**Module (CSS and CSS 3) -2**

1]What are the benefits of using CSS?

1. Consistency and Global Styling

2. Easier to Maintain and Efficient Updates

3. Time-Saving and Faster Loading Times

4. Enhanced Design Options and Responsive Design

5. SEO-Friendly

6. Accessibility and Print-Friendly Pages

7. Platform Independence and Device Compatibility

8. CSS Minification and Page Load Efficiency

9. Cascade, Inheritance, and Class/ID Usage

10. CSS Sprites, Animations, and Effects

What are the disadvantages of CSS?

2] What are the disadvantages of CSS?

1. Cross-Browser Issues and Browser Compatibility

2. Learning Curve

3. Security Concerns

4. Complex Layouts and Limited Layout Control

5. File Size and Performance Impact

6. Over-Specificity and Overriding Styles

7. Maintenance Challenges

3] What is the difference between CSS2 and CSS3?

|  |  |  |
| --- | --- | --- |
| **Parameters** | CSS2 | CSS3 |
| Released In | 1998 | 1999 |
| specification | Single Document | Divided into individual modules |
| **Styling Option** | More than CSS | Large, along with the support for animations |
| **Fonts** | Web-safe fonts | Special fonts analogous to Google Fonts and Typecast |
| **Selectors** | Simple selectors | A sequence of simple selectors |
| **Rounded Borders** | No | Yes |
| **Grid System and Template Layout** | No | Yes |

4] Name a few CSS style components

* Selector: class name, id name or element name that is target
* Attribute: name of the attribute you want to style, for example border, color, background, position etc.
* Value of Property: value that will be assigned to attribute.
* Wrappers: elements for wrapping general layout.
* Main Sections: root sections of the page.

5] What do you understand by CSS opacity?

The opacity property sets the opacity level for an element.

The opacity-level describes the transparency-level, where 1 is not transparent at all, 0.5 is 50% see-through, and 0 is completely transparent.

When using the opacity property to add transparency to the background of an element, all of its child elements become transparent as well.



6] How can the background color of an element be changed?

he background-color property sets the background color of an element.

The background of an element is the total size of the element, including padding and border (but not the margin).

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <style>

        h1{

            background-color: black;

            color: aliceblue;

        }

    </style>

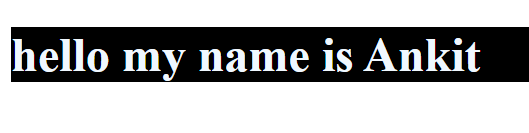
</head>

<body>

 <h1>hello my name is Ankit</h1>

</body>

</html>



7] How can image repetition of the backup be controlled?

This task can be achieved by using the background-repeat property that will help us to control the repetition of the image. The background-repeat property in CSS is used to repeat the background image both horizontally and vertically. It also decides whether the background image will be repeated or not.

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <style>

        h1{

            height: 100px;

            width: 100px;

            background-image: url(data:image/jpeg;base64,);

            background-repeat: no-repeat;

        }

    </style>

</head>

<body>

 <h1> Ankit</h1>

</body>

</html>



8] What is the use of the background-position property?

The background-position property sets the starting position of a background image.

**Tip:**By default, a [background-image](https://www.w3schools.com/cssref/pr_background-image.php) is placed at the top-left corner of an element, and repeated both vertically and horizontally.

How to position a background-image:

body {  
  background-image: url('w3css.gif');  
  background-repeat: no-repeat;  
  background-attachment: fixed;  
  background-position: center;  
}

Value left top

left center

left bottom

right top

right center

right bottom

center top

center center

center bottom If you only specify one keyword, the other value will be "center"

x% y% The first value is the horizontal position and the second value is the vertical. The top left corner is 0% 0%. The right bottom corner is 100% 100%. If you only specify one value, the other value will be 50%. Default value is: 0% 0%

xpos ypos The first value is the horizontal position and the second value is the vertical. The top left corner is 0 0. Units can be pixels (0px 0px) or any other CSS units. If you only specify one value, the other value will be 50%. You can mix % and positions

initial Sets this property to its default value. Read about initial

inherit Inherits this property from its parent element. Read about inherit.

Example

How to position a background-image to be centered at top:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <style>

        body{

            background-image: url(data:image/webp;base64,UklGRloXAABXRUJQVlA4IE4XAAAwfQCdASo+AT4BPoE+mkqlIyIpojOamTAQCWdu/Dp7XXVVqVK0jAtV0++jIU+WP62O4vbS4KDOPmmz8lmw1vy9/W/sHfrt1yR6LSlED4YRfpxqPLB//+HzK/wi9nUp7AcqlH0V7Ed+nQlibIJhxgf+Cmy9bsPgfNK8cdTACprtjd9j2+0nHhf4x9cIv4KlzlbzSrjCYx1EQtmmI0OqOW2wFN1H8b7HWk8QCJ8AENPghf0hjq7SFhpBBjowhUBjuzaDfzS+k0

);

            background-repeat: no-repeat;

            background-attachment: fixed;

            background-position: center top;

        }

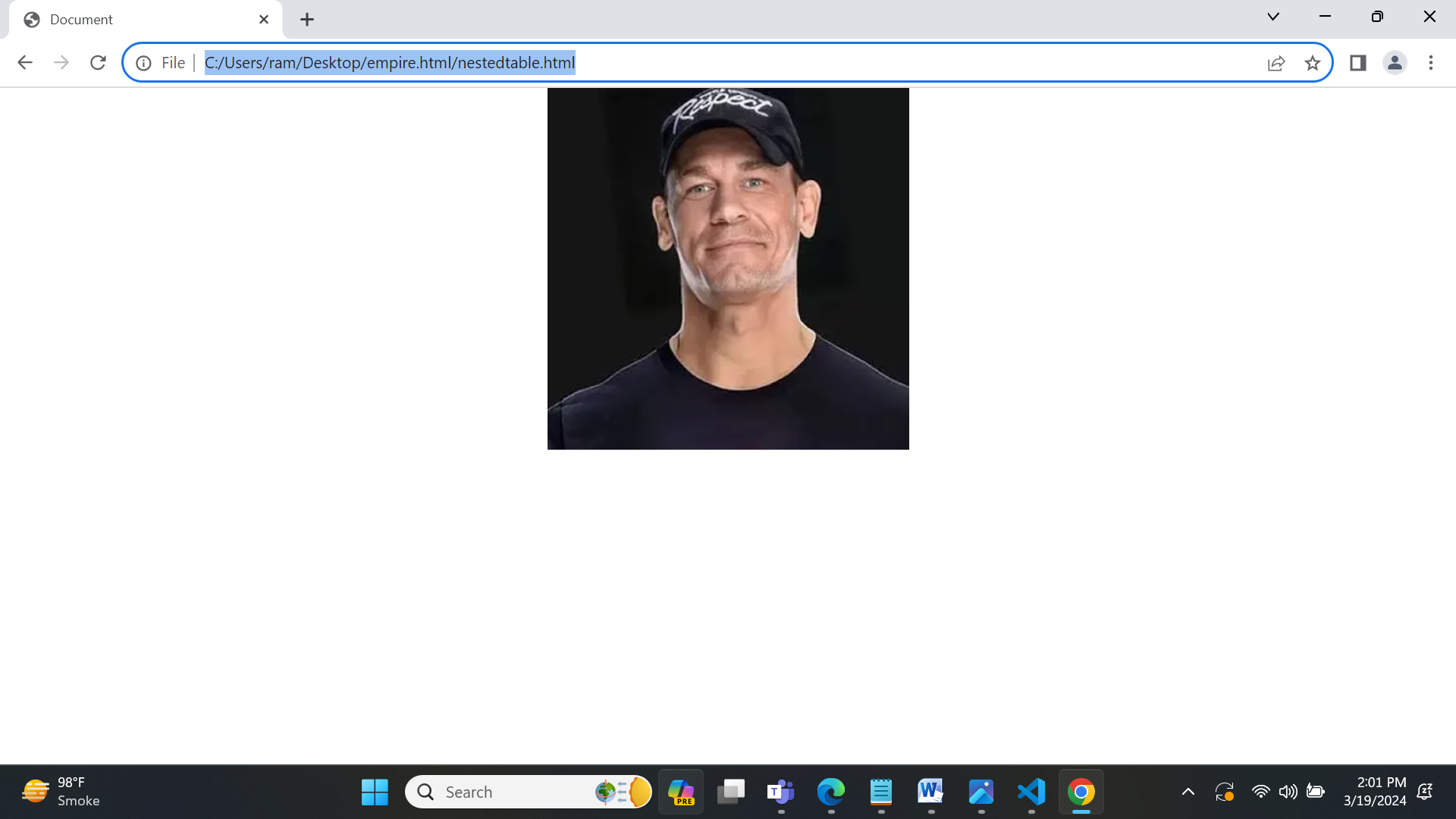
    </style>

</head>

<body>

</body>

</html>



Example

How to position a background-image to be bottom right:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <style>

        body{

            background-image: url(data:image/webp;base64,);

            background-repeat: no-repeat;

            background-attachment: fixed;

            background-position: bottom top;

        }

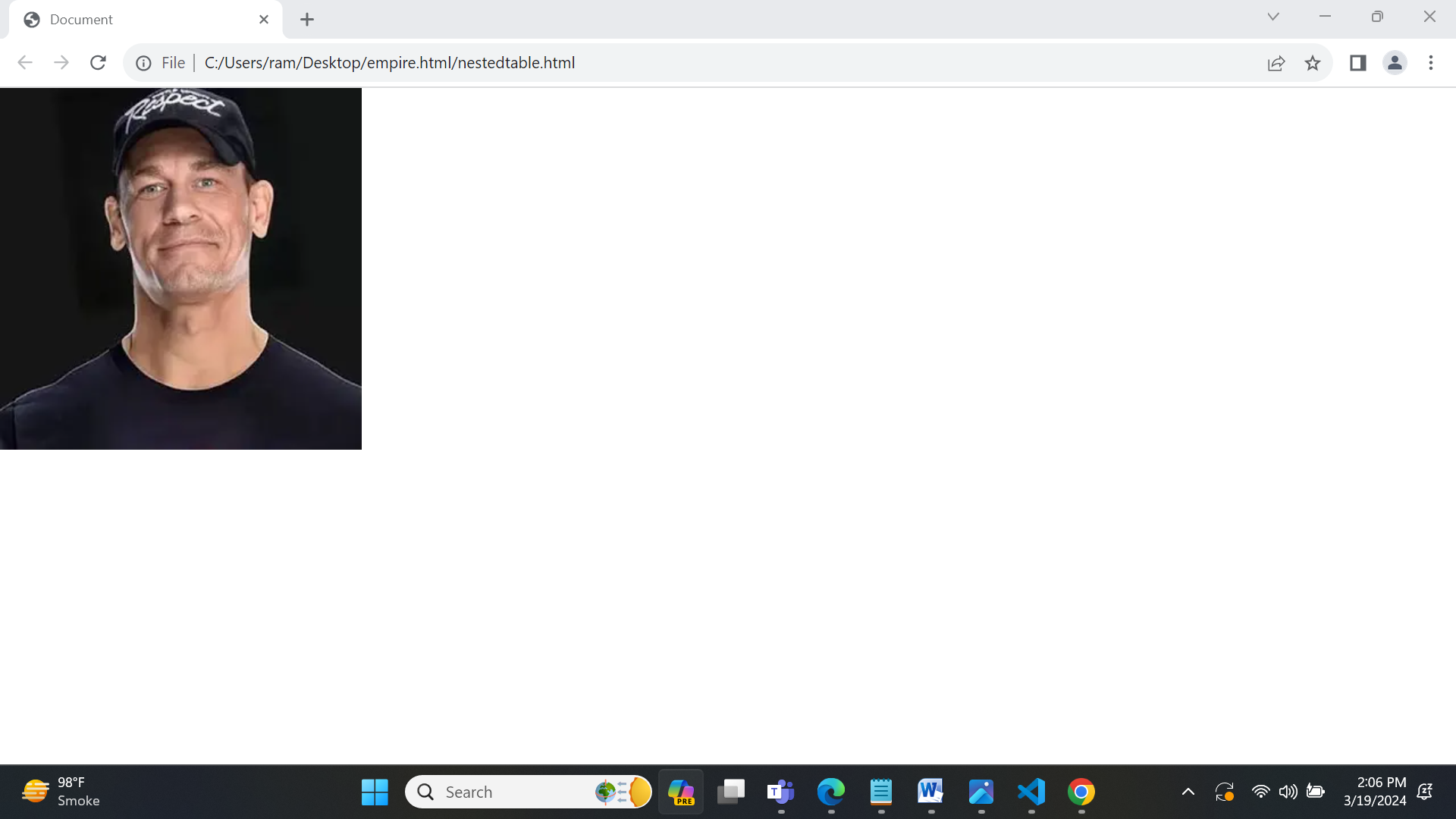
    </style>

</head>

<body>

</body>

</html>



Example

How to position a background-image using percent:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <style>

        body{

            background-image: url(data:image/webp;base64,);

            background-repeat: no-repeat;

            background-attachment: fixed;

            background-position: 50% 50%;

        }

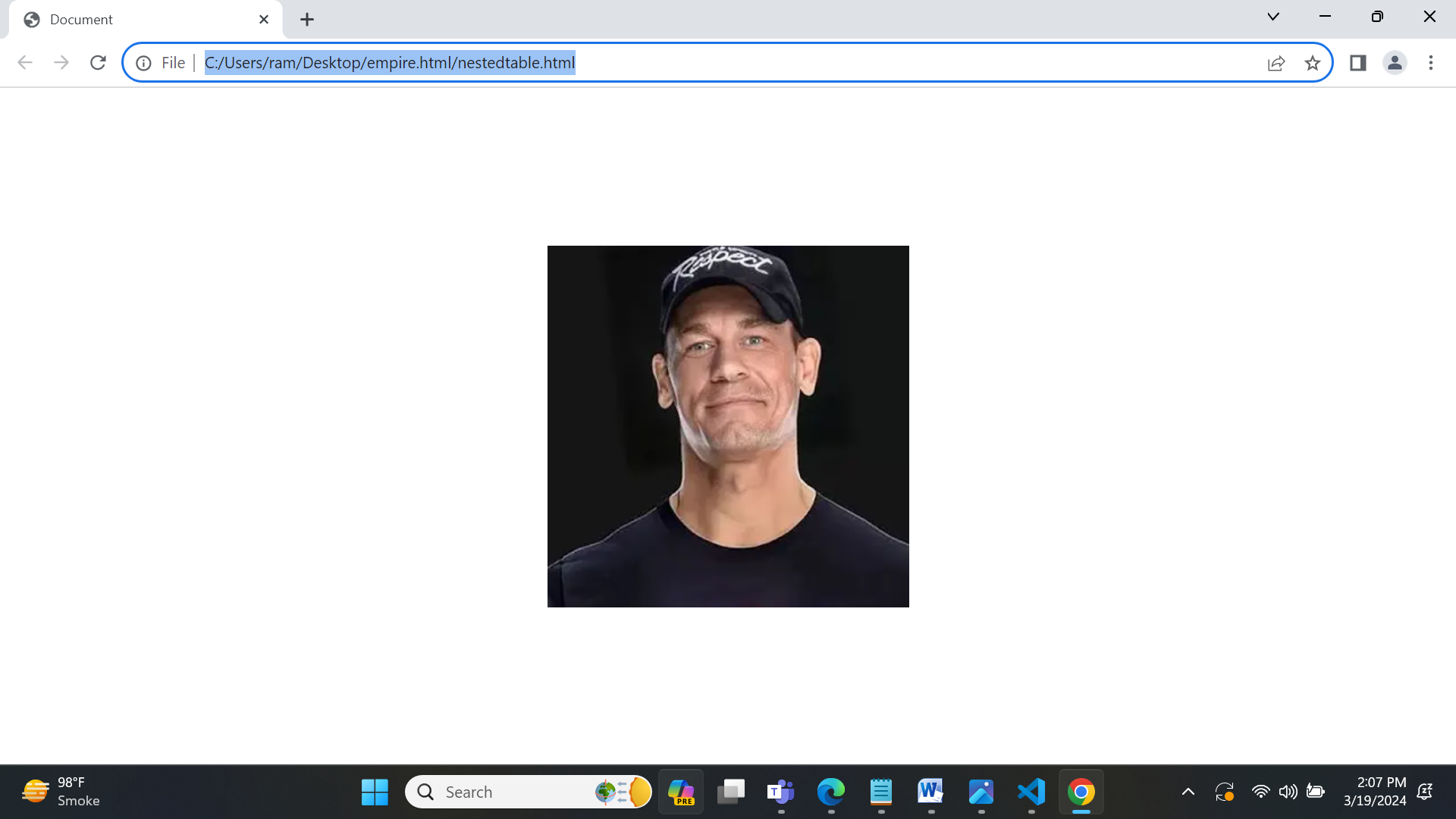
    </style>

</head>

<body>

</body>

</html>



Example

How to position a background-image using pixels:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <style>

        body{

            background-image: url(data:image/webp;base64,);

            background-repeat: no-repeat;

            background-attachment: fixed;

            background-position: 50px 150px;

        }

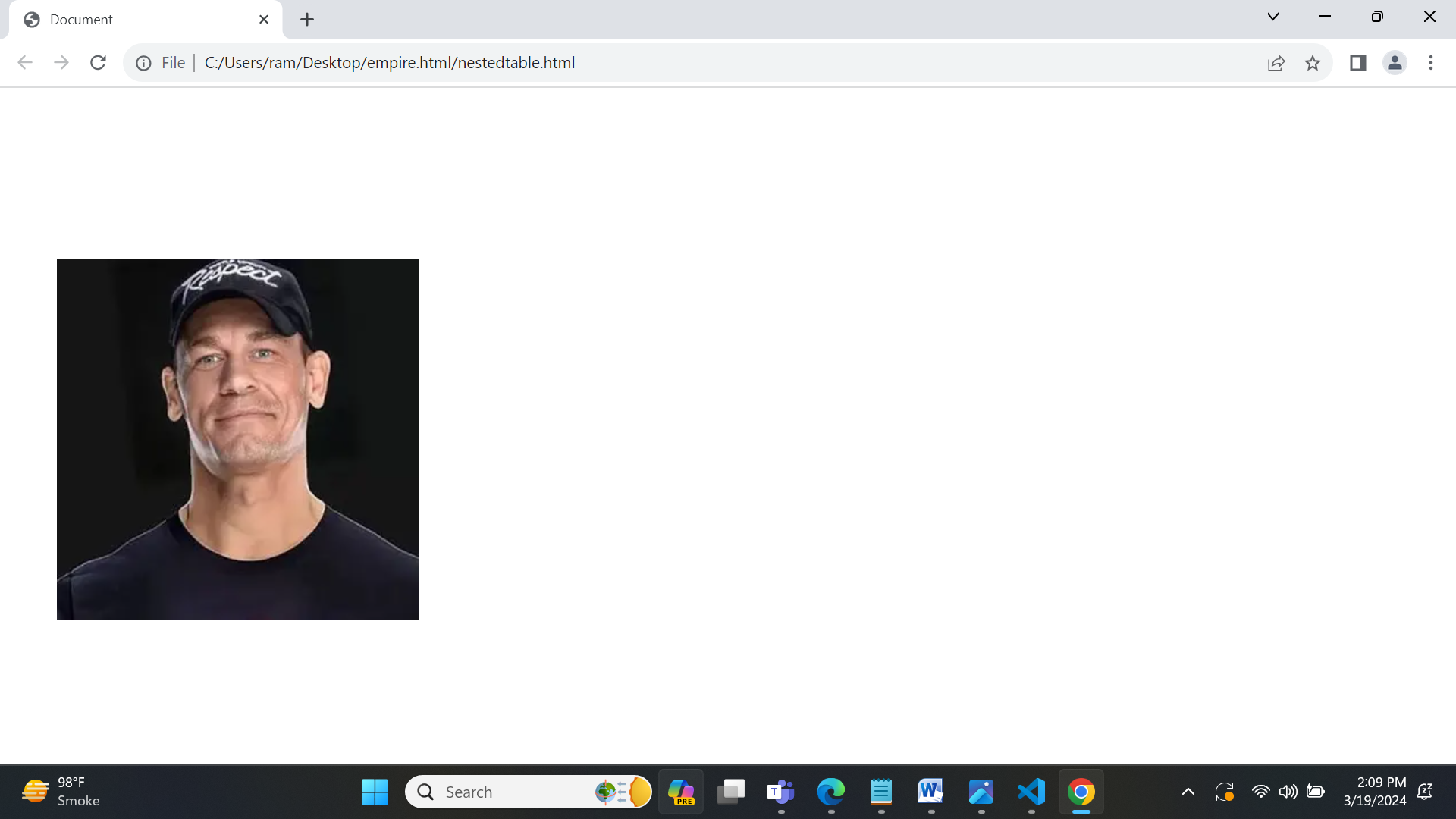
    </style>

</head>

<body>

</body>

</html>



9] Which property controls the image scroll in the background?

CSS background-attachment property.

A background-image that will not scroll with the page (fixed):

body {

background-image: url("img\_tree.gif");

background-repeat: no-repeat;

background-attachment: fixed;

}

The background-attachment property sets whether a background imagescrolls with the rest of the page, or is fixed.

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <style>

        body{

            background-image: url(data:image/webp;base64,);

            background-repeat: no-repeat;

            background-attachment: fixed;

        }

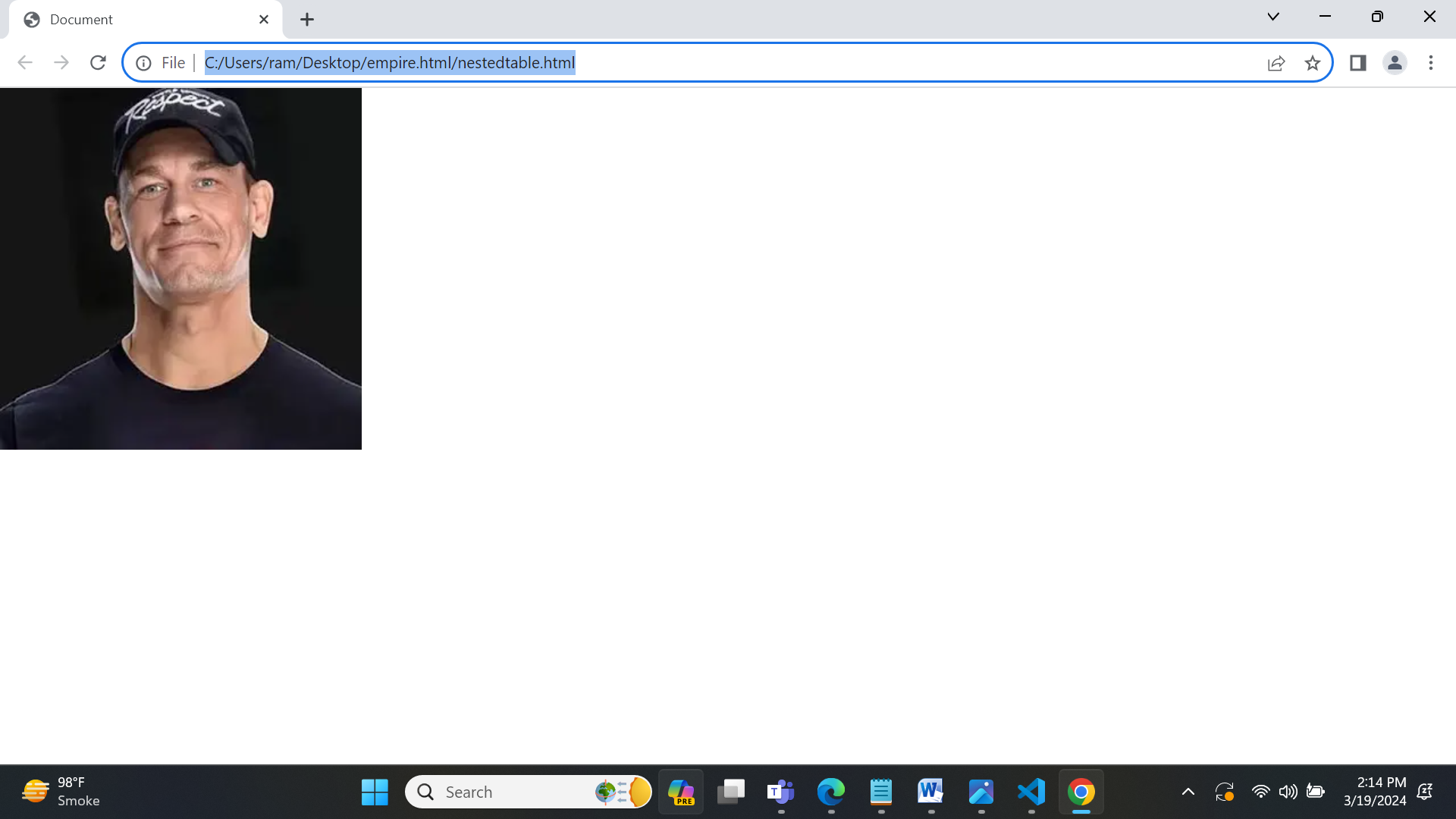
    </style>

</head>

<body>

</body>

</html>



10] Why should background and color be used as separate properties?

Using background and color as separate properties allows for more flexibility and controls in styling elements. It helps in easily changing the background or text color without affecting other properties.

11] How to center block elements using CSS1?

To center block elements, we primarily rely on the `margin` property. Since block elements span the full width of their container, we adjust their margins to position them horizontally and vertically. Notably, the `text-align: center` property doesn’t center block elements; it only affects non-block or inline-block elements.

1. Center block elements using margin property:

We need to specify the margin from left and right such that it looks centered. We do not need to do this manually, we have one property value “auto” which will automatically set the margin such that our block element is placed in the center. Use the below CSS property to center your block element.

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8" />

    <meta http-equiv="X-UA-Compatible" content="IE=edge" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <style>

        h2,

        p {

            text-align: center;

        }

        .myblock {

            margin: auto;

            border: 2px solid red;

            width: fit-content;

            padding: 15px;

            text-align: center;

            background-color: lightyellow;

        }

        header {

            font-size: 40px;

            background-color: lightgreen;

            margin: auto;

            width: fit-content;

        }

        .myinline {

            padding: 10px;

            border: 2px solid blue;

        }

        .holder {

            text-align: center;

        }

    </style>

</head>

<body>

    <header>hello</header>

    <div class="myblock">

        div who has default display : block

    </div>

    <div class="holder">

        <div style="display: inline-block" class="myinline">

            inline block paragraph 1

        </div>

        <div style="display: inline-block" class="myinline">

            inline block paragraph 2

        </div>

    </div>

</body>

</html>



2. Centering Element Using display block Property:

We have one image that has some space around it, so by default the non-block element will come next to the img tag and not on the next line. After setting the “display: block” property, we can make our image to block element. It can be centered using “margin: auto” property.

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8" />

    <meta http-equiv="X-UA-Compatible" content="IE=edge" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <style>

        header {

            font-size: 20px;

            margin: auto;

            width: 30%;

            background-color: lightgreen;

            margin-bottom: 10px;

        }

        p {

            display: inline-block;

        }

        img {

            display: block;

            margin: auto;

        }

    </style>

</head>

<body>

    <header>

        centering image using display: block

    </header>

    <img src="https://tse1.mm.bing.net/th?id=OIP.\_YYy6s8YyeN8QARP4vGIdQHaEK&pid=Api&P=0&h=220" alt="">

    <p>

        paragraph came to the new line even

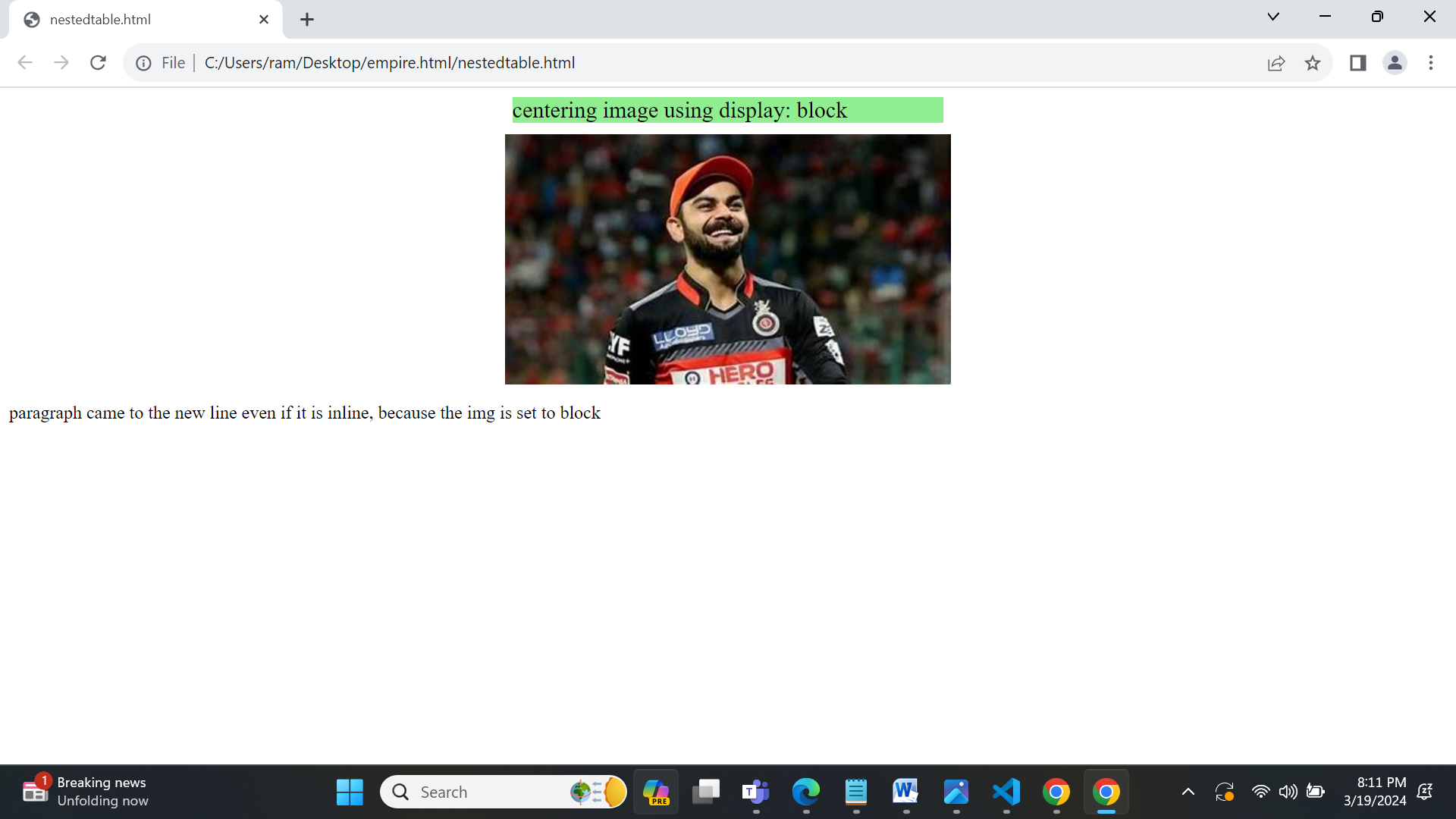
        if it is inline, because the img is

        set to block

    </p>

</body>

</html>



12] How to maintain the CSS specifications?

To maintain CSS specifications, it's important to follow the standard syntax and rules defined by CSS. Make sure to use proper selectors, property-value pairs, and adhere to the correct formatting. Regularly check for any updates or changes to the CSS specifications to stay up-to-date.

Some common CSS selectors include:

- Element selectors (e.g., p, h1, div)

- Class selectors (e.g., .my-class)

- ID selectors (e.g., #my-id)

- Attribute selectors (e.g., [type="text"])

- Pseudo-class selectors (e.g., :hover, :nth-child(2))

- Pseudo-element selectors (e.g., ::before, ::after)

These selectors allow you to target specific elements or groups of elements in your HTML to apply CSS styles.

13] What are the ways to integrate CSS as a web page?

There are a few ways you can integrate CSS into a web page. The most common methods are:

1. Inline CSS: You can apply CSS styles directly to individual HTML elements using the style attribute. For example:

html

<p style="color: blue;">This text is blue.</p>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

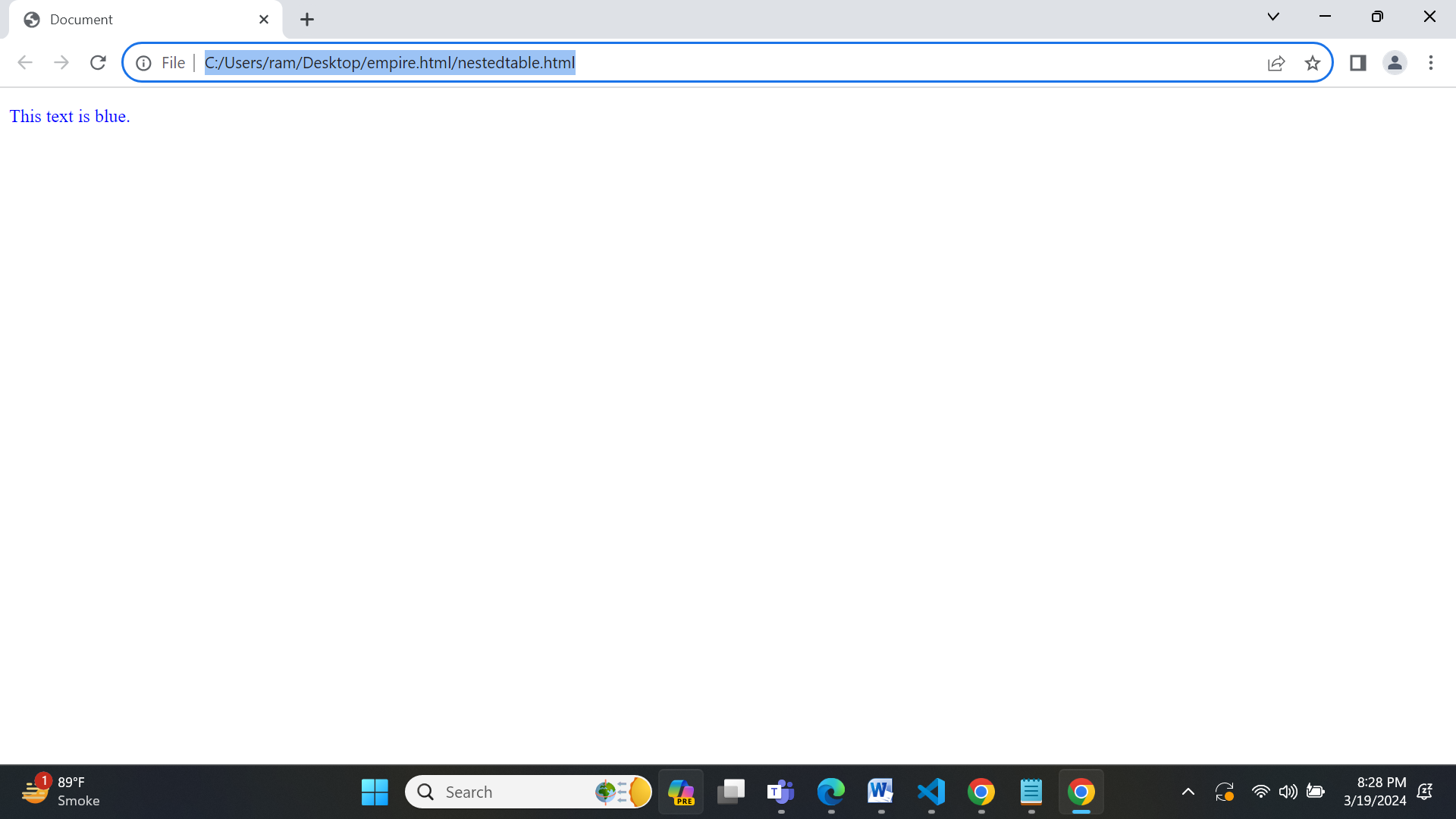
</head>

<body>

    <p style="color: blue;">This text is blue.</p>

</body>

</html>



2. Internal CSS: You can include CSS styles within the <style> tags in the <head> section of your HTML document. For example:

html

<head>

<style>

p {

color: blue;

}

</style>

</head>

<body>

<p>This text is blue.</p>

</body>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <style>

        p{

            color: red;

          text-align: center;

        }

    </style>

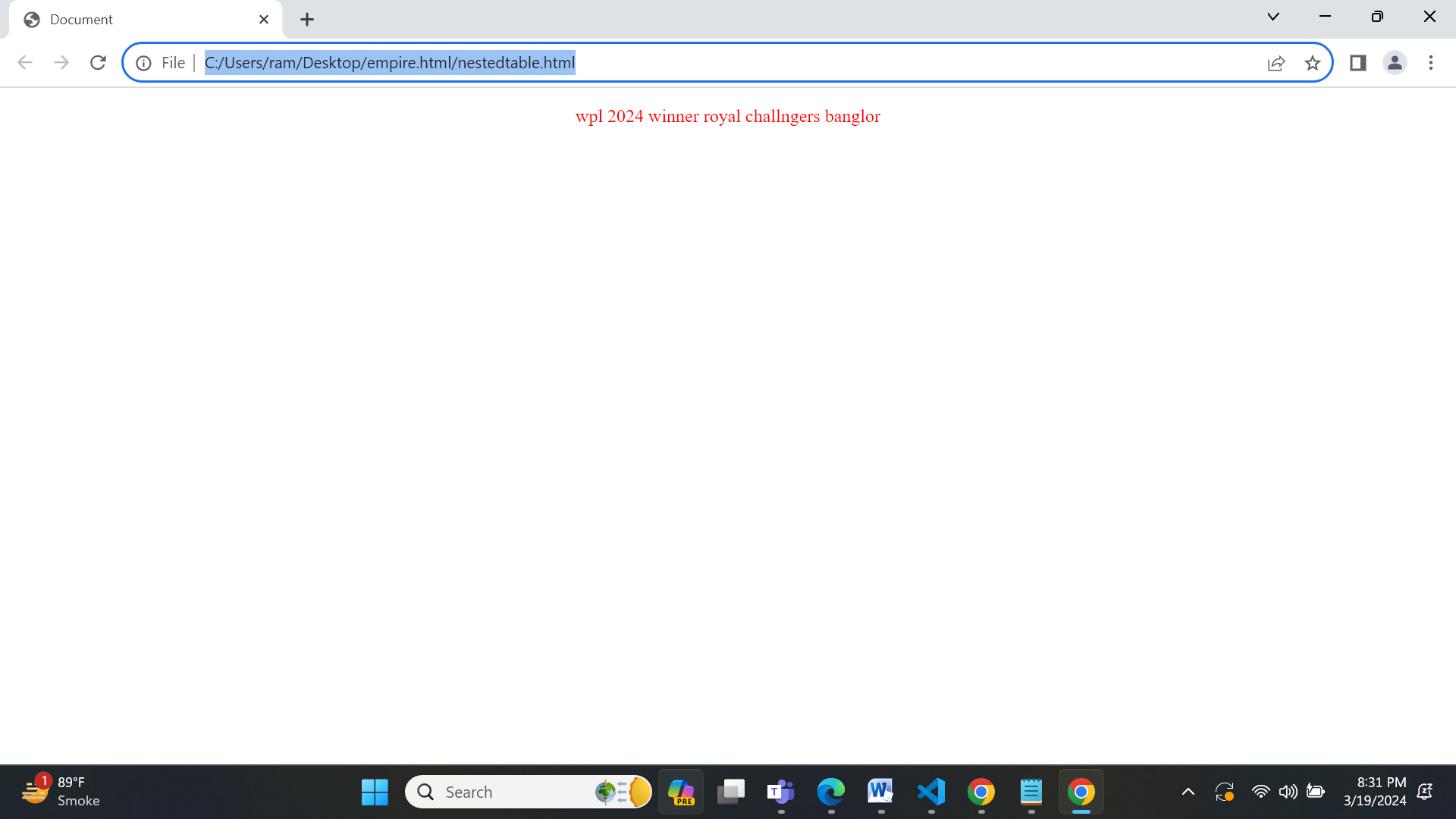
</head>

<body>

    <p>wpl 2024 winner royal challngers banglor</p>

</body>

</html>



3. External CSS: You can link an external CSS file to your HTML document using the <link> tag. The CSS styles are then defined in a separate .css file. For example:

html

<head>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<p>This text is styled using CSS from an external file.</p>

</body>

Html file

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="ak.css">

</head>

<body>

    <h1>EMPIRE ANKIT</h1>

</body>

</html>

css

h1{

    color: blueviolet;

}

These methods provide different levels of flexibility and organization when working with CSS. Let me know if you need more information!

13] What is embedded style sheets?

Embedded Stylesheet: It allows you to define styles for a particular HTML document as a whole in one place. This is done by embedding the <style></style> tags containing the CSS properties in the head of your document. Embedded style sheets are particularly useful for HTML documents that have unique style requirements from the rest of the documents in your project. However, if the styles need to be applied across multiple documents, you should link to an external style sheet instead of using individual embedded style sheets. Using embedded stylesheets holds a distinct advantage over inline styles which only allow you to address one HTML element at a time.

Syntax: The CSS syntax for embedded style sheets is exactly the same as other CSS code, apart from the fact that it is now wrapped within the <style></style> tags. The <style> tag takes the ‘type’ attribute that defines the type of style sheet being used (ie. text/CSS).

Example 1: Below is an HTML document with the CSS styling for the entire web page enclosed within the <style></style> tags. These properties would be applied to all corresponding elements in the HTML document.

<!DOCTYPE html>

<html>

<head>

    <title>Page Title</title>

    <style>

        h2 {

            font-size: 1.5rem;

            color: #2f8d46;

            text-align: center;

        }

        p {

            font-variant: italic;

        }

    </style>

</head>

<body>

    <h2>Welcome To hell</h2>

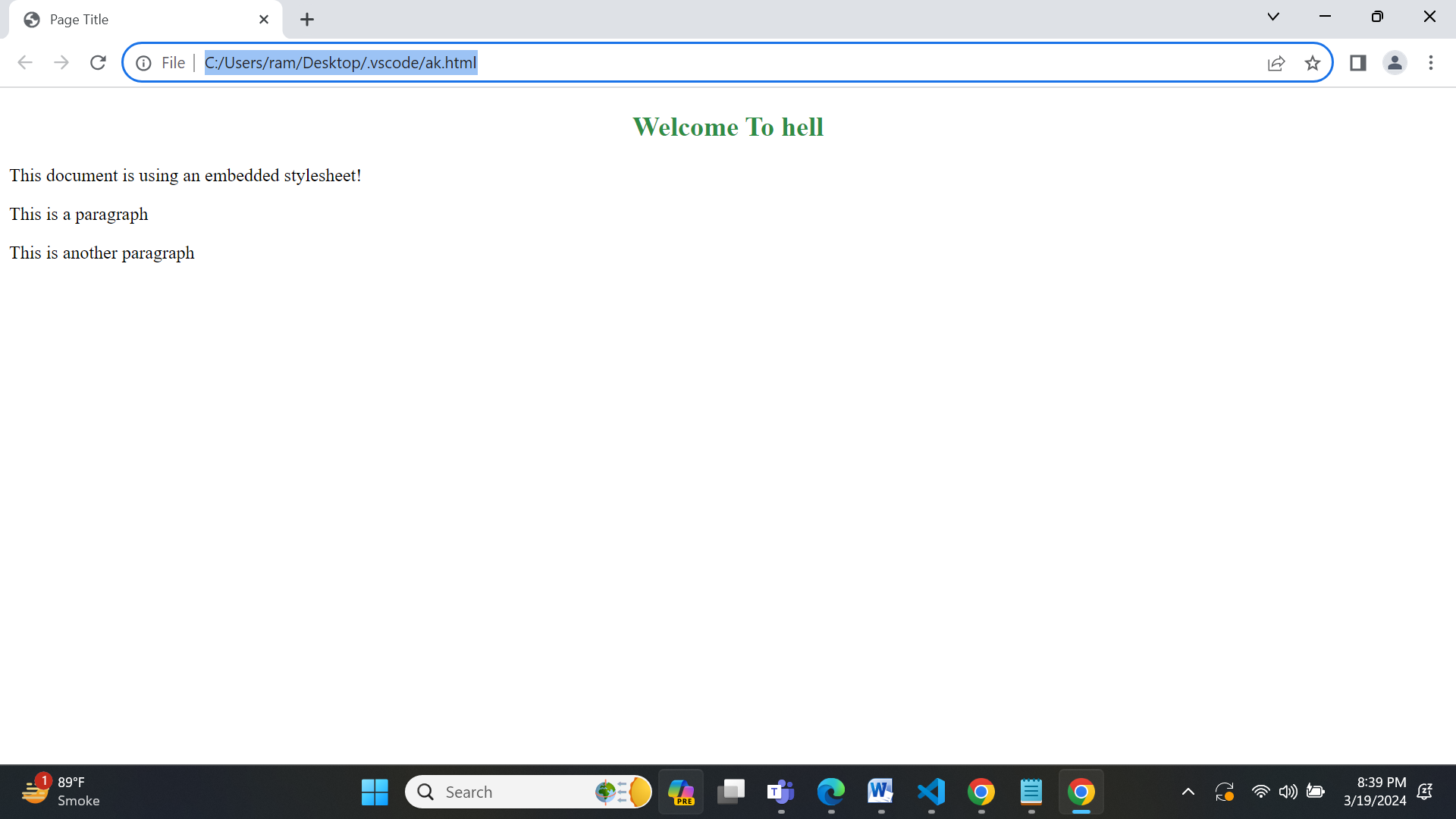
    <p>This document is using an embedded stylesheet!</p>

    <p>This is a paragraph</p>

    <p>This is another paragraph</p>

</body>

</html>



Example 2: When the list of CSS rule sets is inserted in the style element, it will apply the associated properties to all elements on the web page. In case you want to be more selective and distinctly style an element or a group of elements, use classes and IDs, as shown below.

<!DOCTYPE html>

<html>

<head>

    <title>Page Title</title>

    <!-- Embedded stylesheet -->

    <style>

        h2 {

            font-size: 1.5rem;

            color: #2f8d46;

            text-align: center;

        }

        .p-content {

            font-variant: italic;

        }

    </style>

</head>

<body>

    <h2>Welcome To ahemdabad</h2>

    <p class="p-content">

        This document is using an

        embedded stylesheet!

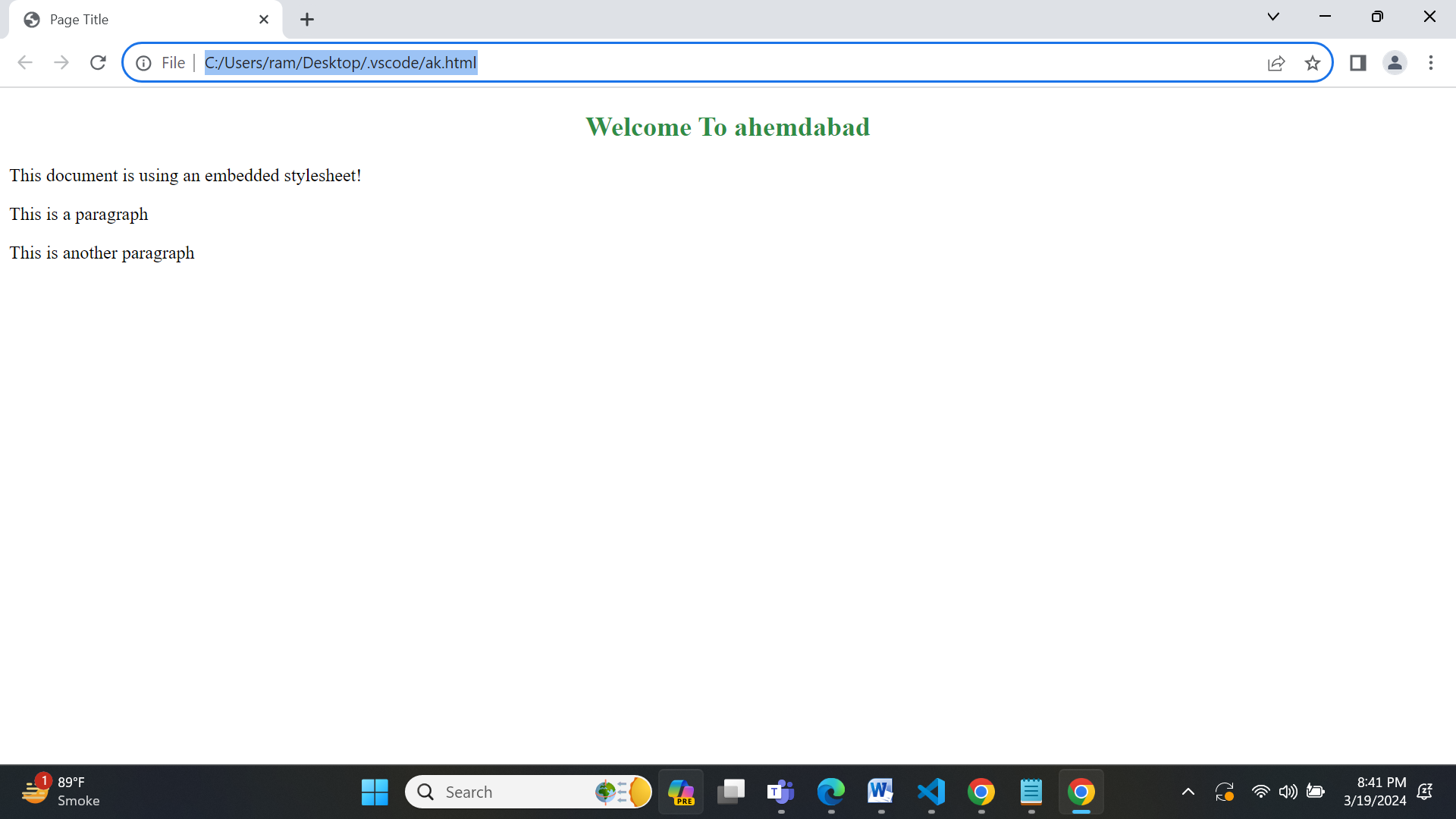
    </p>

    <p>This is a paragraph</p>

    <p>This is another paragraph</p>

</body>

</html>



14] What are the external style sheets?

External CSS: You can link an external CSS file to your HTML document using the <link> tag. The CSS styles are then defined in a separate .css file. For example:

html

<head>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<p>This text is styled using CSS from an external file.</p>

</body>

Html file

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="ak.css">

</head>

<body>

    <h1>hello Empire</h1>

</body>

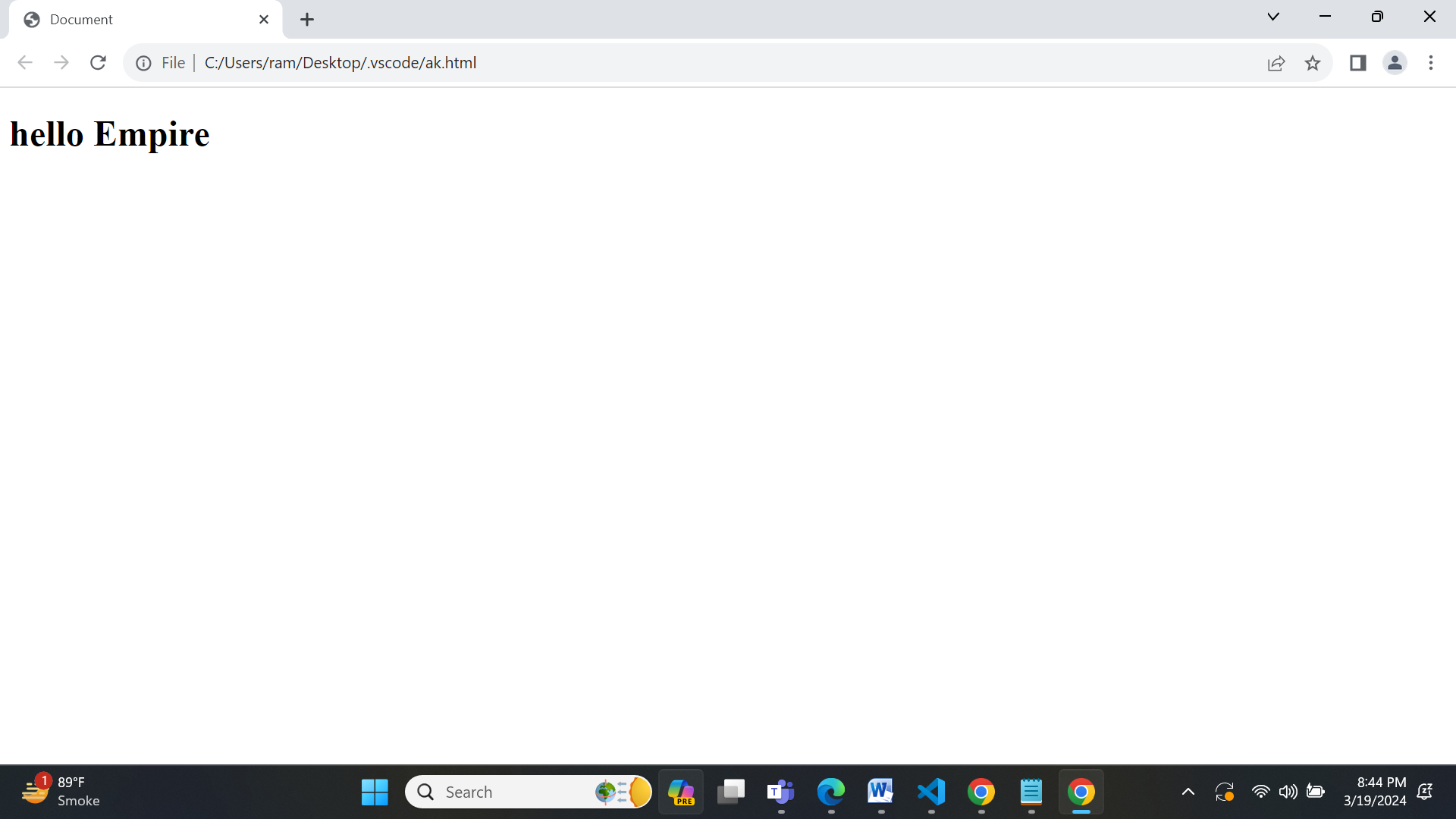
</html>

css

h1{

    color: blueviolet;

}



15] What are the advantages and disadvantages of using external style sheets?

The advantages of External Style Sheets are:

- Using them, the styles of multiple documents can be controlled from one file.

- Classes can be created for use on multiple HTML element types in many documents.

- In complex situations, selector and grouping methods can be used to apply styles.

The disadvantages of External Style Sheets are:

- In order to import style information for each document, an extra download is needed.

- Until the external style sheet is loaded, it may not be possible to render the document.

- For small number of style definitions, it is not viable.

16] What is the meaning of the CSS selector?

What is a selector?

A CSS selector is the first part of a CSS Rule. It is a pattern of elements and other terms that tell the browser which HTML elements should be selected to have the CSS property values inside the rule applied to them. The element or elements which are selected by the selector are referred to as the subject of the selector.

If you have more than one thing which uses the same CSS then the individual selectors can be combined into a selector list so that the rule is applied to all of the individual selectors.

h1 {

color: blue;

}

.special {

color: blue;

}

I could also combine these into a selector list, by adding a comma between them.

h1, .special {

color: blue;

}

White space is valid before or after the comma. You may also find the selectors more readable if each is on a new line.

h1,

.special {

color: blue;

}

Types of selectors

There are a few different groupings of selectors, and knowing which type of selector you might need will help you to find the right tool for the job. In this article's subarticles we will look at the different groups of selectors in more detail.

Type, class, and ID selectors

Type selectors target an HTML element such as an <h1>:

CSS

Copy to Clipboard

h1 {

}

Class selectors target an element that has a specific value for its class attribute:

CSS

Copy to Clipboard

.box {

}

ID selectors target an element that has a specific value for its id attribute:

CSS

Copy to Clipboard

#unique {

}

Attribute selectors

This group of selectors gives you different ways to select elements based on the presence of a certain attribute on an element:

CSS

Copy to Clipboard

a[title] {

}

Or even make a selection based on the presence of an attribute with a particular value:

CSS

Copy to Clipboard

a[href="https://example.com"]

{

}

Pseudo-classes and pseudo-elements

This group of selectors includes pseudo-classes, which style certain states of an element. The :hover pseudo-class for example selects an element only when it is being hovered over by the mouse pointer:

CSS

Copy to Clipboard

a:hover {

}

It also includes pseudo-elements, which select a certain part of an element rather than the element itself. For example, ::first-line always selects the first line of text inside an element (a <p> in the below case), acting as if a <span> was wrapped around the first formatted line and then selected.

CSS

Copy to Clipboard

p::first-line {

}

17] What are the media types allowed by CSS?

CSS Media Queries

The @media rule, introduced in CSS2, made it possible to define different style rules for different media types.

Media queries in CSS3 extended the CSS2 media types idea: Instead of looking for a type of device, they look at the capability of the device.

Media queries can be used to check many things, such as:

width and height of the viewport

orientation of the viewport (landscape or portrait)

resolution

Using media queries are a popular technique for delivering a tailored style sheet to desktops, laptops, tablets, and mobile phones (such as iPhone and Android phones).

CSS Media Types

Value Description

All Used for all media type devices

Print Used for print preview mode

Screen Used for computer screens, tablets, smart-phones etc.

You can also link to different stylesheets for different media and different widths of the browser window.

<link rel="stylesheet" media="print" href="print.css">

<link rel="stylesheet" media="screen" href="screen.css">

<link rel="stylesheet" media="screen and (min-width: 480px)" href="example1.css">

<link rel="stylesheet" media="screen and (min-width: 701px) and (max-width: 900px)" href="example2.css">

etc....

320px — 480px: Mobile devices

481px — 768px: iPads, Tablets

769px — 1024px: Small screens, laptops

1025px — 1200px: Desktops, large screens

1201px and more —  Extra large screens, TV

Example: if the viewport is less than 480 pixels, the background-color will be pink.

@media screen and (min-width: 480px) {

body {

background-color: pink;

}

}

18] What is the rule set?

A CSS ruleset is various affirmations to various pieces or elements of the document. The objective is to apply a bunch of properties for certain distinct qualities to a solitary, or a particular arrangement of components in the connected HTML page.

Example 1:

<!DOCTYPE html>

<html>

<head>

  <title>Rulesets in CSS</title>

  <style>

    h1 {

      color: yellow;

    }

    p:first-child{

      background-color: red;

      color: white;

      font-size: 15px;

      border-radius: 50px

      ;

      text-transform: uppercase

      ;

      font-weight: bold;

    }

    body {

      text-align: center;

    }

  </style>

</head>

<body>

  <div>

    <p>royal challngers banglore</p>

    <h1>virat kholi</h1>

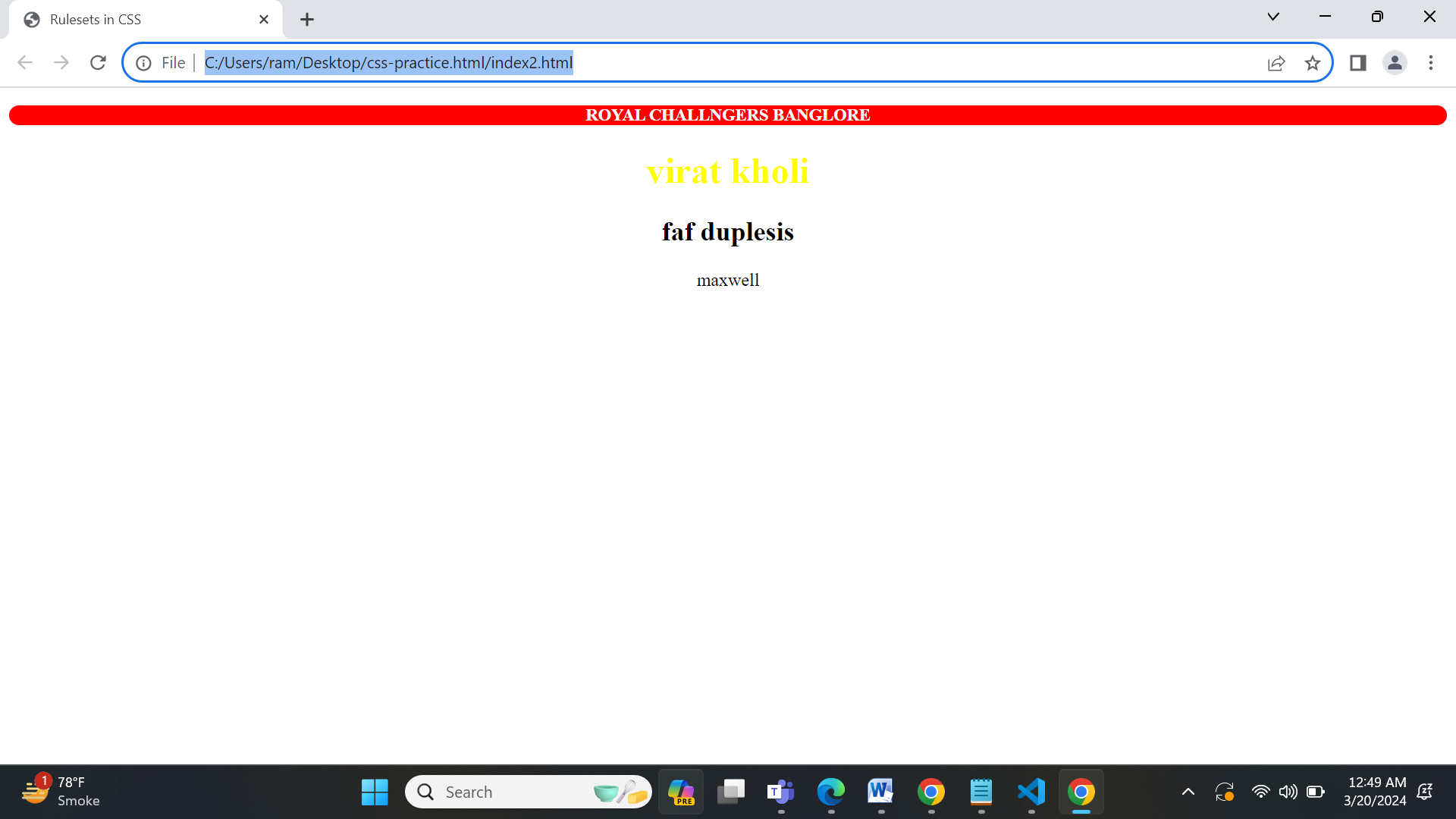
    <h2>faf duplesis</h2>

    <p>maxwell </p>

  </div>

</body>

</html>



EXAMPLE 2:

<!DOCTYPE html>

<html>

<head>

  <title>Rulesets in CSS</title>

  <style>

  h1 {

    color: green;

  }

  span {

    background-color: red;

    color: white;

    padding: 5px;

    font-size: 15px;

    border-radius: 50px;

    text-transform: uppercase;

  }

  h2 {

    font-size: small;

  }

  .divClass {

    font-family: "Franklin Gothic Medium",

          "Arial Narrow", Arial, sans-serif;

  }

  body {

    text-align: center;

  }

  </style>

</head>

<body>

  <div class="divClass">

  <span> Empire Ankit </span>

  <h1>Empire ARmy</h1>

  <h2>

    hello everyone how are you <br />

    i dont care

  </h2>

  </div>

</body>

</html>

