

# Internet Web Site Design

## HTML JavaScript

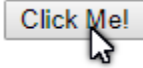
- JavaScript is the programming language of HTML and the Web.
- JavaScript is easy to learn.
- JavaScript makes HTML pages more dynamic and interactive.

### The HTML <script> Tag

- The <script> tag is used to define a client-side script (JavaScript).
- The <script> element either contains scripting statements, or it points to an external script file through the src attribute.
- Common uses for JavaScript are image manipulation, form validation, and dynamic changes of content.
- Old JavaScript examples may use a type attribute: <script type="text/javascript">. The type attribute is not required. JavaScript is the default scripting language in HTML.

### JavaScript Can Change HTML Content

- One of many JavaScript HTML methods is getElementById().
- This example uses the method to "find" an HTML element (with id="demo") and changes the element content (innerHTML) to "Hello JavaScript":

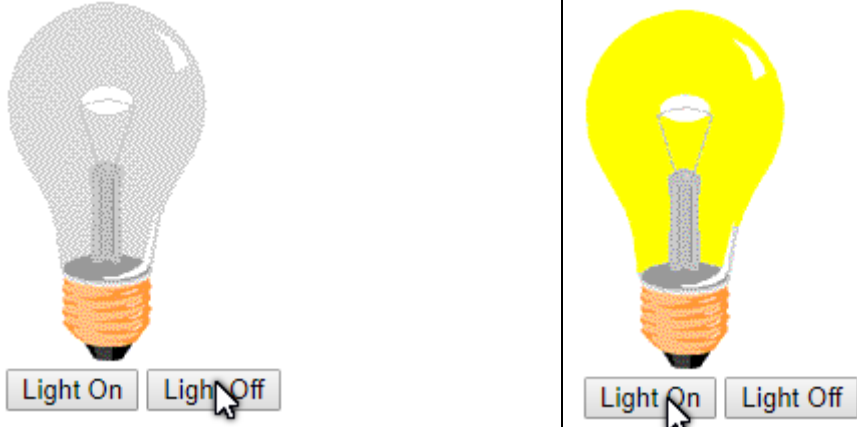
<b>HTML Script</b>	<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;   &lt;p id="demo"&gt;JavaScript can change HTML content.&lt;/p&gt;    &lt;button type="button" onclick='Click()'&gt;Click Me!&lt;/button&gt;    &lt;script&gt;   function Click(){   document.getElementById("demo").innerHTML = "Hello JavaScript!"   }   &lt;/script&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
<b>Result</b>	JavaScript can change HTML content. 	Hello JavaScript! 

**Note:** JavaScript accepts both double and single quotes.

# Internet Web Site Design

## JavaScript Can Change HTML Attribute Values

– In this example JavaScript changes the value of the src (source) attribute of an <img> tag:

<b>HTML Script</b>	<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt;   &lt;img id="myImage" src="pic_bulboff.gif" width="100" height="180"&gt;   &lt;br&gt;   &lt;button type="button" onclick="light(1)"&gt;Light On&lt;/button&gt;   &lt;button type="button" onclick="light(0)"&gt;Light Off&lt;/button&gt;   &lt;script&gt;   function light(sw) {     var pic;     if (sw == 0) {       pic = "pic_bulboff.gif"     } else {       pic = "pic_bulbon.gif"     }     document.getElementById('myImage').src = pic;   }   &lt;/script&gt; &lt;/body&gt; &lt;/html&gt;</pre>
<b>Result</b>	

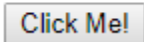

# Internet Web Site Design

## JavaScript Can Change HTML Styles (CSS)

- Changing the style of an HTML element, is a variant of changing an HTML attribute:

<b>HTML Script</b>	<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;p id="demo"&gt;Hello JavaScript&lt;/p&gt; &lt;button type="button" onclick="document.getElementById('demo').style.fontSize ='35px'"&gt;Click Me!&lt;/button&gt;&lt;/body&gt; &lt;/html&gt;</pre>	
<b>Result</b>	<p>Hello JavaScript</p> 	<p><b>Hello JavaScript</b></p> 

- Changing the style of an HTML element, is a variant of changing an HTML attribute:

<b>HTML Script</b>	<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;p id="demo"&gt;Hello JavaScript&lt;/p&gt; &lt;button type="button" onclick="document.getElementById('demo').style.display='none'"&gt; Click Me!&lt;/button&gt; &lt;/html&gt;</pre>	
<b>Result</b>	<p>Hello JavaScript</p> 	

- Showing hidden HTML elements can also be done by changing the display style:

<b>HTML Script</b>	<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;p id="demo" style="display:none"&gt;Hello JavaScript!&lt;/p&gt; &lt;button type="button" onclick="document.getElementById('demo').style.display='block'" &gt;Click Me!&lt;/button&gt; &lt;/html&gt;</pre>	
<b>Result</b>		<p><b>Hello JavaScript!</b></p> 



# Internet Web Site Design

## Where to Put JavaScript Code

- You can place any number of scripts in an HTML document.
- Scripts can be placed in the <body>, or in the <head> section of an HTML page, or in both.

## JavaScript in <head>

- In this example, a JavaScript function is placed in the <head> section of an HTML page.
- The function is invoked (called) when a button is clicked:

<b>HTML Script</b>	<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt; &lt;script&gt; function myFunction() {   document.getElementById("demo").innerHTML="Paragraph changed."; } &lt;/script&gt; &lt;/head&gt; &lt;body&gt; &lt;p id="demo"&gt;A Paragraph.&lt;/p&gt; &lt;button type="button" onclick="myFunction()"&gt;Try it&lt;/button&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
<b>Result</b>	<p>A Paragraph.</p> 	<p>Paragraph changed.</p> 

## JavaScript in <body>

- In this example, a JavaScript function is placed in the <body> section of an HTML page.
- The function is invoked (called) when a button is clicked:

```
<!DOCTYPE html>
<html>
<body>
<p id="demo">A Paragraph.</p>
<button type="button" onclick="myFunction()">Try it</button>
<script>
function myFunction() {
  document.getElementById("demo").innerHTML = "Paragraph changed.";
}
</script>
</body>
</html>
```

# Internet Web Site Design

## External JavaScript

- Scripts can also be placed in external files:

### External file: myScript.js

```
function myFunction() {  
    document.getElementById("demo").innerHTML = "Paragraph changed."  
}
```

- External scripts are practical when the same code is used in many different web pages.
- JavaScript files have the file extension .js.
- To use an external script, put the name of the script file in the src (source) attribute of a <script> tag:

```
<!DOCTYPE html>  
<html>  
<body>  
<p id="demo">A Paragraph.</p>  
<button type="button" onclick="myFunction()">Try it</button>  
<script src="myScript.js"></script>  
</body>  
</html>
```

## External JavaScript Advantages

- Placing scripts in external files has some advantages:
  - It separates HTML and code
  - It makes HTML and JavaScript easier to read and maintain
  - Cached JavaScript files can speed up page loads
- To add several script files to one page – use several script tags:

```
<script src="myScript1.js"></script>  
<script src="myScript2.js"></script>
```

## The HTML <noscript> Tag

- The <noscript> tag is used to provide an alternate content for users that have disabled scripts in their browser or have a browser that doesn't support client-side scripts:

```
<noscript>Sorry, your browser does not support JavaScript!</noscript>
```

# Internet Web Site Design

## JavaScript Output

### JavaScript Display Possibilities

- JavaScript can "display" data in different ways:
  - Writing into an HTML element, using innerHTML.
  - Writing into the HTML output using document.write().
  - Writing into an alert box, using window.alert().
  - Writing into the browser console, using console.log().

### Using innerHTML

- To access an HTML element, JavaScript can use the document.getElementById(id) method.
- The id attribute defines the HTML element. The innerHTML property defines the HTML content:

<b>HTML Script</b>	<pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt; &lt;p id="demo"&gt;A Paragraph.&lt;/p&gt; &lt;button type="button" onclick="myFunction()"&gt;Try it&lt;/button&gt; &lt;script&gt; function myFunction() {     document.getElementById("demo").innerHTML="Paragraph changed."; } &lt;/script&gt; &lt;/body&gt; &lt;/html&gt;                 </pre>	
<b>Result</b>	<p>A Paragraph.</p> <input type="button" value="Try it"/>	<p>Paragraph changed.</p> <input type="button" value="Try it"/>

### Using document.write()

- For testing purposes, it is convenient to use document.write():

HTML Script	Result
<pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt; &lt;h2&gt;My First Web Page&lt;/h2&gt; &lt;p&gt;My first paragraph.&lt;/p&gt; &lt;script&gt; document.write(5 + 6); &lt;/script&gt; &lt;/html&gt;                 </pre>	<p><b>My First Web Page</b></p> <p>My first paragraph.</p> <p>11</p>

# Internet Web Site Design

- Using document.write() after an HTML document is fully loaded, will delete all existing HTML:

<b>HTML Script</b>	<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt; &lt;h2&gt;My First Web Page&lt;/h2&gt; &lt;p&gt;My first paragraph.&lt;/p&gt; &lt;button type="button" onclick="document.write(5 + 6)"&gt; Try it &lt;/button&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
<b>Result</b>	<p><b>My First Web Page</b></p> <p>My first paragraph.</p> <input type="button" value="Try it"/>	11

**Note:** The document.write() method should only be used for testing.

## Using window.alert()

- You can use an alert box to display data:

<b>HTML Script</b>	<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt; &lt;script&gt; window.alert("Hello World"); &lt;/script&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
<b>Result</b>	<p>An embedded page on this page says</p> <p>Hello World</p> <input type="button" value="OK"/>	

# Internet Web Site Design

## Using console.log()

- For debugging purposes, you can use the console.log() method to display data.

<b>HTML Script</b>	<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;body&gt; &lt;script&gt; console.log(5 + 6);&lt;/script&gt; &lt;/body&gt; &lt;/html&gt;</pre>
<b>Result</b>	