Basic web page development in HTML5: html, CSS, javascript

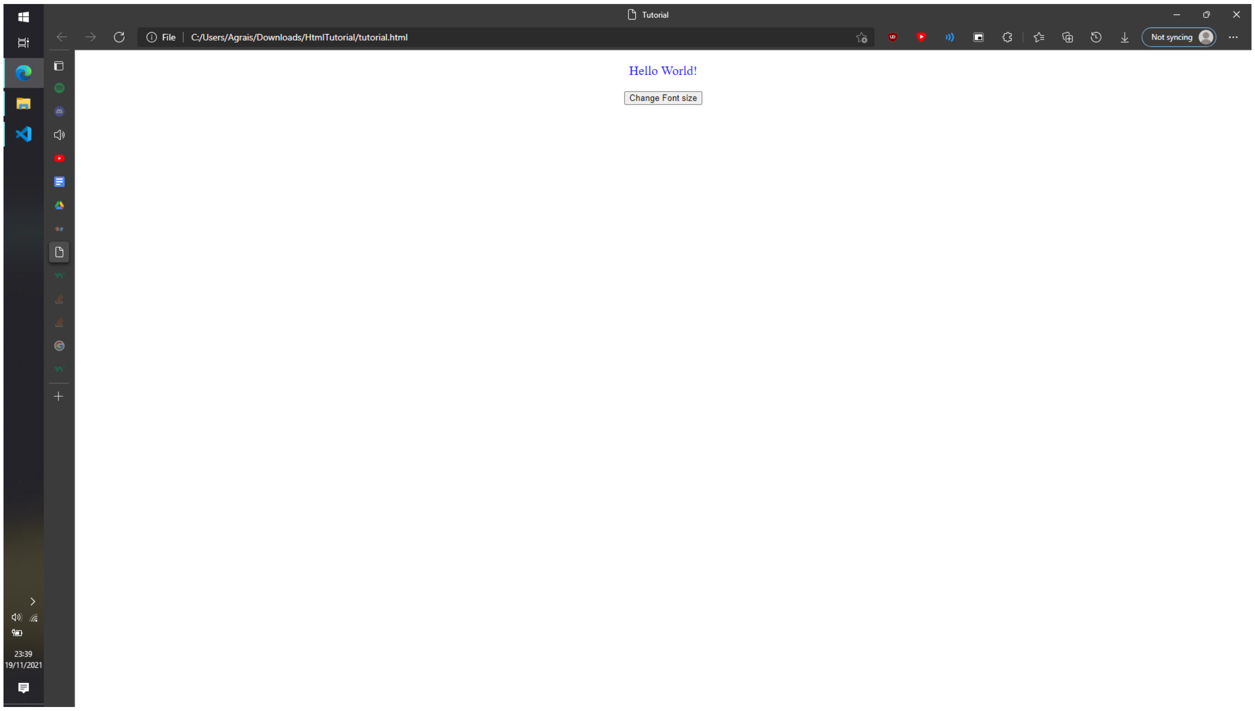
How html, CSS and javascript work together

Webpage development uses HTML5. This is made up of 3 kinds of code, HTML, CSS and Javascript. HTML is a markup language used by browsers to structure the words, images and videos a page will display. CSS is a style sheet language used to describe the presentation of a document written in a markup language, in this case being HTML. Javascript is a scripting language that allows implementation of complex features on web pages. These 3 languages work together to allow web pages to be dynamic and interactive.

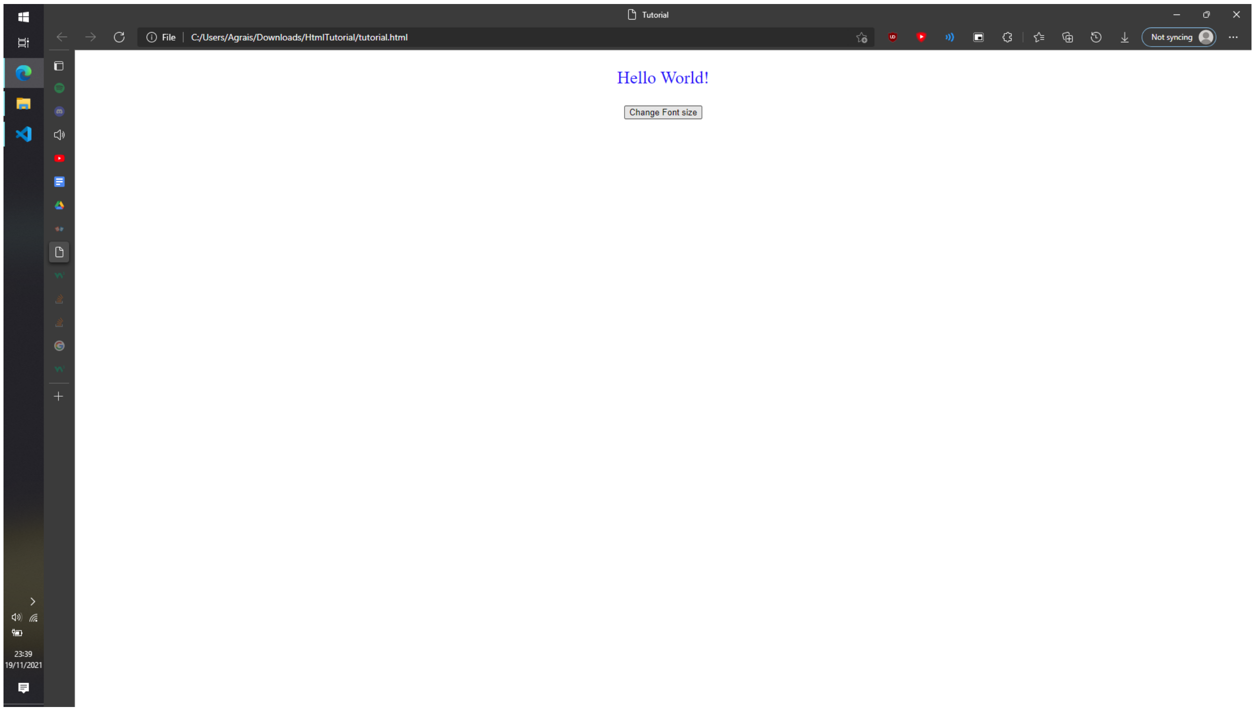
All 3 languages work together with HTML being the main file. Theoretically the HTML file doesn't need to work with any external files as the CSS and JS functionality can be put into the file. However this is bad practice as it leads to non-cohesive code. Therefore the HTML file will contain a link to the CSS file that will be used to style its elements. The CSS file will then reference the objects in the HTML file and the desired styles upon them. In the HTML file javascript code can be placed in <script> tags, or be referenced from external .js files.

Using all 3 languages together allows for dynamic and interactive web pages. This is mainly done through Javascript. Javascript can interact with the webpage via the DOM (document object model) to query the page state and modify. An example would be waiting for a click event on a certain object and modifying the CSS. HTML and CSS are modifiable by Javascribt allowing for dynamic web pages.

Example - Hello world with button program



This example website displays a ‘Hello World!’ text with a button that increases the font size of the text. The image below shows the text increasing in size after clicking the button a few times. This example demonstrates how js code can be used to modify CSS styling to create an interactive website.



Example.html - html file

```
<!DOCTYPE html>
<html>
    <head>
        <link rel="stylesheet" href="styles.css">
        <title>Tutorial</title>
    </head>
    <body>
        <p id="text">Hello World!</p>
        <button id="button" type="button">Change Font size</button>
        <script src="tutorial.js"></script>
    </body>
</html>
```

The html file works based on tags. Each tag has a start and end, for example the paragraph tag <p> Text </p>. The first tag, <!DOCTYPE> tells the browser what version of html is being used. The <html> tag is the root of the file and is the container of all other html elements.

The <head> tag is a container for metadata and is placed between the <html> and <body> tags. In this example I used the <link> tag to define a link to an external style sheet. Then I used a <title> tag for the title which is shown in the pages title bar, this element is required for all html files.

The <body> tag contains all the contents of a html document. The <p> element defines a paragraph where text can be added. Additionally I added an id to this element so that it can be identified in CSS and js for modification. The id attribute is global and can be added to almost all tags. The <button> tag defines a clickable button, with the text of the button between tags. Finally there is a <script> tag. A <script> tag can be used 2 ways, one way is to place js code between the tags, the other is to link towards an external script via the src attribute, which is what is done in the example above. The <script> tag is placed at the end of the code as it manipulates DOM (document object model) and if placed before html elements, they will not be loaded in and return an error when attempted to use/modify.

Style.css - CSS file

```
body {
    text-align:center;
}

#text {
    color: blue;
    font-size: 20px;
}
```

The CSS file works by setting style rules for specific DOM elements. The syntax for styling a DOM element starts with the element then the properties to be styled inside curly brackets. In the example file above, the body tag is styled such that all text contained in the tag is horizontally centered.

The text element to be styled is referenced via the id attribute. The # character signifies that the element to be styled is being referenced via id. Then I apply a blue color and 20px font size to text inside the element.

Not shown here but a group of elements can be referenced by a class attribute, this is done via a . character.

Example.js - js file

```
document.getElementById('button').addEventListener('click', function() {
    let text = document.getElementById('text');
    let newFontSize = parseFloat(getComputedStyle(text).fontSize) + 1;
    text.style.fontSize = newFontSize + 'px';
});
```

The js file makes the website interactive by increasing the font size of the text by clicking the button. This is achieved through js code, first the DOM element for the button is retrieved by calling document.getElementById.

Then to make the button responsive to a click, an eventListener is added to the button which runs a function in response to a click. Here this function is defined as an anonymous function as it doesn't require a name and it is only used once in this eventListener. In the function the DOM element for the text is retrieved in a similar way as the button, then the fontSize is computed and incremented by 1 into a variable. Then the new font size is set to the text. The function getComputedStyle has to be used instead of simply accessing the property as the font size is defined in the CSS style file and therefore accessing the property directly will return " ".