JavaScript

ECMAScript (ECMA-262)

Why Study JavaScript?

JavaScript is one of the 3 languages all web developers must learn:

1. HTML to define the content of web pages

2. CSS to specify the layout of web pages

3. JavaScript to program the behavior of web pages

What is JavaScript?

- JavaScript is a scripting language， is an interpreted language
- JavaScript was designed to add interactivity to HTML pages
- A JavaScript consists of lines of executable computer code
- A JavaScript is usually embedded directly into HTML pages
- Everyone can use JavaScript without purchasing a license

Are Java and JavaScript the Same?

Are JS, JScript and JavaScript the Same?

What can a JavaScript Do?

- JavaScript is used in millions of Web pages to improve the design, put dynamic text into an HTML page, read and write HTML elements, react to events, validate forms, detect browsers, create cookies, and much more.
- JavaScript is the most popular scripting language on the internet, and works in all major browsers, such as Internet Explorer, Mozilla, Firefox, Netscape, Opera.

How to Put a JavaScript Into an HTML Page?

```
……
<body>
    <script type="text/javascript">
    document.write("Hello World!")
    </script>
</body>
</html>
```

The code above will produce this output on an HTML page:

```
Hello World!
```

How to Handle Older Browsers?

- Browsers that do not support JavaScript will display the script as page content. To prevent them from doing this, we may use the HTML comment tag:

```
<script type="text/javascript">
<!--
document.write("Hello World!")
//-->
</script>
```

Ending Statements With a Semicolon?

JavaScript Where To ...

- JavaScripts in the body section will be executed WHILE the page loads.
- JavaScripts in the head section will be executed when CALLED.
- Using an External JavaScript

```
<html>
  <head>
  <script src="xxx.js"></script>
  <script type="text/javascript">
   ....
  </script>
  </head>
  <body>
  <script type="text/javascript">
   ....
  </script>
  </body>
</html>
```

JavaScript Variables

- Rules for variable names:

  --Variable names are case sensitive

  --They must begin with a letter or the underscore character
- Declare a Variable

  [var] strname = *some value*
- Assign a Value to a Variable

  [var] strname = "Hege"
- Lifetime of Variables

JavaScript If...Else

- Very often when you write code, you want to perform different actions for different decisions. You can use conditional statements in your code to do this.

```
if (condition) {code }
```

```
If (condition) {code }
else {code }
```

```
if (condition1) {code }
else if (condition2) {code }
else {code }
```

e.g. JavaScript If...Else

Note that if is written in lowercase letters. Using uppercase letters (IF) will generate a JavaScript error!}

```
<script type="text/javascript">
   var d = new Date()
   var time = d.getHours()
   if (time>6 && time<10)
   { document.write("<b>Good morning</b>") }
   else if (time>10 && time<20)
   { document.write("<b>Good day</b>") }
   else { document.write("< b>Good night</b>") }
</script>
```

JS Switch

- You should use the switch statement if you want to select one of many blocks of code to be executed.

```
switch (n)
 { case 1: execute code block 1
           break
   case 2: execute code block 2
           break
   default: code to be executed if n is
           different from case 1 and 2
 }
```

```
<script type="text/javascript">
var d=new Date()
theDay=d.getDay()
switch (theDay)
{
case 5:
  document.write("Finally Friday")
  break
case 6:
  document.write("Super Saturday")
  break
default:
  document.write("I'm looking forward to this weekend!
}
</script>
```

JS Popup Boxes

- In JavaScript we can create three kinds of popup boxes: Alert box, Confirm box, and Prompt box.

```
alert("sometext")
```

```
confirm("sometext")
```

```
prompt("sometext","defaultvalue")
```

JS Functions

- A function is a reusable code-block that will be executed by an event, or when the function is called.
- The syntax for creating a function is:

```
function
functionname(var1,var2,...,varX)
{
some code
 }
```

Note: A function with no parameters must include the parentheses () after the function name:

JS Functions

- The return statement is used to specify the value that is returned from the function.

```
function prod(a,b)
{
x=a*b
return x
}
```

```
product=prod(2,3)
```

```
<html>
<head>
<script type="text/javascript">
function displaymessage()
{
alert("Hello World!")
}
</script>
</head>
<body>
<form>
<input type="button" value="Click me!"
onclick="displaymessage()" >
</form>
</body>
</html>
```

JS Loops

- Very often when you write code, you want the same block of code to run over and over again in a row. Instead of adding several almost equal lines in a script we can use loops to perform a task like this.
- The for Loop

```
for
(var=startvalue;var<=endvalue;var=var+increment)
{
    code to be executed
}
```

JS Loops

- The while loop

```
while (var<=endvalue)
{
    code to be executed
}
```

- The do...while Loop

```
do
{
    code to be executed
}
while (var<=endvalue)
```

JS break and continue Statements

- The break command will break the loop and continue executing the code that follows after the loop.
- The continue command will break the current loop and continue with the next value.

```
var i=0
for (i=0;i<=10;i++)
{
if (i==3){break}
document.write("The number is " + i)
document.write("<br />")
}
```

if (i==3) {continue} , the Result ?

JS Events

- By using JavaScript, we have the ability to create dynamic web pages. Events are actions that can be detected by JavaScript.
- onload and onUnload
- onFocus, onBlur and onChange
- onSubmit
- onMouseOver and onMouseOut

```
<form method="post" action="xxx.htm"
onsubmit="return checkForm()">
```

JS Special Characters

- The backslash (\) is used to insert apostrophes, new lines, quotes, and other special characters into a text string.

```
var txt="We are the so-called \"Vikings\"."
document.write(txt)
```

Code	Outputs
\'	single quote
\"	double quote
\&	ampersand
\\	backslash
\n	new line

JS Guidelines

- JavaScript is Case Sensitive
- JavaScript ignores extra spaces.
- break up a code line within a text string with a backslash.
- Comments

You can add comments to your script by using two slashes // or by using /* and */ (this creates a multi-line comment block)

JS Browser Detection

```
<html>
<head>
<script type="text/javascript">
function detectBrowser()
{var browser=navigator.appName
 var b_version=navigator.appVersion
 var version=parseFloat(b_version)
 if ((browser=="Netscape"||browser=="Microsoft Internet
Explorer") && (version>=8))
{alert("Your browser is good enough!")}
 else {alert("It's time to upgrade your browser!")}
}
</script>
 </head>
<body onload="detectBrowser()">
</body>
<html>
```

JS Cookies

- What is a Cookie?

  A cookie is a variable that is stored on the visitor's computer. Each time the same computer requests a page with a browser, it will send the cookie too. With JavaScript, you can both create and retrieve cookie values.

  (1) Name cookie

  (2) Password cookie

  (3) Date cookie

```
<html>
<head>
<script type="text/javascrip
function getCookie(c_name
{
if (document.cookie.length
  {
  c_start=document.cookie.
  if (c_start!=-1)
    {
    c_start=c_start + c_nam
    c_end=document.cookie
    if (c_end==-1) c_end=do
    return unescape(docume
    }
  }
return null
}
function setCookie(c_name
{
var exdate=new Date()
exdate.setDate(exdate.get
document.cookie=c_name
((expiredays==null) ? "" : ";
}
function checkCookie()
{
username=getCookie('user
if (username!=null)
  {alert('Welcome again '+u
else
  {
  username=prompt('Pleas
  if (username!=null && use
    {
    setCookie('username',us
    }
  }
}
</script>
</head>
<body onLoad="checkCoo
</body>
</html>
```

```
function getCookie(c_name)
{
```

```
function checkCookie()
{
username=getCookie('username')
if (username!=null)
  {alert('Welcome again '+username+'!')}
else
  {
  username=prompt('Please enter your name:',"")
  if (username!=null && username!="")
    {
    setCookie('username',username,365)
    }
  }
}
```

)

JS Form Validation

The function below checks if a required field has been left empty.

```
function validate_required(field,alerttxt)
{
with (field)
{
if (value==null||value=="")
  {alert(alerttxt);return false}
else {return true}
}
}
```

JS Form Validation

The function below checks if the content has the general syntax of an email.

```
function validate_email(field,alerttxt)
{
with (field)
{apos=value.indexOf("@")
 dotpos=value.lastIndexOf(".")
 if (apos<1||dotpos-apos<2)
  {alert(alerttxt);return false}
 else {return true}
}
}
```

JS Animation

```
<html>
<head>
<script type="text/javascript">
function mouseOver()
{document.b1.src ="images/b_blue.gif"}
function mouseOut()
{document.b1.src ="images/b_pink.gif"}
</script>
</head>
<body>
<a href="http://www.w3pop.com" target="_blank"
onmouseover="mouseOver()" onmouseout="mouseOut()">
<img border="0" alt="welcome" src="images/b_pink.gif"
name="b1" width="26" height="26" /></a>
</body>
</html>
```

JS Framework：jQuery

- jQuery is a JavaScript Library.
- jQuery greatly simplifies JavaScript programming.
- jQuery is easy to learn.
- There are lots of other JavaScript frameworks out there, but jQuery seems to be the most popular, and also the most extendable.
- The jQuery team knows all about cross-browser issues, and they have written this knowledge into the jQuery library. jQuery will run exactly the same in all major browsers, including Internet Explorer 6!

What is jQuery?

- jQuery is a lightweight, "write less, do more", JavaScript library.
- The purpose of jQuery is to make it much easier to use JavaScript on your website.
- jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code.
- jQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

How to use jQuery?

```
<head>
<script src="jquery-1.12.0.min.js"></script>
</head>
```

Google CDN (Content Delivery Network)

```
<head>
<script
src="https://ajax.googleapis.com/ajax/libs/jquer
y/1.12.0/jquery.min.js"></script>
</head>
```

jQuery Syntax

The jQuery syntax is tailor-made for **selecting** HTML elements and performing some **action** on the element(s).

```
$(selector).action()
```

- A $ sign to define/access jQuery
- A (*selector*) to "query (or find)" HTML elements
- A jQuery *action*() to be performed on the element(s)
- All selectors in jQuery start with the dollar sign and parentheses: $()

```
$(this).hide()

$(".test").hide()

$("#test").hide()
```

Example

jQuery is tailor-made to respond to events in an HTML page. When a user clicks on a button, the element with id="test" will be hidden:

```
$(document).ready(function(){
    $("button").click(function(){
        $(".test").hide();
    });
});
```

Homework 3

- 1.what is logo?
- 2.what is CSS?

- Translate
- A Web page consists of HTML code for document structure and CSS code for document presentation. CSS brings unprecedented power of style definition to the web developer. Style sheets can be attached to multiple Web pages, and a page can easily switch style by using different style sheets. The arrangement gives great flexibility to page styling and site maintenance.

The End