

# CGS 3175: Internet Applications Fall 2007

## Introduction To JavaScript – Part 1

Instructor :     Dr. Mark Llewellyn  
                  markl@cs.ucf.edu  
                  HEC 236, 407-823-2790  
                  <http://www.cs.ucf.edu/courses/cgs3175/fall2007>

School of Electrical Engineering and Computer Science  
University of Central Florida



## Things to Try Yourself (From Advanced XHTML & CSS – Part 3)

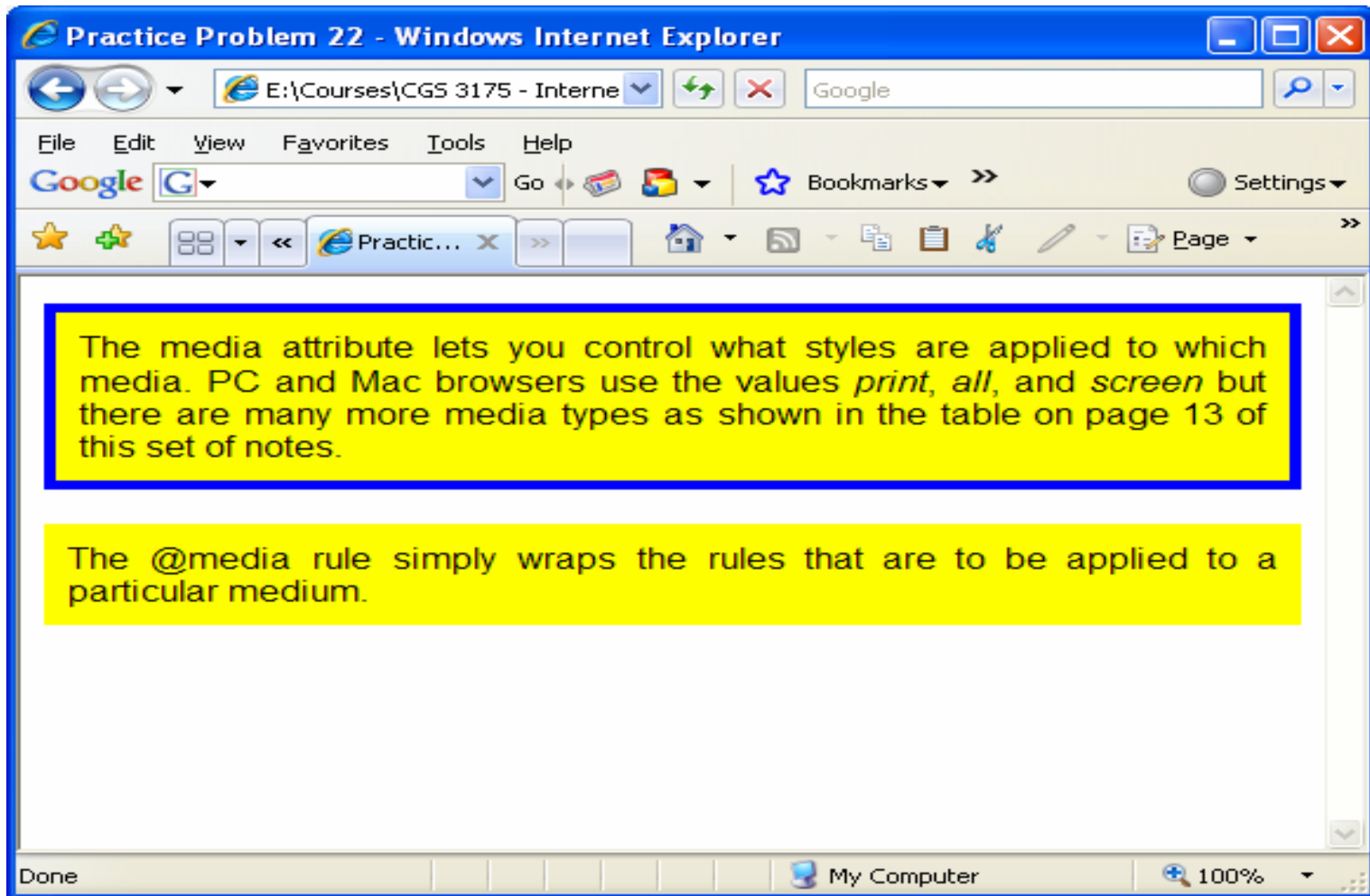
22. Create an external CSS style sheet for the XHTML document shown on the next page that will produce the browser version and print version which are shown on page 33.

- Note: you will need to slightly modify the XHTML document by (1) adding the link tag for the external style sheet and (2) modifying the paragraph tags for the style sheet identifiers you create.
- Note: The border on the screen version is 5px wide and on the print version it is 10pt wide. The font size for the paragraphs is set at 16pt.



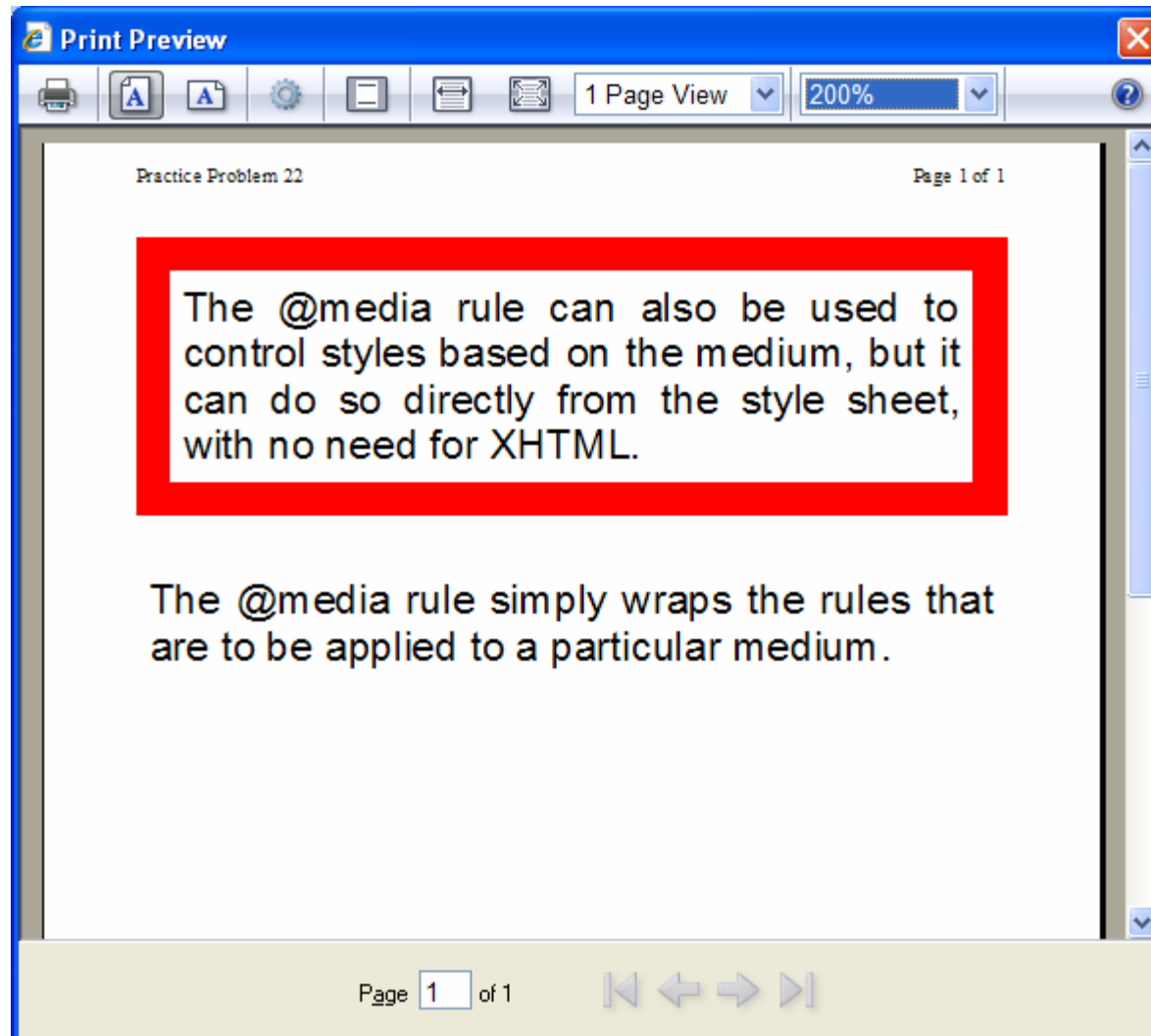
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Practice Problem 22 </title>
</head>
<body>
    <p>
        The media attribute lets you control what styles are applied to which
        media. PC and Mac browsers use the values <i>print</i>, <i>all</i>,
        and <i>screen</i>
        but there are many more media types as shown in the table on page 13
        of this set of notes.
    </p>
    <p>
        The @media rule can also be used to control styles based on the
        medium, but it can do so
        directly from the style sheet, with no need for XHTML.
    </p>
    <p>
        The @media rule simply wraps the rules that are to be applied to a
        particular medium.
    </p>
</body>
</html>
```





The browser version of the document





The print version of the document



# Modified XHTML For Practice Problem 22

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Practice Problem 22 </title>
<link rel="stylesheet" href="practice problem 22 CSS.css" media="all" type="text/css" />
</head>
<body>
  <p id="screen">
    The media attribute lets you control what styles are applied to which
    media. PC and Mac browsers use the values <i>print</i>, <i>all</i>, and
    <i>screen</i>
    but there are many more media types as shown in the table on page 13 of this set
    of notes.
  </p>
  <p id="print" class="before">
    The @media rule can also be used to control styles based on the medium, but it can
    do so
    directly from the style sheet, with no need for XHTML.
  </p>
  <p>
    The @media rule simply wraps the rules that are to be applied to a particular
    medium.
  </p>
</body>
</html>
```



# CSS For Practice Problem 22

```
/* CSS for Practice Problem 22 */
p {
    font: 16px sans-serif;
    background: yellow;
    padding: 10px;
    text-align: justify;
}
@media screen {
    p#print {
        display: none;
    }
    p#screen {
        border: 5px solid blue;
    }
}
@media print {
    p {
        padding: 0.05in;
    }
    p#print {
        border: 10pt solid red;
    }
    .before {
        page-break-before: always;
    }
    p#screen {
        display: none;
    }
}
```



# Introduction to JavaScript

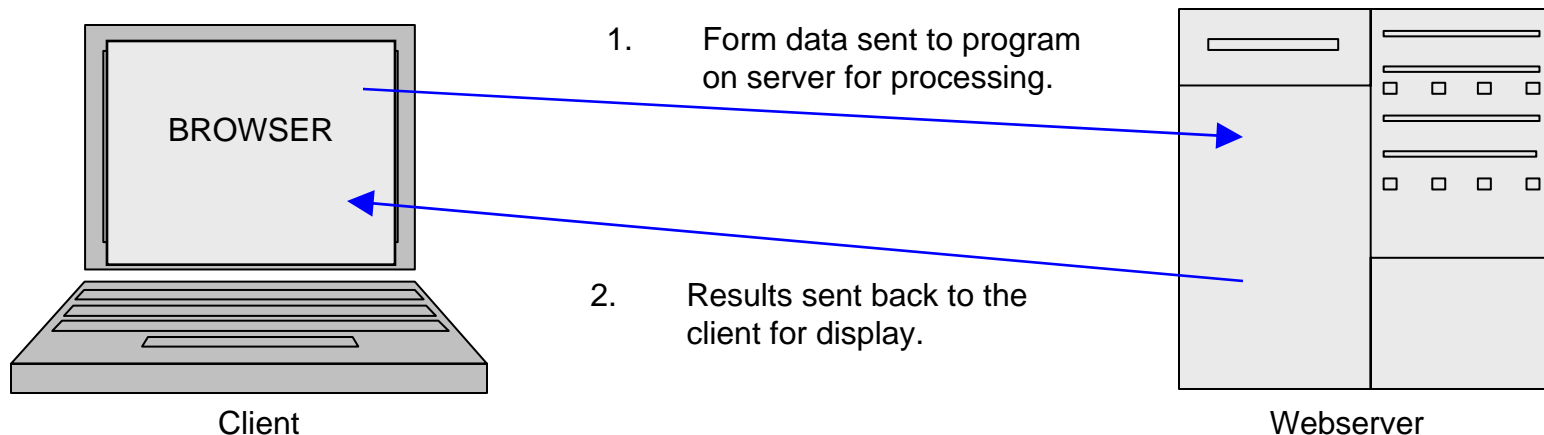
- Before we go any further let's get one thing very clear: JavaScript is **not** Java!
- JavaScript is a **scripting** language. (XHTML is a mark-up language.)
- Using JavaScript, you can incorporate techniques and effects that will make your Web pages come alive for the visitor allowing them a great deal of interaction with your site.
- We'll examine many aspects of JavaScript as we go along, but first some background material.





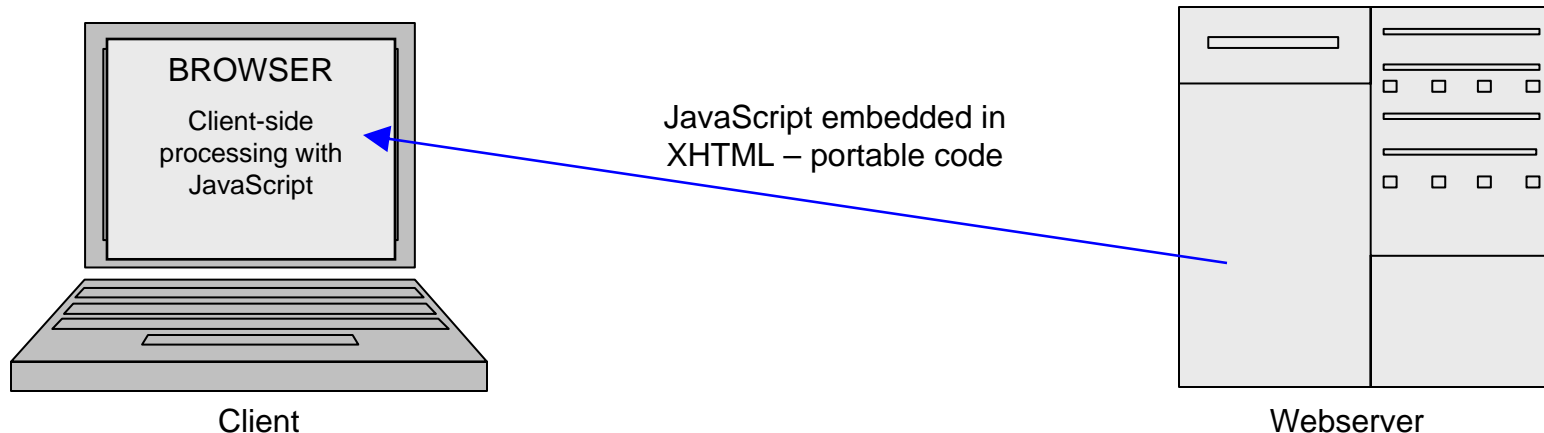
# Server-side and Client-side Programming

- Server-side scripts are programs that reside on a Web server and are executed on behalf of a client (in response to their requests typically from elements in Web pages).
  - For example, in assignment #4 you are utilizing a server-side script (written in PHP which is a server-side scripting language) to process the contents of your XHTML form.



# Server-side and Client-side Programming

- Client-side scripts are programs that reside on the client's machine (they are downloaded to your machine as part of the Web browser's cache memory) and are executed on behalf of the client's request.

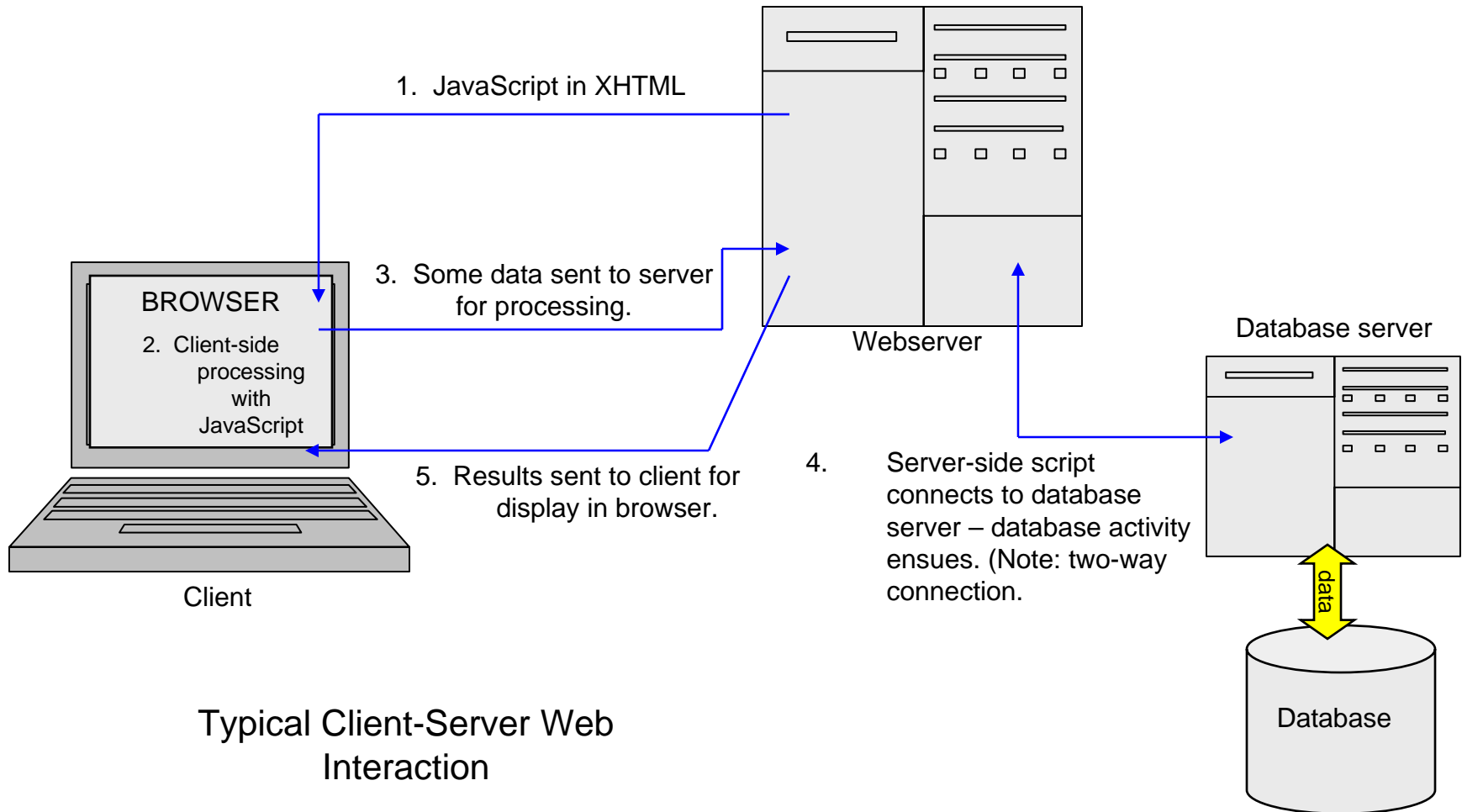


# Server-side and Client-side Programming

- While there are certain advantages to client-side scripting, client-side programs can never completely replace server-side scripts.
- Tasks such as running a search form or processing a purchase order or e-commerce situations must be run from a central server because only the server has connections to the databases needed to complete these types of operations.
- Indeed one of the primary reasons for using a server-side scripting language is to develop interactive Web sites that communicate with a database.
- With this in mind a more realistic picture of typical client-server interaction is shown on the next page where both client-side and server-side processing are occurring in parallel.



# Server-side and Client-side Programming



Typical Client-Server Web Interaction



## Server-side and Client-side Programming

- JavaScript is a client-side object-oriented scripting language that is interpreted by a Web browser.
- JavaScript is considered object-oriented because it is used to work with the objects associated with a Web page document: the browser window, the document itself, and the elements such as forms, images, and links contained within the page.



# How To Use JavaScript

- JavaScript is designed to work inside Web pages and within Web browsers. In this spirit, it extends the XHTML philosophy of using tags.
- The `<script>` tag is used to embed JavaScript in XHTML documents.
- The `<script>` tag can appear either in the HEAD or BODY sections of an XHTML document.
- Comments within the `<script>` tag, i.e., JavaScript comments begin with `/*` and end with `*/`. If the comment fits on a single line you can use `//` to start the comment with no ending delimiter.



# A JavaScript Enhanced Web Page – Version 1

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>A First JavaScript Enhanced Web Page </title>
```

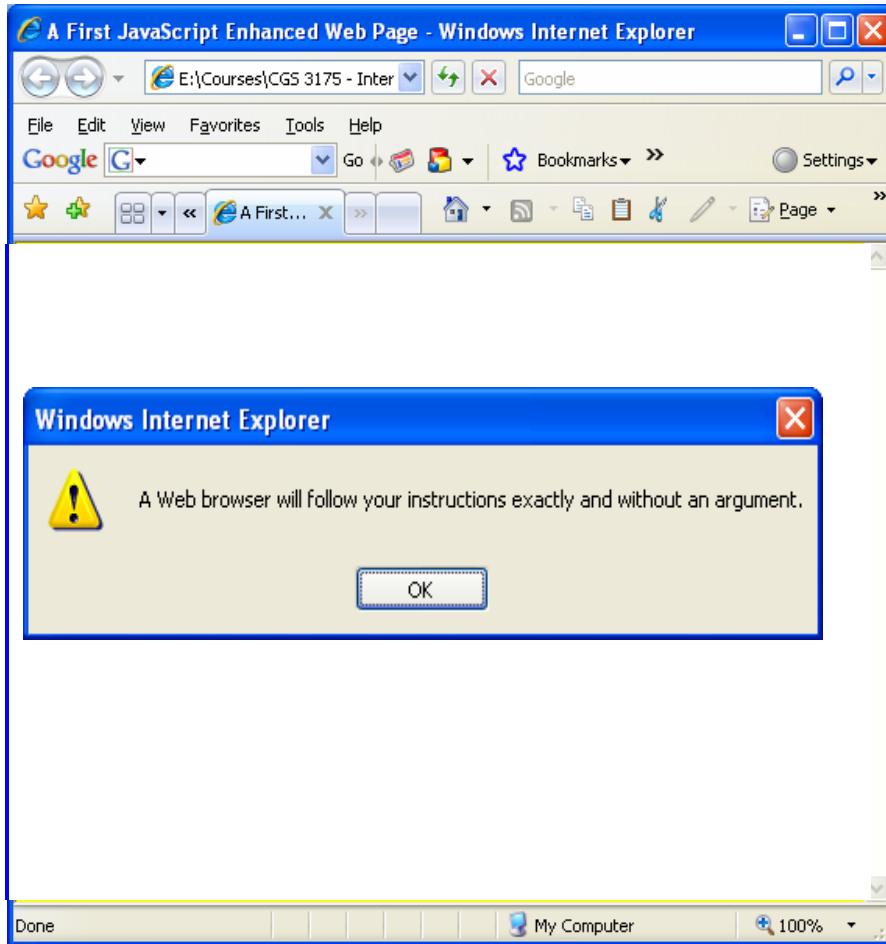
```
<script type="text/javascript">
    //this script simply pops up an alert window.
    window.alert("A web browser will follow your instructions exactly and
without an argument.");
</script>
```

```
<style type="text/css">
<!-- body {background-color:yellow;}
-->
</style>
</head>
<body>
<h1> Isn't it nice how computers do what they are told?
</h1>
</body>
</html>
```

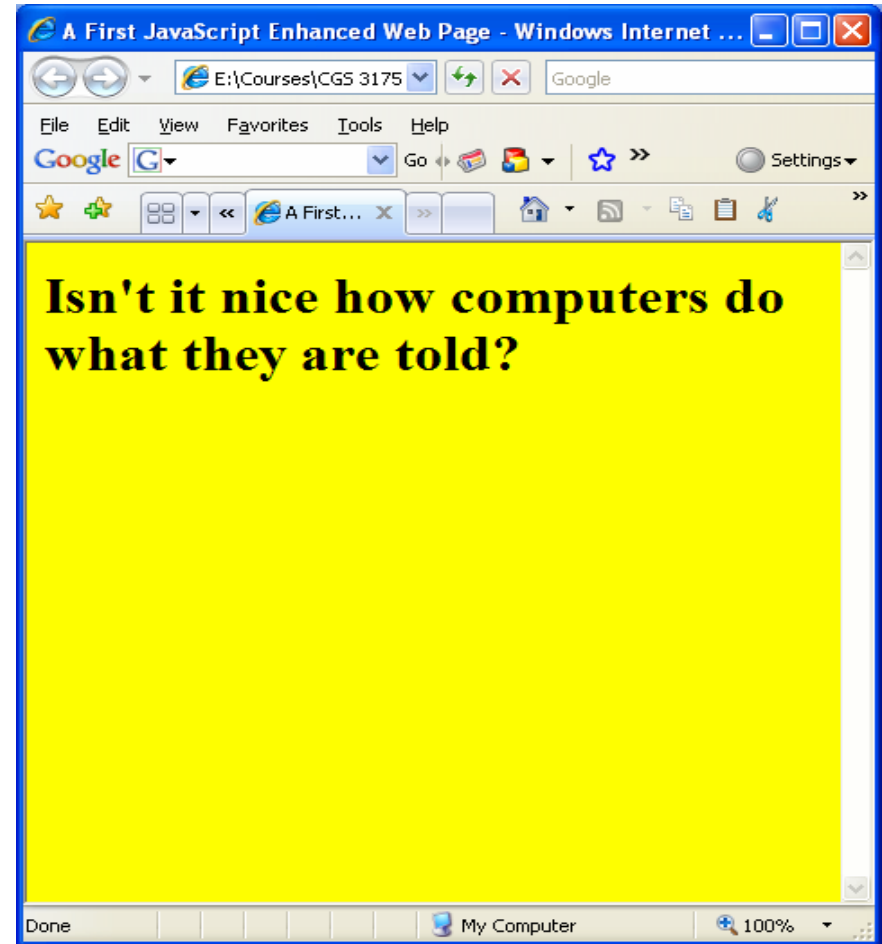
The JavaScript appearing in the HEAD of the XHTML document.



# A JavaScript Enhanced Web Page – Version 1



First screen



Second screen





# A JavaScript Enhanced Web Page – Version 2

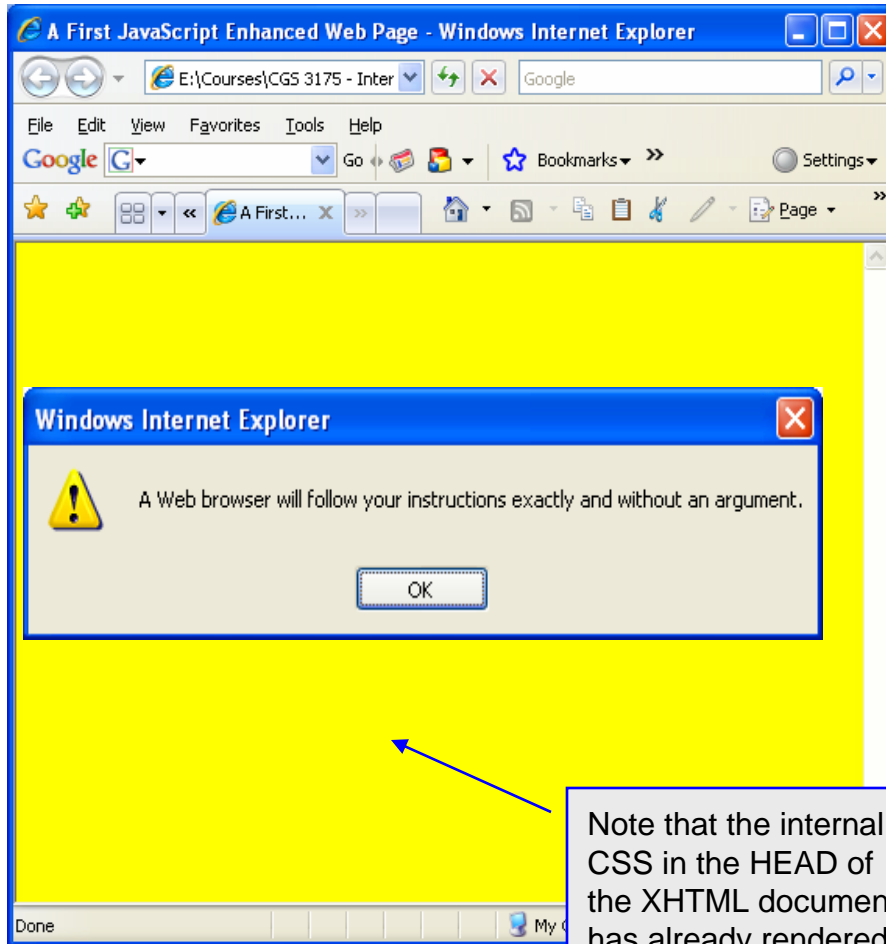
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title> A First JavaScript Enhanced Web Page </title>
<style type="text/css">
<!--  body {background-color:yellow;  }
-->
</style>
</head>
<body>
```

The JavaScript appearing in the BODY of the XHTML document.

```
<script type="text/javascript">
    //this script simply pops up an alert window.
    window.alert("A Web browser will follow your instructions exactly and
without an argument.");
</script>
<h1>Isn't it nice how computers do what they are told?</h1>
</body>
</html>
```

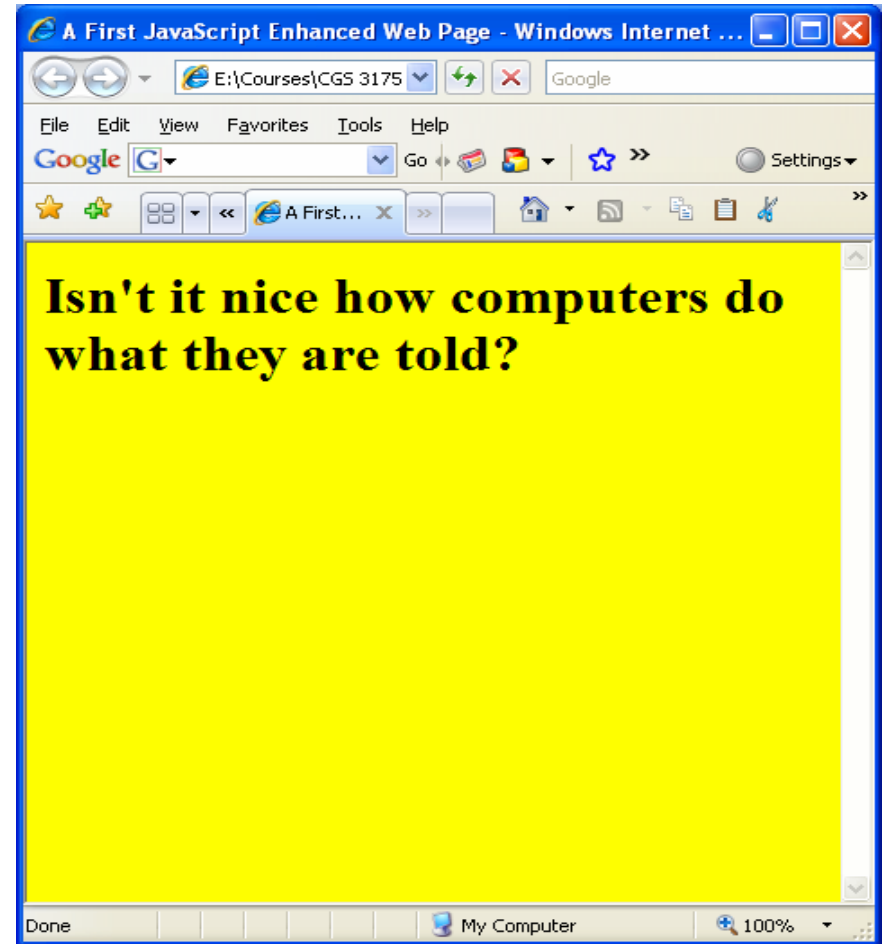


# A JavaScript Enhanced Web Page – Version 2



First screen

Note that the internal CSS in the HEAD of the XHTML document has already rendered the background color.



Second screen



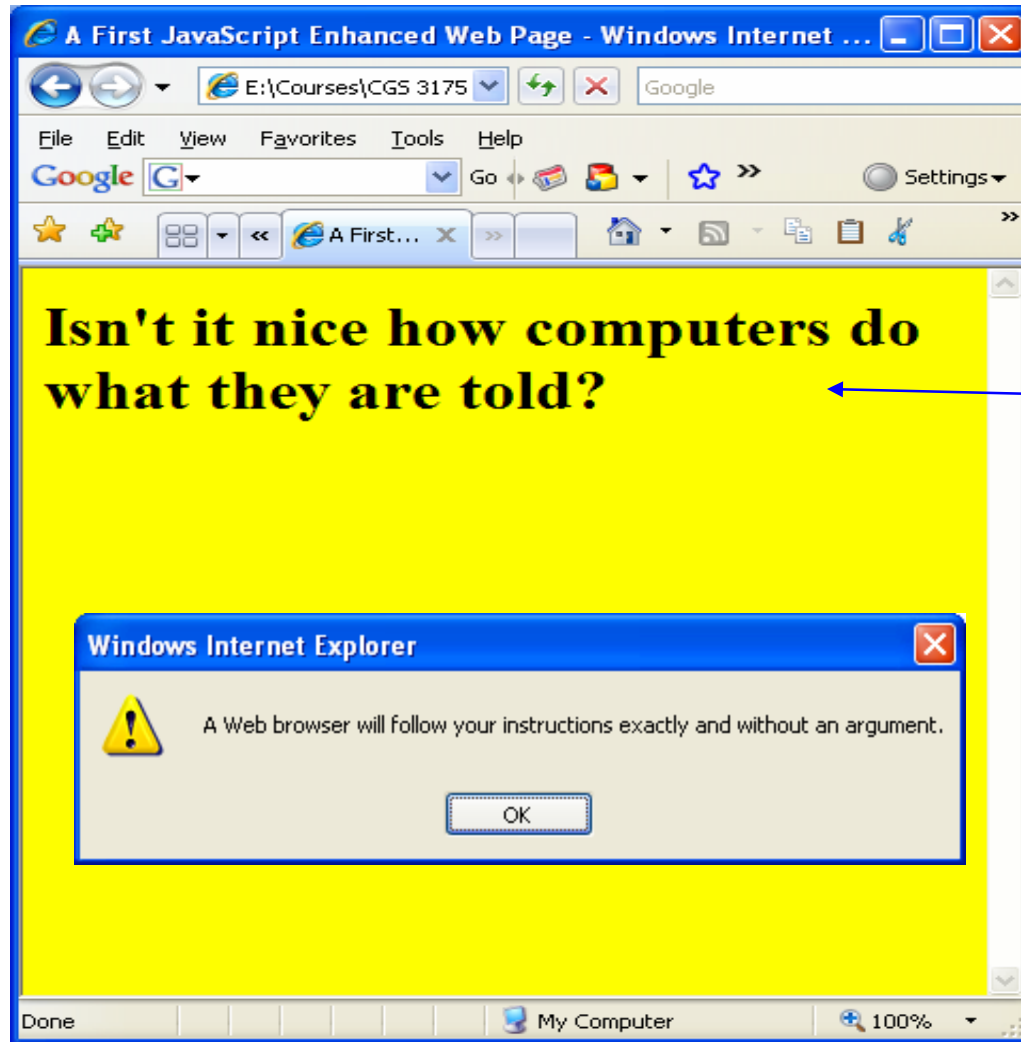
# A JavaScript Enhanced Web Page – Version 3

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title> A First JavaScript Enhanced Web Page </title>
<style type="text/css">
<!-- body {background-color:yellow; }
-->
</style>
</head>
<body>
<h1>Isn't it nice how computers do what they are told?</h1>
<script type="text/javascript">
  //this script simply pops up an alert window.
  window.alert("A Web browser will follow your instructions exactly and
without an argument.");
</script>
</body>
</html>
```

The JavaScript appearing in the BODY of the XHTML document, but after the line of text.



# A JavaScript Enhanced Web Page – Version 3



Note that the line of text appears at the same time as the alert window (actually it appears first, but is very rapidly followed by the alert window since it is rendered in normal flow).

Only screen



# How To Use JavaScript

- In keeping with the modern convention of separating content from presentation in Web pages, another very common technique for locating scripts is to place them in a file external to the XHTML document in which they will be activated.
- As we did with CSS, where one external style sheet could be used by several different XHTML documents to style their presentation, we can do the same with scripts.
- By creating a library (a file) of scripts in an external file and linking the XHTML pages to the library any XHTML document can utilize any of the scripts in the library.
- The next couple of pages we'll rework the same example, but this time use an external library for the script itself.



# A JavaScript Enhanced Web Page – Version 4

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title> A First JavaScript Enhanced Web Page </title>
<style type="text/css">
<!--  body {background-color:yellow;  }
-->
</style>
<script type="text/javascript" src="myscriptlibrary.js">
</script>
</head>
<body>
<h1>Isn't it nice how computers do what they are told?</h1>
</body>
</html>
```

The <script> tag appears in the HEAD section of the XHTML document and references a script library named "myscriptlibrary.js". (As with images, the "src" attribute is specified by a URL.)



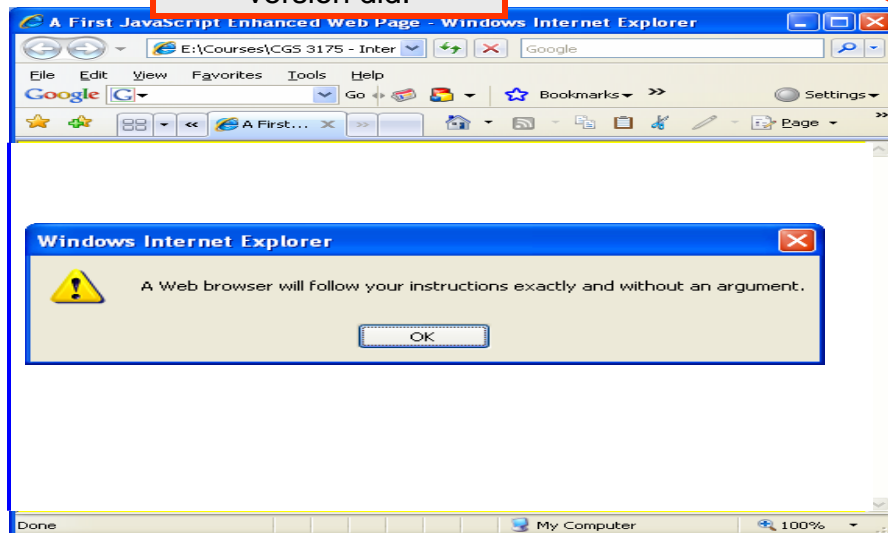
# A JavaScript Enhanced Web Page – Version 4

```
/* This is my JavaScript library of scripts */  
  
//this script is used in the first few JavaScript  
examples.  
  
window.alert("A Web browser will follow your  
instructions exactly and without an argument.");
```

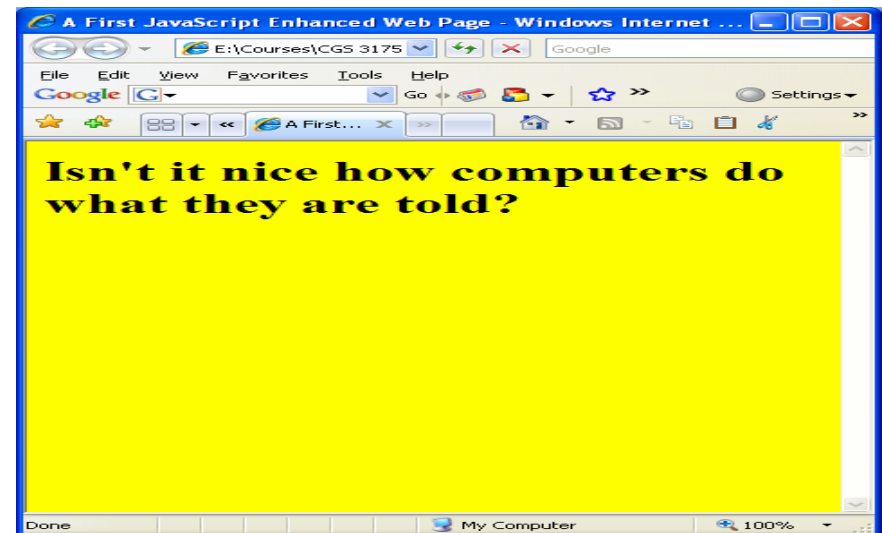
The actual script is unchanged in this library file.

The use of an external library for scripts will make more sense when we start writing JavaScript functions. For a simple case like the one shown here, there is little justification for an external script.

Version 4 behaves exactly as our first version did.



First screen



Second screen



## Things to Try Yourself

23. Create a 5<sup>th</sup> and 6<sup>th</sup> version of the XHTML document in this set of notes using the external script library. In version 5, put the `<script>` element before the `<h1>` element and in version 6, put the `<script>` element after the `<h1>` element. What effect does this have on the way the browser displays the page? Is the effect similar to any of the other versions we already created?
24. Using version 2 (page 17) of the XHTML document in this set of notes as a template. Modify the document using the JavaScript statement `document.writeln(" your text here ");` so that the new version of the document simply prints the message in the user's browser as shown on the next page.





# Things to Try Yourself

