CITS1231 Web Technologies

JavaScript: Image Rollover and Pop-up and Pull-Down Menus
Objectives

- Understand how to work with the JavaScript `document.images` collection
- Create image objects and image object arrays
- Set the properties of image objects
- Create image rollovers with image objects and the `document.images` collection
Objectives

- Understand how to create text rollovers
- Understand how to work with pop-up and pull-down menus
- Hide and unhide objects in a Web page
as the pointer moves over the image, the image changes to one containing a colored drop shadow
Working with Image Objects

- Referencing an Inline Image
  - Each inline image is part of an array of images in the document called the **image collection**
    
    ```javascript
    document.images[idref]
    document.images.idref
    ```
  - Other forms
    ```javascript
    document.getElementById("id")
    document.name
    ```
Working with Image Objects

- Referencing an Inline Image

```html
<ol>
  <li>document.images[0]</li>
</ol>
```
Working with Image Objects

• Creating an Image object
  – JavaScript treats all inline images as objects known as **image objects**
  – To create a new image object
    
    ```javascript
    image = new Image(width, height);
    ```
Working with Image Objects

- Properties of Image Objects

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>alt</td>
<td>Alternate text displayed by non-graphical browsers</td>
<td>document.images[5].alt = &quot;The World of Shakespeare&quot;;</td>
</tr>
<tr>
<td>border</td>
<td>The size of the image border in pixels</td>
<td>document.images[0].border = 0;</td>
</tr>
<tr>
<td>complete</td>
<td>A Boolean value indicating whether the image has been</td>
<td>if (document.images[0].complete) {</td>
</tr>
<tr>
<td></td>
<td>completely loaded by the browser</td>
<td>commands to run</td>
</tr>
<tr>
<td></td>
<td></td>
<td>}</td>
</tr>
<tr>
<td>height</td>
<td>The height of the image in pixels</td>
<td>document.images[0].height = 50;</td>
</tr>
<tr>
<td>hspace</td>
<td>The horizontal space around the image in pixels</td>
<td>document.images[0].hspace = 10;</td>
</tr>
<tr>
<td>id</td>
<td>The id of the image object</td>
<td>document.images[5].id = &quot;logo&quot;;</td>
</tr>
<tr>
<td>lowsrc</td>
<td>The URL of a low resolution version of the image</td>
<td>document.images[5].lowsrc = &quot;logo_small.jpg&quot;;</td>
</tr>
<tr>
<td>name</td>
<td>The name of the image object</td>
<td>document.images[5].name = &quot;logo&quot;;</td>
</tr>
<tr>
<td>src</td>
<td>The URL of the graphic image to be displayed in the</td>
<td>document.images[5].src = &quot;logo.jpg&quot;;</td>
</tr>
<tr>
<td></td>
<td>image object</td>
<td></td>
</tr>
<tr>
<td>vspace</td>
<td>The vertical space around the image in pixels</td>
<td>document.images[0].vspace = 10;</td>
</tr>
<tr>
<td>width</td>
<td>The width of the image in pixels</td>
<td>document.images[0].width = 100;</td>
</tr>
</tbody>
</table>
Working with Image Objects

- Detecting Image Objects
  - It is possible that your users will be running a very early browser version
  - Can use object detection to determine each user’s level of browser support
Creating an Image Rollover

- An **image rollover** is created when you change the source of an inline image from one graphic file to another.
Creating an Image Rollover

- Preloading the Images
  - Performance is an important consideration when creating a rollover effect
  - Can preload all of the image objects a user may need, storing the images in the browser’s memory
  - It is often more efficient to store your image objects in arrays
Creating an Image Rollover

- Preloading the Images
  - Example

```html
<link href="styles1.css" rel="stylesheet" type="text/css" />
<script type="text/javascript">
if (document.images) {
    var Imgover = new Array();
    Imgover[0] = new Image();
    Imgover[1] = new Image();
    Imgover[2] = new Image();
    Imgover[3] = new Image();
    Imgover[4] = new Image();

    var ImgOut = new Array();
    ImgOut[0] = new Image();
    ImgOut[1] = new Image();
    ImgOut[2] = new Image();
    ImgOut[3] = new Image();
    ImgOut[4] = new Image();
}
</script>
</head>
```
Creating an Image Rollover

• Swapping Image Objects
  – Once the images are preloaded, you can use JavaScript to swap the source for one image with the source for another

```javascript
function Rollover(i) {
  if (document.images) document.images[i].src = ImgOver[i].src;
}

function RollOut(i) {
  if (document.images) document.images[i].src = ImgOut[i].src;
}
</script>
</head>
```
Creating an Image Rollover

• Running the Image Rollover

```html
<div id="links">
  <a href="plays.htm">
    <img src="plays_out.jpg" alt="The Plays" onmouseover="Rollover(0)" onmouseout="RollOut(0)" />
  </a>

  <a href="sonnets.htm">
    <img src="son_out.jpg" alt="The Sonnets" onmouseover="Rollover(1)" onmouseout="RollOut(1)" />
  </a>

  <a href="bio.htm">
    <img src="bio_out.jpg" alt="Biography" onmouseover="Rollover(2)" onmouseout="RollOut(2)" />
  </a>

  <a href="globe.htm">
    <img src="globe_out.jpg" alt="The Globe" onmouseover="Rollover(3)" onmouseout="RollOut(3)" />
  </a>

  <a href="stratford.htm">
    <img src="strat_out.jpg" alt="Stratford" onmouseover="Rollover(4)" onmouseout="RollOut(4)" />
  </a>
</div>
```
Creating a Text Rollover

• Using the Hover Pseudo-Class
  
  a:hover {styles}
  
  – Example
  
  a:hover {color: red; font-weight:bold}

• General Text Rollovers
  
  – Modify the style properties of an element in response to the rollover event
Working with Menus

- contents of a menu (made visible when user clicks the menu or moves over it with the mouse)
- five menus

Place descriptive text about Shakespeare’s plays here
Working with Menus

• Creating a Pop-Up Menu
  – In a **pop-up menu**, a user clicks an object on the page and the menu appears, sometimes elsewhere on the page.
Working with Menus

- Creating a Pull-Down Menu
  - In a **pull-down menu**, part of the menu is visible
  - When a user clicks the visible part, the rest of the menu is revealed

![Diagram of a pull-down menu showing The Tragedies with sub-options such as Hamlet, King Lear, Macbeth, Othello, and Romeo and Juliet. The diagram illustrates the process where the user clicks the title to pull down the rest of the menu.]
Creating Pop-Up Menu Functions

```javascript
function hideIt(object) {
    object.style.visibility = "hidden";
}

function showIt(object) {
    object.style.visibility = "visible";
}
```
Creating Pop-Up Menu Functions

- Displaying Menu Contents

```javascript
<script type="text/javascript">
  var ActiveMenu = null;

  function hideActive() {
    if (ActiveMenu !== null) {
      hideIt(ActiveMenu);
      ActiveMenu = null;
    }
  }

  function popupMenu(M) {
    hideActive();
    ActiveMenu = document.getElementById(M);
    showIt(ActiveMenu);
  }
</script>
```
Creating Pop-Up Menu Functions

• Calling the Menu Functions

```html
<div id="Home" class="MenuTitle">
    <a href="ws.htm">Home Page</a>
</div>
<div id="Comedy" class="MenuTitle">
    <a href="JavaScript:popMenu('ComedyMenu')">The Comedies</a>
</div>
<div id="History" class="MenuTitle">
    <a href="JavaScript:popMenu('HistoryMenu')">The Histories</a>
</div>
<div id="Tragedy" class="MenuTitle">
    <a href="JavaScript:popMenu('TragedyMenu')">The Tragedies</a>
</div>
<div id="Romance" class="MenuTitle">
    <a href="JavaScript:popMenu('RomanceMenu')">The Romances</a>
</div>
```
Creating Pop-Up Menu Functions

• Calling the Menu Functions

```html
<div id="main" onclick="hideActive()"
  <p id="falstaff"><img src="falstaff.jpg" alt="" /></p>
  <p id="title"><img src="theplays.jpg" alt="The Plays of Shakespeare" /></p>
  <p id="firstp">
    William Shakespeare is credited with having written 37 plays, starting with <i>Henry VI, Part 1</i> in 1589, and continuing through <i>The Tempest</i> in 1611. He was the master of an incredible range of genres. From the classic tragedies of <i>Hamlet</i> and <i>King Lear</i> to the bawdy <i>The Merry Wives of Windsor</i>, Shakespeare wrote plays that evoked emotions of sorrow, pity, mirth, and wonder in his audience. To read or study these remarkable works online, select a play from the pull-down menu at the top of the page.
  </p>
</div>
```