**Practical use of pseudo**

By now you should know that developers use

pseudo-classes to style and

make web pages more interactive.

In this video, you will explore pseudo-classes

that you can use to style links on web pages,

namely web-link states.

Although, you can use web-link states

for many functionalities,

developers often use them to style

the color of links that a user already opened.

For instance, a HTML file can have two a tags,

each with its own href

that links to a different page of a website.

To give the links their initial color,

you can use the link pseudo-class selector,

and to let the links change color,

once a user has visited these pages,

you can use the visited pseudo-class selector.

But if you're using

multiple pseudo-classes for a specific element,

there is an important rule to keep in mind.

The order in which you apply pseudo-classes in

a document determines which style will be displayed.

So be careful not to

accidentally overwrite a style you want to set.

Developers use the acronym LVHA to

remind them of the order that they need to

follow when writing pseudo-class rules.

LVHA stands for link, visited, hover, active.

So always apply pseudo-class styles in the correct order.

First link, then visited,

next hover, and lastly active.

Now that you know more about web-link states,

let's explore how to use the link,

visited, hover, and active pseudo-classes.

Additionally, you will learn

two more effects that you can

create with the hover pseudo-class.

In the pseudo\_links.html file,

I added three tags: a heading tag,

followed by a paragraph tag,

and another paragraph tag.

In the heading tag, I added

the name of the restaurant, Little Lemon,

and the first paragraph tag

has a link to their Instagram page.

Now I go to my CSS code by clicking on the

pseudo\_links.css file at the top of my screen.

In this file, I already added properties

related to the pseudo-classes we discussed previously,

namely link, visited, hover, and active.

To display the output of this code,

I moved back to my HTML file and

click the "View" icon at the top right of my screen.

By default, the link to

the Little Lemon Instagram page displays in

a blue color because I set it to blue with

the link pseudo-class in the CSS code.

As expected, when I move the mouse over the link,

the color changes to green because I added green

as the value of the color

attribute in the hover pseudo-class.

Let's test what happens when I click and hold the link.

It changes to white text on a black background,

just like defined with the active pseudo-class property.

When I click on the link to

visit the Little Lemon Instagram page,

the system asked me to connect to an external source,

since I am using Visual Studio Code.

I click the "Cancel" option for now.

But after I visit the link,

the color of the link will change to violet because I

defined it to do that with the visited pseudo-class.

I'm now going to demonstrate how you

can use the pseudo-class hover.

To do this, I use the pseudoshadow.html

file I created previously.

In this file, I added four paragraph tags,

each belonging to a class called textual.

I also added random text,

commonly used in publishing and

graphic design as placeholder text.

The CSS code that I added is fairly simple.

The first property defines

the properties of the textual class.

The second property defines the effect that

the pseudo-class hover will have on the textural class.

For the hover pseudo-class,

I use the shorthand for a box-shadow property.

As you've seen earlier, box-shadow

adds one or more shadows to an element.

The horizontal offset is set to five pixels,

the vertical offset is set to zero pixels,

the blur radius is set to 25 pixels,

and the color of the shadow is set to black.

I moved back to my HTML file to

output this code and click on the "View" icon.

The Little Lemon page with blocks of texts displays.

The texts looks like regular blocks of texts.

But when I hover over a block of code with my mouse,

it gets highlighted with a box-shadow around the edges.

I get the same effect when I

hover over any other block of text

as all the blocks of texts belong

to the same textual class.

This is an interesting effect that is simple to

create with the help of the hover pseudo-class.

In this video, you learned how to use pseudo-classes like

web-link states to improve the interactivity of web pages.

You also explored additional effects

that you can create with the hover effect.

Acronum

LVHA :link :visited :hover :active

Weblinks

## Question

You are developing a web page and want to add effect to links on the page. You use pseudo-classes related to web-link states to do this. But to prevent overwriting one of the rules, you must place them in the following order in the CSS file:



Active, hover, visited, link (AHVL)



link, visited, hover, active. (LVHA)



Link, hover, active, visited (LHAV)

Correct

That is right! Developers use the acronym LVHA to remind them of the order that they need to follow when writing pseudo-class rules. LVHA stands for link, visited, hover, active.