



Javascript

Dr. MURDERER Srivastava



Content

- The Javascript Programming Language
- Objects Attributes and Methods
- HTML Elements for Interaction
- Javascript Events
- Creating elements on the fly
- Iteration
- Selection
- Guessing Game
- Helper Objects
- Asynchronous Programming



The Javascript Programming Language

A decorative network diagram in the top right corner, consisting of various sized circles (nodes) connected by thin lines (edges). Some nodes are solid grey, while others are hollow white with a grey outline. The connections form a complex, interconnected web.

HTML and CSS programming languages do not have nearly the power of Javascript.

Javascript is a procedural language.


With HTML and especially CSS the order of things is not always important, but with Javascript, the order that you do things in is very important.

A decorative network diagram in the bottom left corner, similar to the one in the top right, featuring nodes of different sizes and colors (solid grey and hollow white) connected by lines.

A decorative network diagram in the top right corner, consisting of various sized circles (nodes) connected by thin lines (edges). Some nodes are solid grey, while others are hollow white with a grey border. The connections form a complex, interconnected web.

Javascript is also called an object oriented language.

That means that it will be working with objects in the world of the browser.

A decorative network diagram in the bottom left corner, similar to the one in the top right, featuring a cluster of interconnected nodes and lines.

```
<html>
  <body>
    <h1> Hello World!!</h1>
    <button onclick = "changeThisPageFunc();"> Click Me!</button>
    <script type = "text/javascript">
      changeThisPageFunc = function() {
        document.body.style.backgroundColor = "#734c78";
        document.body.innerHTML = "<h1> I am a little colorful today.</h1>";
      }
    </script>
  </body>
</html>
```



Hello World!!

Click Me!

I am a little colorful today.

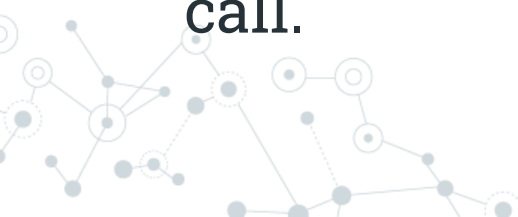
A decorative network diagram in the top right corner, consisting of various sized grey circles connected by thin grey lines, some of which are dashed.

The button tag is part of HTML that is going to allow us to make interactive websites.

In addition to buttons there are other things like checkboxes, and text input.

The onclick attribute is the way we answer the question, “what should this button do when someone clicks on it.”

The way we are going to do this is through a function call.

A decorative network diagram in the bottom left corner, consisting of various sized grey circles connected by thin grey lines, some of which are dashed.



Functions are abstractions of things that need doing.

The script tag is one way that we can include Javascript in our HTML.

But like CSS we can, and should, also include Javascript by putting it in its own file and including it.

The stuff inside the script tag is Javascript code.




A decorative network diagram in the top right corner, consisting of various sized circles (nodes) connected by thin lines (edges). Some nodes are solid grey, while others are hollow white with a grey border. The connections form a complex, interconnected web.

This code contains a function definition which is how we create our very own abstractions.

In the above code, there are two steps to our abstraction.

One changes the background color of the body, and the second one changes our message.

These lines are also abstractions of a very complicated set of steps that we as programmers don't need to know the details of right now.

A decorative network diagram in the bottom left corner, similar to the one in the top right, featuring nodes of different sizes and colors (solid grey and hollow white) connected by thin lines.

We just need to know which abstractions the browser understands.



Change the background color

```
<html>
<body>
  <h1> Hello World!!</h1>
  <button onclick = "changeThisPageFunc();"> Click Me!</button>
  <script type = "text/javascript">
    changeThisPageFunc = function() {
      document.body.style.backgroundColor = "#7da0ad";
      document.body.innerHTML = "<h1> I am a little colorful today.</h1>";
    }
  </script>
</body>
</html>
```



I am a little colorful today.



Change font color using style tag

```
<html>
  <head>
    <style>
      h1 {
        color: #bc80de;
      }
    </style>
  </head>
  <body>
    <h1> Hello World!!</h1>
    <button onclick = "changeThisPageFunc();"> Click Me!</button>
    <script type = "text/javascript">
      changeThisPageFunc = function() {
        document.body.style.backgroundColor = "#7da0ad";
        document.body.innerHTML = "<h1>I am a little colorful today.</h1>";
      }
    </script>
  </body>
</html>
```



Hello World!!

Click Me!

I am a little colorful today.



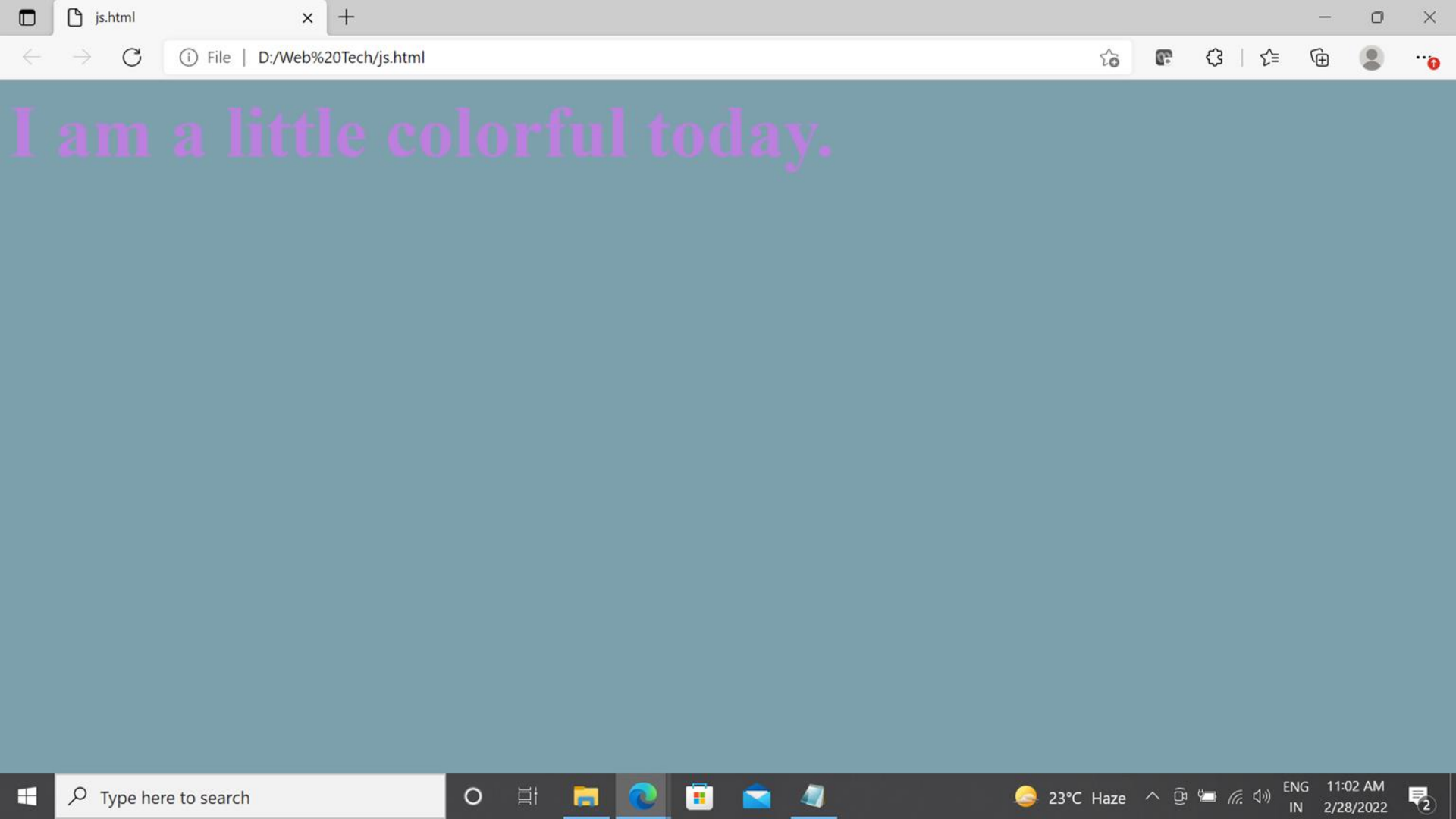
Change font-size to 48pt.

```
<html>
  <head>
    <style>
      h1 {
        color: #bc80de;
        font-size: 48pt;
      }
    </style>
  </head>
  <body>
    <h1> Hello World!!</h1>
    <button onclick = "changeThisPageFunc();"> Click Me!</button>
    <script type = "text/javascript">
      changeThisPageFunc = function() {
        document.body.style.backgroundColor = "#7da0ad";
        document.body.innerHTML = "<h1>I am a little colorful today.</h1>";
      }
    </script>
  </body>
</html>
```


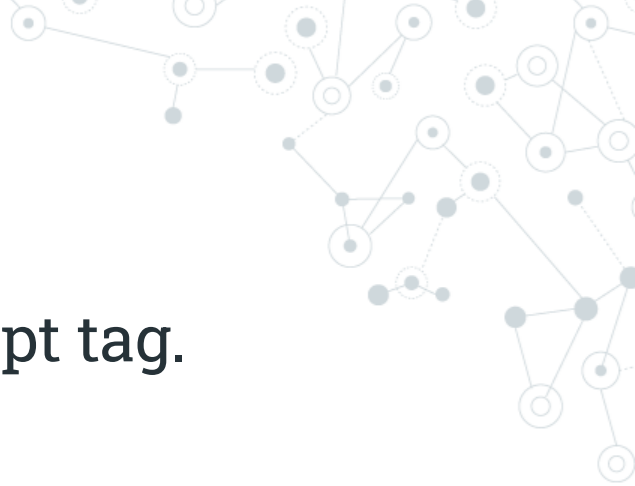


Hello World!!

Click Me!



I am a little colorful today.



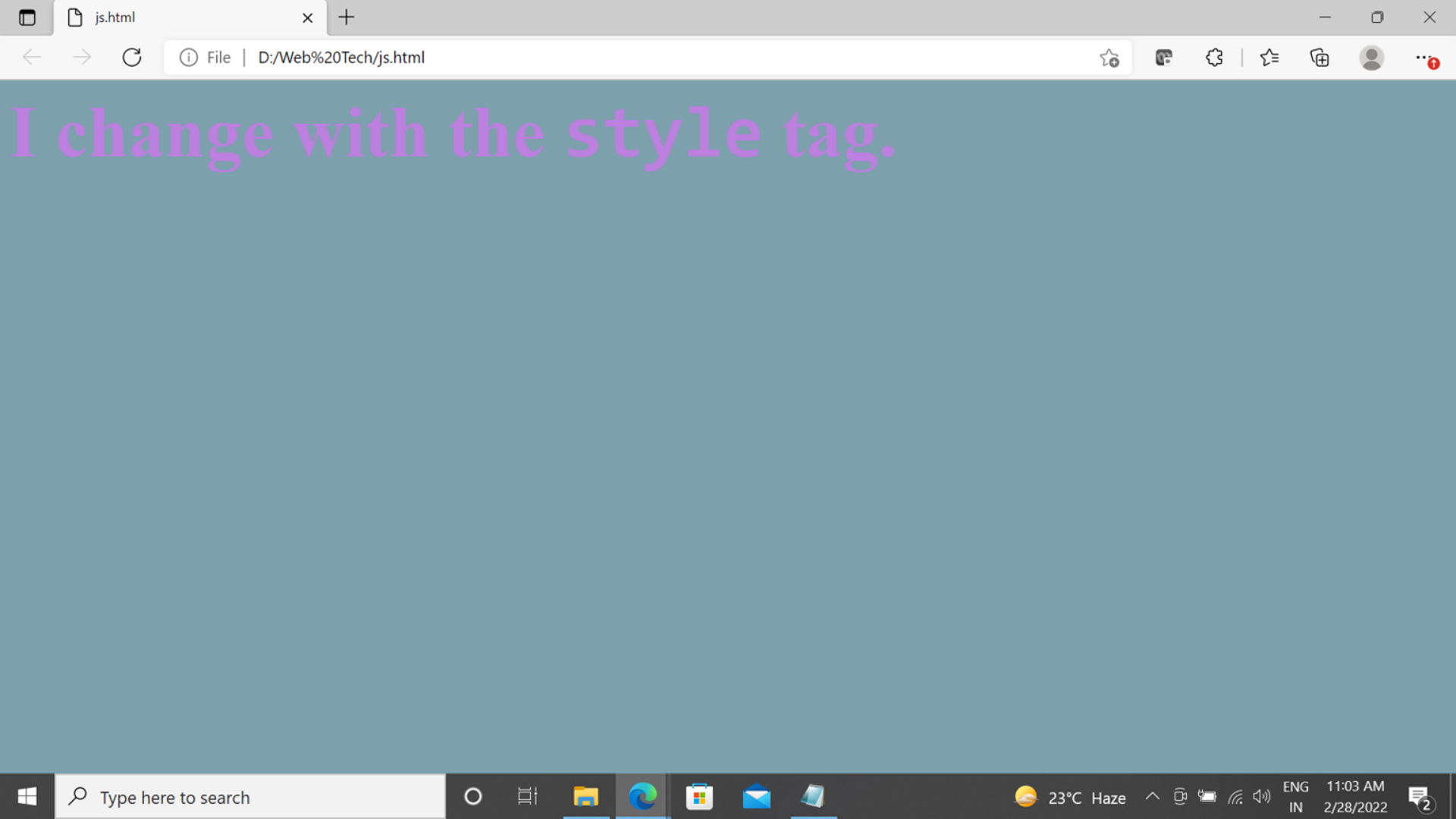
Change content of h1 in script tag.

```
<html>
  <head>
    <style>
      h1 {
        color: #bc80de;
        font-size: 48pt;
      }
    </style>
  </head>
  <body>
    <h1> Hello World!!</h1>
    <button onclick = "changeThisPageFunc();"> Click Me!</button>
    <script type = "text/javascript">
      changeThisPageFunc = function() {
        document.body.style.backgroundColor = "#7da0ad";
        document.body.innerHTML = "<h1>I change with the <code>style</code> tag.</h1>";
      }
    </script>
  </body>
</html>
```



Hello World!!

Click Me!



I change with the style tag.

```
<html>
  <head>
    <style>
      .myclass {
        background-color: #cb03a8;
      }
    </style>
  </head>
  <body>
    <h1> Hello World!!</h1>
    <button onclick = "changeThisPageFunc();"> Click Me!</button>
    <script type = "text/javascript">
      changeThisPageFunc = function() {
        alert("body has class =" + document.body.className);
        document.body.className = "myclass";
        alert(" body has class =" + document.body.className);
        document.body.innerHTML = "<h1>I am a little colorful today.</h1>";
      }
    </script>
  </body>
</html>
```



Hello World!!

Click Me!

Hello World!!

Click Me!

This page says
body has class =

OK

Hello World!!

Click Me!

This page says
body has class =myclass

OK


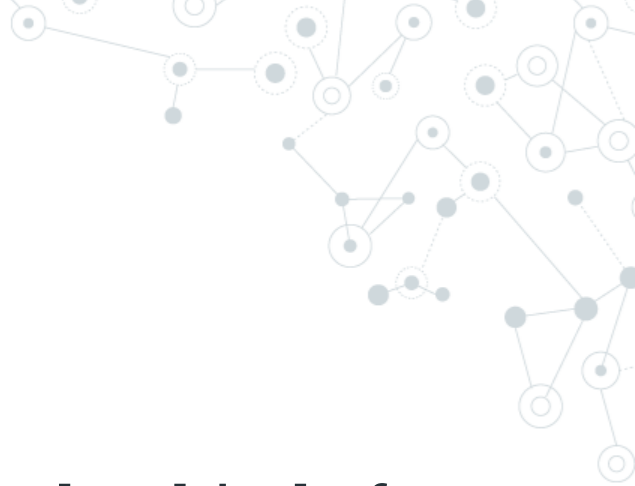
I am a little colorful today.



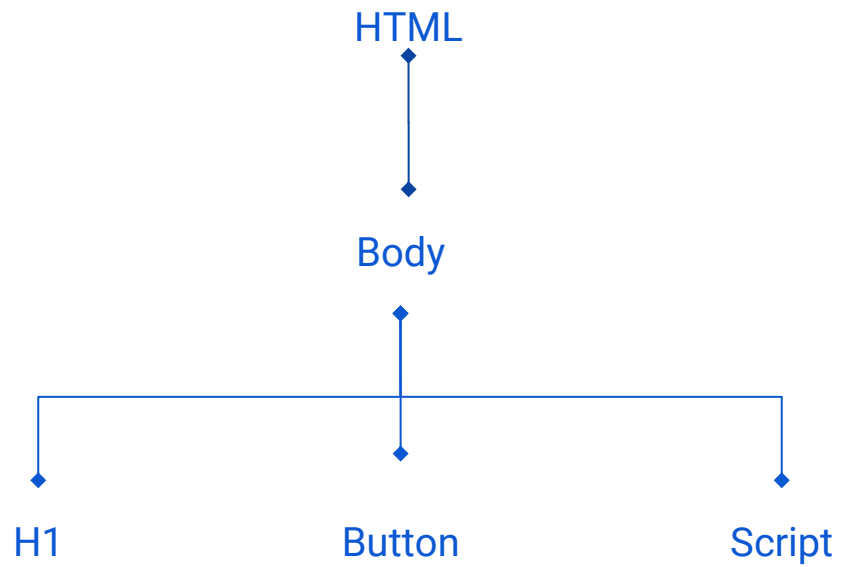
Object Attributes and Methods

```
<html>
  <body>
    <h1> Hello World!!</h1>
    <button onclick = "changeThisPageFunc();" > Click Me!</button>
    <script type = "text/javascript">
      changeThisPageFunc = function() {
        document.body.style.backgroundColor = "#8f2c01";
        document.body.innerHTML = "<h1>I am a little colorful today.</h1>";
      }
    </script>
  </body>
</html>
```





If you think about the HTML only, what kind of tree you get from the document?





document: refers to the entire HTML document (or page).

document is a name that refers to an object.

Object is the representation of the page.

Objects have attributes.


Attributes contain some property of the object.

For example `document.height` tells us how many pixels high the entire page is.






Our example illustrates several other attributes:

- `document.body`
 - `document.body.style`
 - `document.body.style.backgroundColor`
 - `document.body.innerHTML`
- 




We can use attributes in two ways.

- We can use attributes to ask the object a question.
 - We can set attributes to a particular value for the object.
- 

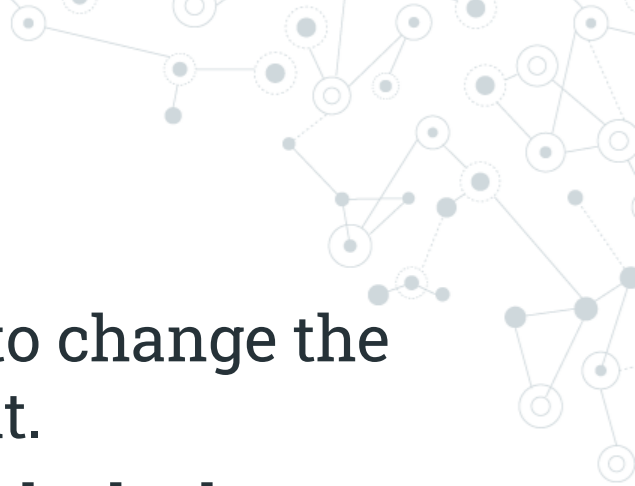


For example:

- `document.body.style.backgroundColor` could be interpreted as asking the document “tell me your background color”.
 - `document.body.style.backgroundColor = "#8f2c01"` should be interpreted as “set your background color to #8f2c01”.
- 


Operators:

- . (dot) is for asking one object to return another object.
- = is for setting the attribute of the object to a particular value; assignment statement

A decorative network diagram in the top right corner, consisting of various sized circles (nodes) connected by thin lines (edges). Some nodes are solid grey, while others are hollow white with a grey border. The connections are a mix of solid and dashed lines.

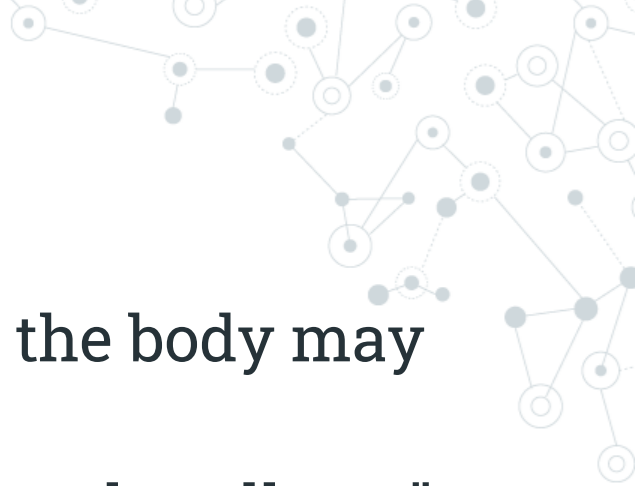
The `innerHTML` attribute allows us to change the html of any element in our document.

Before changing, the HTML value of the body wipes out all of the old tags and replaces it with whatever is in quotes.

A decorative network diagram in the bottom left corner, similar to the one in the top right, with nodes and connecting lines.


```
<html>
  <body>
    <h1> Hello World!!</h1>
    <button onclick = "changeThisPageFunc();"> Click Me!</button>
    <script type = "text/javascript">
      changeThisPageFunc = function() {
        document.body.h1.style.backgroundColor = "#8f2c01"
      }
    </script>
  </body>
</html>
```



A decorative network diagram in the top right corner, consisting of various sized circles (nodes) connected by thin lines (edges). Some nodes are solid grey, while others are hollow white with a grey outline. The connections are a mix of solid and dashed lines.

The reason this will not work is that the body may contain many h1 tags.

So we need a way to identify, and “get a handle on” the exact h1 that we want to change.

A decorative network diagram in the bottom left corner, similar to the one in the top right, with nodes of varying sizes and colors (solid grey and hollow white) connected by solid and dashed lines.

```
<html>
  <body>
    <h1> Hello World!!</h1>
    <button onclick = "changeThisPageFunc();"> Click Me!</button>
    <main>
      <h1>Hello Main</h1>
      <p>The five boxing wizards jumped quickly.</p>
    </main>
    <script type = "text/javascript">
      changeThisPageFunc = function() {
        var myMain;
        document.body.style.backgroundColor = "#8f2c01";
        myMain = document.querySelector('main');
        myMain.innerHTML = "<h3>Where have all the flowers gone?</h3>";
        myMain.style.height = "50px";
        myMain.style.width = "50%";
        myMain.style.backgroundColor = "#65419a";
      }
    </script>
  </body>
</html>
```

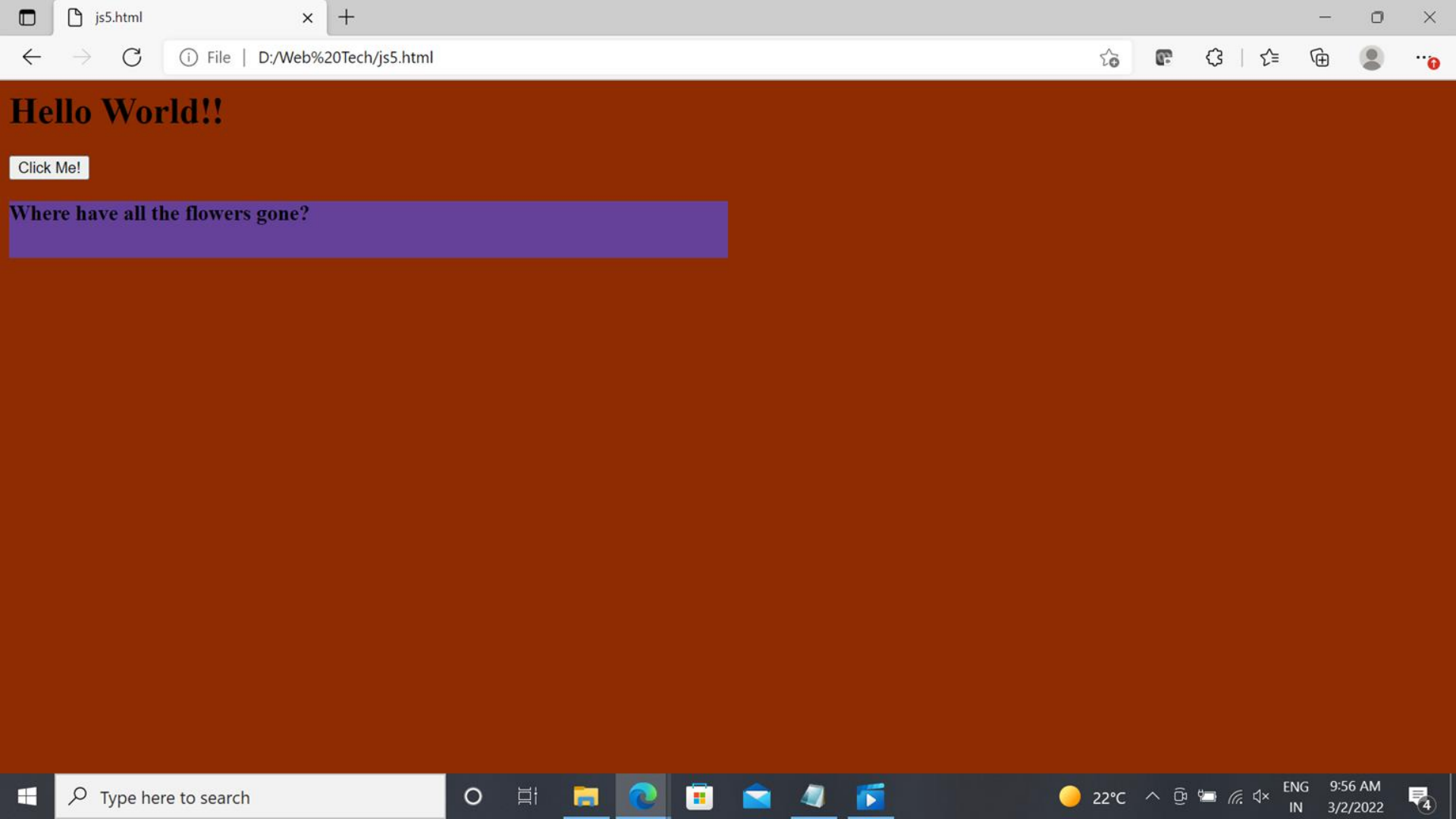


Hello World!!

Click Me!

Hello Main


The five boxing wizards jumped quickly.



Hello World!!

Click Me!

Where have all the flowers gone?



More assignment statements that only change the main tag.

myMain is a variable.

Variables are names that we give to Javascript objects, in this case it is the main tag.

Why not just use `document.body.main`?






Because it does not exist.

The document object has a body attribute because all documents have a body, but not all documents have a main.

So, we need to find the main in our document and give it a name so we can do things with it.

This is what is going on in the statement: `myMain = document.querySelector('main');`






The right hand side of the assignment statement
contains a method.

Tells an object to go do something.

A method is just name for a function that is
attached to an object, and so it is an abstraction.





Everything you learned about selectors in CSS can be used to tell this method what you want it to find!

'main' is a parameter to the `querySelector` method, and could contain any selector.

`document.head` and `document.body` are the only tags that can be accessed directly.

All other tags must be accessed using `querySelector`.



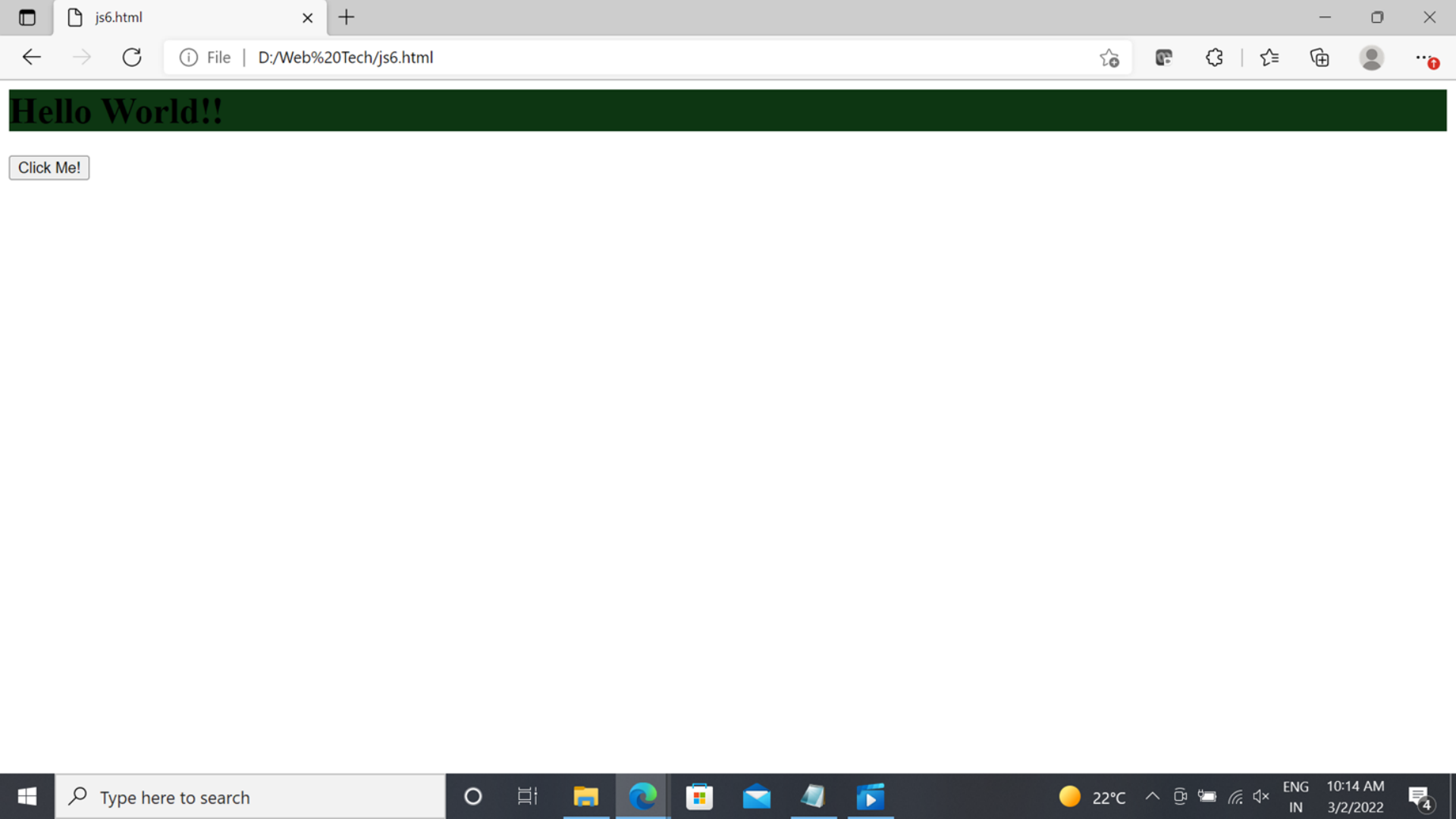
Modify the example so that you can change the color of the h1.

```
<html>
  <body>
    <h1> Hello World!!</h1>
    <button onclick = "changeThisPageFunc();" > Click Me!</button>
    <script type = "text/javascript">
      changeThisPageFunc = function() {
        var myMain;
        myMain = document.querySelector('h1')
        myMain.style.backgroundColor = "#0d3612";
      }
    </script>
  </body>
</html>
```



Hello World!!

Click Me!



Hello World!!

Click Me!



HTML Elements for Interaction

Button

```
js - Notepad
File Edit Format View Help
<html>
  <body>
    <button>Click Me</button>
  </body>
</html>
```

Ln 3, Col 34

100%

Windows (CRLF)

UTF-8

Type here to search

25°C

11:22 AM
3/6/2022

js.html x +

File | C:/Users/Gargi/OneDrive/Documents/js.html

about:blank CS541.pdf Best Image Process...

Click Me

Full-screen Snap

Type here to search

25°C Sunny 11:21 AM 3/6/2022

The Input Tag

Many of the elements used in constructing a user interface use the input tag.

All of the input elements have a common set of attributes, the most important is the type attribute as that is what determines how the input element will look.

- type
- name
- id
- value

Text Input

js - Notepad

File Edit Format View Help

```
<html>
  <body>
    <button>Click Me</button>
    <input type = "text" value = "change me" size = "40">
  </body>
</html>
```

Ln 4, Col 55

100%

Windows (CRLF)

UTF-8

Type here to search

25°C Sunny

11:28 AM
3/6/2022

js.html x +
File | C:/Users/Gargi/OneDrive/Documents/js.html
about:blank CS541.pdf Best Image Process...

Click Me

Type here to search
25°C 11:28 AM 3/6/2022

Password

```
js - Notepad
File Edit Format View Help
<html>
  <body>
    <button>Click Me</button>
    <input type = "text" value = "change me" size = "40">
    <input type = "password" value = "secret" size = "40">
  </body>
</html>
```

↑ value is pre-filled thing

Click Me change me

Secret is written

Checkbox

js - Notepad

File Edit Format View Help

```
<html>
```

```
  <body>
```

```
    <button>Click Me</button>
```

```
    <input type = "text" value = "change me" size = "40">
```

```
    <input type = "password" value = "secret" size = "40">
```

```
    <br />
```

```
    <input type = "checkbox" name = "group1" value = "australia">Australia<br />
```

```
    <input type = "checkbox" name = "group1" value = "mauritius">Mauritius<br />
```

```
    <input type = "checkbox" name = "group1" value = "congo">Congo<br />
```

```
  </body>
```

```
</html>
```

Ln 6, Col 8

100%

Windows (CRLF)

UTF-8

Type here to search

25°C Sunny

11:39 AM
3/6/2022

Click Me
 Australia
 Mauritius
 Congo

Radio

js - Notepad

File Edit Format View Help

```
<html>
```

```
<body>
```

```
<button>Click Me</button>
```

```
<input type = "text" value = "change me" size = "40">
```

```
<input type = "password" value = "secret" size = "40">
```

```
<br />
```

```
<input type = "checkbox" name = "group1" value = "australia">Australia<br />
```

```
<input type = "checkbox" name = "group1" value = "mauritius">Mauritius<br />
```

```
<input type = "checkbox" name = "group1" value = "congo">Congo<br />
```

```
<input type = "radio" name = "group1" value = "libya">Libya<br />
```

```
<input type = "radio" name = "group1" value = "austria">Austria<br />
```

```
<input type = "radio" name = "group1" value = "guyana">Guyana<br />
```

```
</body>
```

```
</html>
```

Ln 10, Col 3

100%

Windows (CRLF)

UTF-8



Type here to search



25°C Sunny

11:45 AM
3/6/2022

Click Me change me

- Australia
- Mauritius
- Congo
- Libya
- Austria
- Guyana

Color

js - Notepad

File Edit Format View Help

<html>

<body>

<button>Click Me</button>

<input type = "text" value = "change me" size = "40">

<input type = "password" value = "secret" size = "40">

<input type = "checkbox" name = "group1" value = "australia">Australia

<input type = "checkbox" name = "group1" value = "mauritius">Mauritius

<input type = "checkbox" name = "group1" value = "congo">Congo

<input type = "radio" name = "group1" value = "libya">Libya

<input type = "radio" name = "group1" value = "austria">Austria

<input type = "radio" name = "group1" value = "guyana">Guyana

<input type = "color">

</body>

</html>

Ln 13, Col 24

100%

Windows (CRLF)

UTF-8

Type here to search

25°C Sunny

11:48 AM
3/6/2022

Click Me change me

- Australia
- Mauritius
- Congo
- Libya
- Austria
- Guyana



Range

js - Notepad

File Edit Format View Help

```
<html>
```

```
<body>
```

```
<button>Click Me</button>
```

```
<input type = "text" value = "change me" size = "40">
```

```
<input type = "password" value = "secret" size = "40">
```

```
<br />
```

```
<input type = "checkbox" name = "group1" value = "australia">Australia<br />
```

```
<input type = "checkbox" name = "group1" value = "mauritius">Mauritius<br />
```

```
<input type = "checkbox" name = "group1" value = "congo">Congo<br />
```

```
<input type = "radio" name = "group1" value = "libya">Libya<br />
```

```
<input type = "radio" name = "group1" value = "austria">Austria<br />
```

```
<input type = "radio" name = "group1" value = "guyana">Guyana<br />
```

```
<input type = "color">
```

```
<input type = "range" min = 0 max = 255 value = 125|
```

```
</body>
```

```
</html>
```

Ln 14, Col 54

100%

Windows (CRLF)

UTF-8

Type here to search

25°C

11:52 AM
3/6/2022

Click Me change me

- Australia
- Mauritius
- Congo
- Libya
- Austria
- Guyana



Date Stuff

- month
- datetime-local
- week
- time



```
<html>
<body>
  <button>Click Me</button>
  <input type = "text" value = "change me" size = "40">
  <input type = "password" value = "secret" size = "40">
  <br />
  <input type = "checkbox" name = "group1" value = "australia">Australia<br />
  <input type = "checkbox" name = "group1" value = "mauritius">Mauritius<br />
  <input type = "checkbox" name = "group1" value = "congo">Congo<br />
  <input type = "radio" name = "group1" value = "libya">Libya<br />
  <input type = "radio" name = "group1" value = "austria">Austria<br />
  <input type = "radio" name = "group1" value = "guyana">Guyana<br />
  <input type = "color">
  <input type = "range" min = 0 max = 255 value = 125>
  <input type = "month">
  <input type = "datetime-local">
  <input type = "week">
  <input type = "time">
</body>
</html>
```



Click Me change me

- Australia
- Mauritius
- Congo
- Libya
- Austria
- Guyana

Slider control, date input (mm/dd/yyyy), week input (Week --, ----), and time input (---:-- --).

Drop Down Menus

js - Notepad

File Edit Format View Help

<html>

<body>

<button>Click Me</button>

<input type = "text" value = "change me" size = "40">

<input type = "password" value = "secret" size = "40">

<input type = "checkbox" name = "group1" value = "australia">Australia

<input type = "checkbox" name = "group1" value = "mauritius">Mauritius

<input type = "checkbox" name = "group1" value = "congo">Congo

<input type = "radio" name = "group1" value = "libya">Libya

<input type = "radio" name = "group1" value = "austria">Austria

<input type = "radio" name = "group1" value = "guyana">Guyana

<input type = "color">

<input type = "range" min = 0 max = 255 value = 125>

<input type = "month">

<input type = "datetime-local">

<input type = "week">

<input type = "time">

<select id = "priority">

<option>High</option>

<option>Medium</option>

<option>Low</option>

</select>

</body>

</html>

Ln 23, Col 11

100%

Windows (CRLF)

UTF-8

Type here to search

25°C

12:00 PM
3/6/2022

Click Me change me

- Australia
- Mauritius
- Congo
- Libya
- Austria
- Guyana

Slider control, date input (mm/dd/yyyy), week input (Week --, ----), time input (---:-- --), and a dropdown menu (High).



Javascript Events

```
<html>
  <head>
    <title>Colors</title>
  </head>
  <body>
    <label for = "redi">Red:</label>
    <input type = "text" id = "redi" min = "0" value = "255" /> <br>
    <label for = "greeni">Green:</label>
    <input type = "text" id = "greeni" min = "0" value = "255" /> <br>
    <label for = "bluei">Blue:</label>
    <input type = "text" id = "bluei" min = "0" value = "255" /> <br>
    <button onclick = "changeColor();">Change Color</button>
    <script>
      changeColor = function() {
        red = document.querySelector('#redi').value;
        green = document.querySelector('#greeni').value;
        blue = document.querySelector('#bluei').value;
        colorStr = "rgb("+red+","+green+","+blue+")";
        document.body.style.backgroundColor = colorStr;
      }
    </script>
  </body>
</html>
```



Colors x +

File | C:/Users/Gargi/OneDrive/Documents/js2.html

about:blank CS541.pdf Best Image Process...

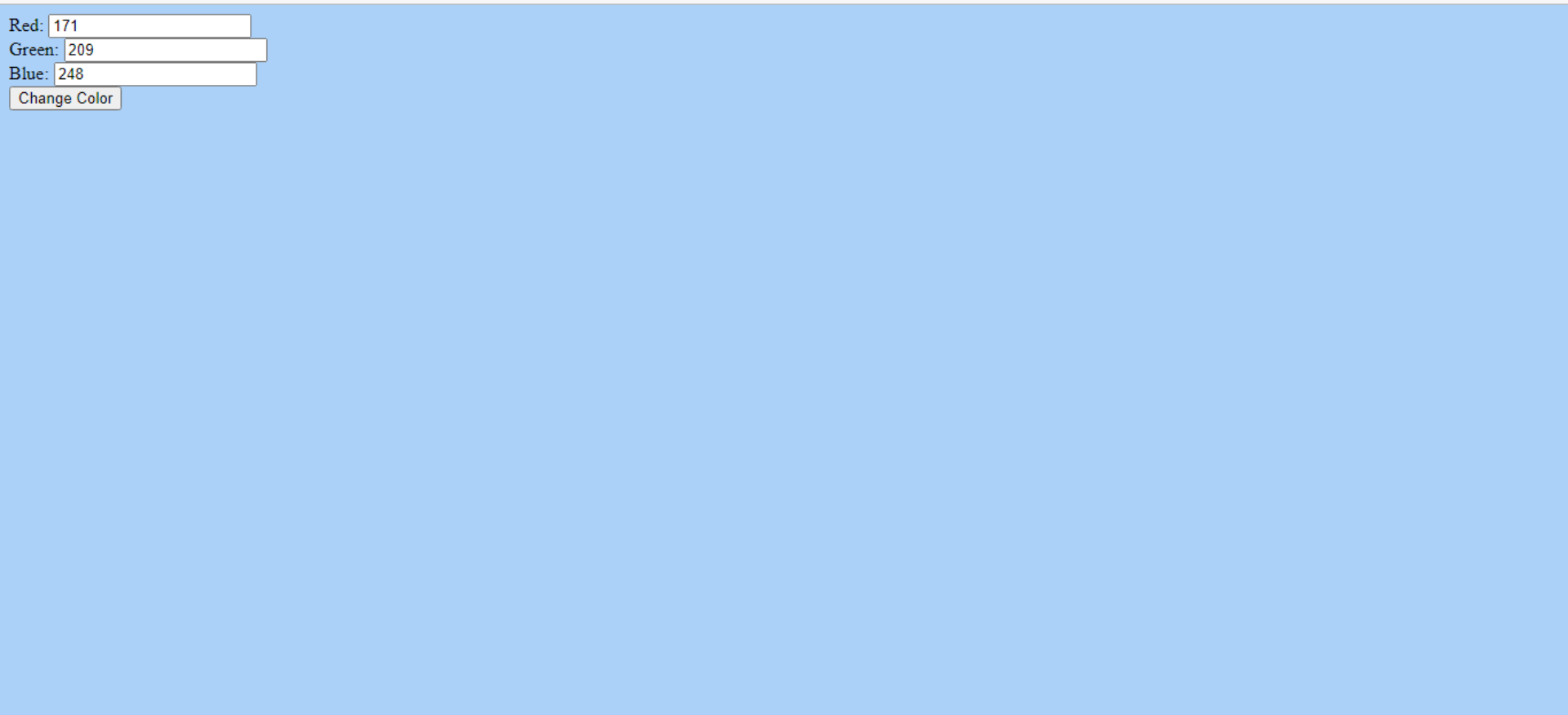
Red:

Green:


Blue:

Type here to search

25°C 12:12 PM 3/6/2022





Red:
Green:
Blue:



We are using the `querySelector` method to obtain a reference to the text input element and reading its value.


The `onclick` method on our button attaches the `changeColor` method to our button.





If we could just type a new value into one of the text input boxes and have that cause the color to change.

Another event type that is commonly used with input is the onchange event.



```
<html>
  <head>
    <title>Colors</title>
  </head>
  <body>
    <label for = "redi">Red:</label>
    <input type = "text" id = "redi" onchange = "changeColor()" value = "255" /> <br>
    <label for = "greeni">Green:</label>
    <input type = "text" id = "greeni" onchange = "changeColor()" value = "255" /> <br>
    <label for = "bluei">Blue:</label>
    <input type = "text" id = "bluei" onchange = "changeColor()" value = "255" /> <br>
    <button onclick = "changeColor();">Change Color </button>
    <script>
      changeColor = function() {
        red = document.querySelector('#redi').value;
        green = document.querySelector('#greeni').value;
        blue = document.querySelector('#bluei').value;
        colorStr = "rgb("+red+", "+green+", "+blue+)";
        document.body.style.backgroundColor = colorStr;
      }
    </script>
  </body>
</html>
```



Red:

Green:

Blue:

A decorative network diagram in the top right corner, consisting of various sized circles (nodes) connected by thin lines (edges). Some nodes are solid grey, some are hollow white, and some are dashed grey. The connections form a complex, branching structure.

Typing return or clicking with the mouse is still not as good as we could do.

There are a few other keyboard specific events we can attach to our text input box.

A decorative network diagram in the bottom left corner, similar to the one in the top right, with nodes of different sizes and styles connected by lines.

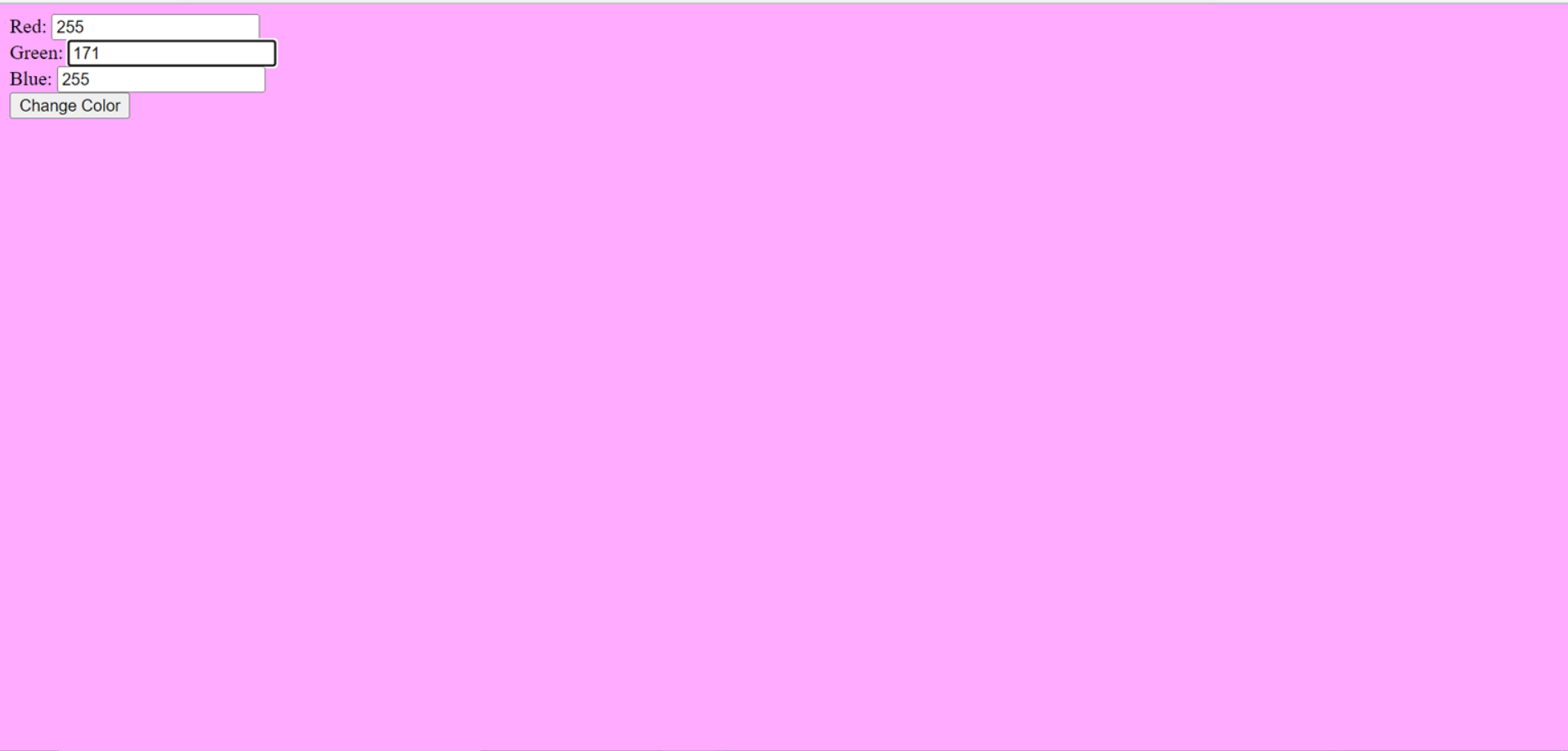
```
<html>
  <head>
    <title>Colors</title>
  </head>
  <body>
    <label for = "redi">Red:</label>
    <input type = "text" id = "redi" onkeyup = "changeColor()" value = "255" /> <br>
    <label for = "greeni">Green:</label>
    <input type = "text" id = "greeni" onkeyup = "changeColor()" value = "255" /> <br>
    <label for = "bluei">Blue:</label>
    <input type = "text" id = "bluei" onkeyup = "changeColor()" value = "255" /> <br>
    <button onclick = "changeColor();">Change Color </button>
    <script>
      changeColor = function() {
        red = document.querySelector('#redi').value;
        green = document.querySelector('#greeni').value;
        blue = document.querySelector('#bluei').value;
        colorStr = "rgb("+red+", "+green+", "+blue+)";
        document.body.style.backgroundColor = colorStr;
      }
    </script>
  </body>
</html>
```




Red:


Green:

Blue:






Onkeyup: This event occurs after the user has typed a key and the key has returned to its up position.

- onkeydown – fires when the key is pressed
 - onkeypress – fires after onkeydown but not for control, shift, alt
 - onkeyup – fires when the key goes up
- 



You may notice some slightly unreliable behavior here.

The problem is that there is a slight timing issue between when the key event happens and when the update to the text input boxes value happens.



```
<html>
  <head>
    <title>Colors</title>
  </head>
  <body>
    <label for = "redi">Red:</label>
    <input type = "range" min = 0 max = 255 id = "redi" onchange = "changeColor()" value = "255" /> <br>
    <label for = "greeni">Green:</label>
    <input type = "range" min = 0 max = 255 id = "greeni" onchange = "changeColor()" value = "255" /> <br>
    <label for = "bluei">Blue:</label>
    <input type = "range" min = 0 max = 255 id = "bluei" onchange = "changeColor()" value = "255" /> <br>
    <script>
      changeColor = function() {
        red = document.querySelector('#redi').value;
        green = document.querySelector('#greeni').value;
        blue = document.querySelector('#bluei').value;
        colorStr = "rgb("+red+", "+green+", "+blue+)";
        document.body.style.backgroundColor = colorStr;
      }
    </script>
  </body>
</html>
```



Red:

Green:

Blue:



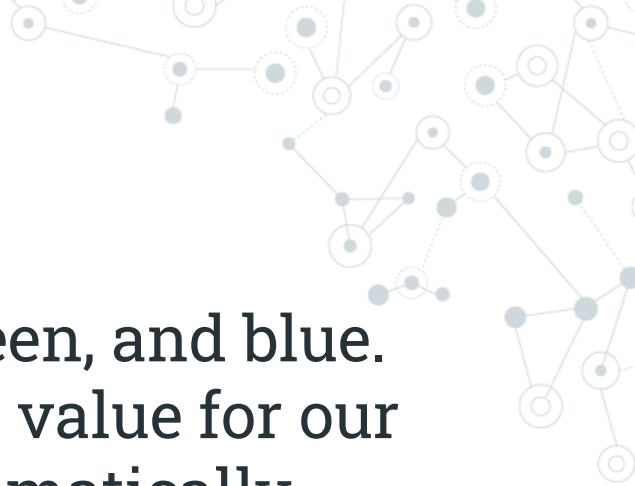

We can go one step further and have the color change as the bar moves!

Change the event from onchange to onmousemove.



```
<html>
  <head>
    <title>Colors</title>
  </head>
  <body>
    <label for = "redi">Red:</label>
    <input type = "range" min = 0 max = 255 id = "redi" onmousemove = "changeColor()" value = "255" /> <br>
    <label for = "greeni">Green:</label>
    <input type = "range" min = 0 max = 255 id = "greeni" onmousemove = "changeColor()" value = "255" /> <br>
    <label for = "bluei">Blue:</label>
    <input type = "range" min = 0 max = 255 id = "bluei" onmousemove = "changeColor()" value = "255" /> <br>
    <script>
      changeColor = function() {
        red = document.querySelector('#redi').value;
        green = document.querySelector('#greeni').value;
        blue = document.querySelector('#bluei').value;
        colorStr = "rgb("+red+", "+green+", "+blue+)";
        document.body.style.backgroundColor = colorStr;
      }
    </script>
  </body>
</html>
```



- 
- 
1. Let's display the values of red, green, and blue.
 2. Let's start with a different default value for our rgb colors and have the page automatically change its background color when the page is loaded.

```
<html>
  <head>
    <title>Colors</title>
  </head>
  <body>
    <label for="redi">Red:</label>
    <input type="range" min=0 max=255 id="redi" onmousemove="changeColor()" onmouseup="showValues()" value="125" /> <span id="redv"></span><br>
    <label for="greeni">Green:</label>
    <input type="range" min=0 max=255 id="greeni" onmousemove="changeColor()" onmouseup="showValues()" value="125" /><span id="greenv"></span><br>
    <label for="bluei">Blue:</label>
    <input type="range" min=0 max=255 id="bluei" onmousemove="changeColor()" onmouseup="showValues()" value="200" /><span id="bluev"></span>
    <br>
    <script>
      changeColor = function() {
        red = document.querySelector('#redi').value;
        green = document.querySelector('#greeni').value;
        blue = document.querySelector('#bluei').value;
        colorStr = "rgb(" + red + "," + green + "," + blue + ")";
        document.body.style.backgroundColor = colorStr;
      }
      showValues = function() {
        document.querySelector('#redv').innerHTML = document.querySelector("#redi").value;
        document.querySelector('#greenv').innerHTML = document.querySelector("#greeni").value;
        document.querySelector('#bluev').innerHTML = document.querySelector("#bluei").value;
      }
      window.onload = function() { changeColor(); showValues(); }
    </script>
  </body>
</html>
```



Red: 125
Green: 125
Blue: 200

Red: 125
Green: 125
Blue: 81

```
theSpan = document.querySelector('#redv');  
theSlider = document.querySelector("#redi");  
sliderVal = theSlider.value;  
theSpan.innerHTML = sliderVal;
```



```
<html>
<body>
  <button onclick = "stop();">Stop</button>
  <script>
    changeColor = function() {
      red = Math.floor(Math.random()*255);
      green = Math.floor(Math.random()*255);
      blue = Math.floor(Math.random()*255);
      colorStr = "rgb(" + red + "," + green + "," + blue + ")";
      document.body.style.backgroundColor = colorStr;

    }
    stop = function() {
      window.clearInterval(intId);
    }
    intId = window.setInterval(changeColor, 1000);
  </script>
</body>
</html>
```





Creating Elements on the Fly



Simple time application that records the date and time whenever you press the button.

```
<html>
  <body>
    <button onclick = "savetime()">Time</button>
    <table id = "timetable">
    </table>
    <script type = "text/javascript">
      savetime = function() {
        t = document.querySelector("#timetable")
        trow = document.createElement("tr")
        td = document.createElement("td")
        contents = document.createTextNode(Date())
        td.appendChild(contents)
        trow.appendChild(td)
        t.appendChild(trow)
      }
    </script>
  </body>
</html>
```





Thank You!