a Week, Flex 4.5

### **Video 5.04: Introducing advanced CSS selectors**

In this video, I will use advanced CSS selectors to further style the Employee Portal application.

You will learn about the descendent, pseudo and id selectors.

Here is the completed Employee Portal application for this video.

When I mouse over any of the Button controls in a Panel container, you can see that the over states are blue.

When I click on a button that is in a Panel container, you can see that its down state is green.

However, when I mouse over and click on the Logout Button control, which is not inside of a Panel container, its over state and down state are both gray.

I have implemented a CSS descendent selector which explicitly targets Button controls inside of Panel containers.

Using advanced CSS selectors, you can implement very fine control over the styles implemented in your application.

To implement a descendent selector, you simply stack components in the selector definition.

This example states that you want to target all of the Spark Button controls that are in the Spark Panel containers.

You can also stack multiple levels of descendents.

In this case, I am targeting the Button controls used in a Scroller component inside of a Panel container.

This is the starter application display for the Employee Portal application.

You can see, that by default, all of the Button controls have the default Button colors.

In Flash Builder, I am opening the associated CSS file for this application.

Remember that it is hooked into the main application via this Style tag.

At the end of the CSS file, I am creating a descendant selector that first targets the Spark Panel container and then all Spark Button controls in the container.

I am setting the color property to black, and the chromeColor, which is the background color for the Button, to green, #64BC48.

When I save the file and run the application, you can see that all of the Button controls in the panels are green with black text.

The Logout button has not changed colors because it is not in a Panel container.

You'll also note that the color for the Scroller components is green.

This happens because the Scroller component is made up of Button controls, and so is also affected by the descendant selector.

I am copying the descendent selector and pasting it below itself.

Now I'm adding the Spark Scroller component as the second element in the selector and changing the chromeColor property to dary gray, #555555.

I'm removing the color property.

This code will now affect the Button controls inside of a Scroller component that's inside of a Panel container.

When I save the file and run the application, you can see that the scrollbars are no longer green, but a dark gray.

A CSS pseudo selector matches components based on an additional condition.

The condition might be dynamic and might not be defined by the document tree.

Pseudo selectors are commonly used to apply styles to component states.

This code shows a pseudo selector for both the Button up and the Button down states.

Note that you can also define universal pseudo selectors.

This code applies the same color text to all components that are in an "up" state.

In the CSS file, I'm changing the Panel container descendant selector so that it applies to the down state of the Button control.

I am also adding a selector for the Spark Button control's up state.

This will affect the up state for all Button controls, including the ones inside and outside the Panel containers.

I am setting up its text color to be white and its chromeColor to be gray.

I am copying this descendant selector that applies to the down state of the Button control in a Panel container and pasting it below the selector for the Button control's up state.

I am changing down to over to create the over state for the Spark Button control.

When the user mouses over a button within a Panel container, I want the color of the text to be black, #000000 and the chromeColor to be blue, #0D86B8.

When I save the file and run the application, I can test the results.

All the buttons have an up state with white text and a gray background.

The buttons within the panels also turn blue when I mouse over them.

When I click on a button inside a Panel container, the button turns green.

Notice that the scroll bar is also affect by these changes, but the Logout button outside the panels is not.

I'm returning the CSS file in Flash Builder and locating the descendant selector for the Scroller in the Panel container.

I am changing the selector so that it applies to the over state of the Button control.

I am copying this instance of the descendant selector and pasting it under the first.

I am changing the over state to the down state, saving the file and running the application.

Now when I mouse over or click on the scroll bar, the color is gray.

An id selector styles the component that has an id property that matches the selector defined in the CSS file.

The selector is tied to one specific component instance.

The syntax is to declare the id property prefixed with a hash, or pound, sign.

In this case, the code is targeting a Button control with the id property of submitButton to turn its text to a gray color.

Back in the main application file, I am opening the EmployeeOfTheMonth component by CTRL + clicking on it, and locating the Label control that displays the employee's name.

I am giving it an id property with a value of `employeeName` and saving the file.

In the CSS file, I am creating the id selector by typing `#employeeName`.

I am setting the color property to red, `#FF0000`.

When I save the file and run the application, the employee's name is now red.

For your next step, work through the exercise titled "Using advanced CSS selectors".