UNIVERSITY OF THE WEST of SCOTLAND
UWS

INTRODUCTION TO WEB DEVELOPMENT

Code: COMP07009

Week 10
HTML/JavaScript continued .....

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Rollover Links

A link has different states

- a:link
  Defines the style for normal unvisited link
- a:visited
  Defines the style for visited links
- a:active
  Defines the style for active links
- a:hover
  Defines the style for the hovered link

Notes

- hover selector is used to select elements when you mouse over them
- hover selector can be used on all elements, not only on links
- hover MUST come after :link and :visited (if they are present) in the CSS definition, in order to be effective!

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Rollover Links

- Defined within <style> elements

```
<style>
a:link {color:#FF0000;text-decoration:none;}
a:visited {background-color:red;}
a:active {text-decoration: none}
a:hover {text-decoration:underline;color:green;}
</style>
```

- Eg Link for w3schools website

  W3 Schools
  Link will be displayed based on style definition for its status

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Rollover Events

- There are various events which can be handled when the mouse pointer is moved
- onmouseover - pointer is moved onto an element
- onmousemove – whilst pointer is moving over an element
- onmouseout - pointer is moved out of an element
- onmousedown – button is pressed over an element
- onmouseup – button is released over an element

- The element can be anything from an image to a paragraph

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Rollover Images

- Syntax in HTML

```
<element onmouseover="SomeJavaScriptCode">
```

```
<img src="images/virus.jpg" alt="virus"
onmouseover="this.src='images/worm.jpg'"
onmouseout="this.src='images/virus.jpg'" />
```

- Eg The virus image is displayed within the HTML document
- The image changes to the worm when the pointer is on the image and changes back when the pointer is moved back out

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Loops

- Loops can execute a block of code a number of times, each time with a different value
- Different types
  - For
  - For in
  - While
  - Do while
- For Loop syntax

```
for (statement 1; statement 2; statement 3){
code to be executed
}
```

UNIVERSITY OF THE WEST of SCOTLAND
UWS

For Loop

- statement 1 sets a variable before the loop starts
- Eg var i=0;
- statement 2 defines the condition for the loop to run
- Eg i must be less than or equal to 5
- statement 3 increases a value each time the code block in the loop has been executed
- Eg i++ (same as i=i+1) increment by 1

```
for (i=0; i<=5; i++){
document.write("The number is " + i + "<br />");
}
```

```
The number is 0
The number is 1
The number is 2
The number is 3
The number is 4
The number is 5
```

UNIVERSITY OF THE WEST of SCOTLAND
UWS

While

- Loops through a block of code while a specified condition is true
- Syntax

```
while (condition){
code to be executed
}
```

```
x="",i=0;
while (i<6){
        x=x+"The number is "+i+"<br />";
        i++;
}
document.write(x);
```

Output

```
The number is 0
The number is 1
The number is 2
The number is 3
The number is 4
The number is 5
```

UNIVERSITY OF THE WEST of SCOTLAND
UWS

While

- Using the document.getElementById to output

```
<p>Click the button to loop through a block as long as i is less than 6.</p>
<button onclick="myFunction()">While demo</button>
<p id="demo"></p>
<script>
function myFunction(){
var x="",i=0;
while (i<5)
  {
  x=x + "The number is " + i + "<br>";
  i++;
  }
document.getElementById("demo").innerHTML=x;
}
</script>
```

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Do/While

- A variant of the while loop
- This loop will execute the code once, before checking if the condition is true
- Then it will repeat the loop as long as the condition is true
- Syntax

```
do{
{
code to be executed
}
while (condition);
```

Eg

```
x="",i=6;
do {
        x=x+"The number is "+i+"<br />";
        i++;
}
while (i<6);
document.write(x);
```

- i is assigned 6 which is outwith the condition but the code is still executed once

Output

The number is 6

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Break statement

- The break statement (used earlier to come out of a switch in the case statement) can also be used to break out of a loop
- Eg

```
for (i=0; i<=5; i++){
            if(i==3){
            break;
            }
document.write("The number is " + i + "<br />");
}
```

- Output

```
The number is 0
The number is 1
The number is 2
```

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Continue statement

- Breaks one iteration in the loop if a condition occurs
- Continues with the next iteration in the loop
- Eg

```
for (i=0; i<6; i++){
        if(i==3){
        continue;
        }
document.write("The number is " + i + "<br />");
}
```

- Output

```
The number is 0
The number is 1
The number is 2
The number is 4
The number is 5
```

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Errors

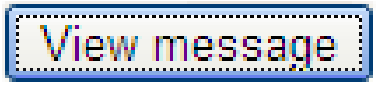
- Errors will occur
- Syntax errors, typos, incorrect input etc
- When an error occurs in JS, normally an error message will be generated
- Technical term is that JS will throw an error
- Try statement lets you test some code for errors
- Allows you to define code to be tested for errors while it is being executed
- Catch statement lets you handle the error
- Allows you to define code to executed, if an error occurs in the try block of code
- Throw statement lets you create custom errors
- Try and Catch come in pairs

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Try and Catch

- Syntax

```
try {
        some code
}
catch(err){
        handle errors
}
```



TypeError 'y' is undefined

Eg

```
<script>
function error()
{
try
  {
        var x=90;
        var z=x/y;
  }
catch(err)
  {
document.write(err.name+"
"+err.message);
  }
}
</script>
<input type="button" value="View
message" onclick="error()" />
```

UNIVERSITY OF THE WEST of SCOTLAND
UWS

JS Objects

- JS has built-in objects, like String, Date, Array etc
- An object is a special kind of data, with properties (attributes) and methods (behaviours)
- Property syntax

```
objectName.propertyName;
```

Eg

```
var message="Hello World!";
var x=message.length;

output -> 12
```

- Method syntax

```
objectName.methodName();
```

Eg

```
var message="Hello World!";
var x=message.toUpperCase();
document.write(x);

output -> HELLO WORLD!
```

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Creating an object

- Create a new instance of an Object and add properties to it

```
person=new Object();
person.firstname="Joe";
person.surname="Bloggs";
person.age=40;
person.haircolour="brown";
```

- Alternative way:

```
person={firstname:"Joe",surname:"Bloggs",age:40,haircolour:"brown"};
```

- Using an object constructor eg using a function to create an object

```
function person(firstname,surname,age,haircolour){
this.firstname=firstname;
this.surname=surname;
this.age=age;
this.haircolour=haircolour;
}
```

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Creating Object Instances

- Once an object has been created, you can create new instances of it
- Eg

```
myFather=new person("Joe","Bloggs",40,"brown");
myMother=new person("Annie","Smith",38,"blue");

document.write("My mother is called "+myMother.firstname+"
"+myMother.surname);
```

- Output

```
My mother is called Annie Smith
```

UNIVERSITY OF THE WEST of SCOTLAND
UWS

JS Arrays

- An array is a special variable
- Can hold more than one value at the same time
- Useful if you have a list of items (eg list of cars)
- Each element in an array is identified by its index number
- Index numbers are inside square brackets []
- The first element in an array has an index number of [0], the second [1], third [2] etc

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Create an array

- Over several statements

```
var myCars=new Array(6);
myCars[0]="BMW";
myCars[1]="Volvo";
myCars[2]="Saab";
```

```
var myCars=new Array();
myCars[0]="BMW";
myCars[1]="Volvo";
myCars[2]="Saab";
```

- If a number is used in array creation, then any unassigned elements are undefined eg myCars[3] to myCars[5]
- Over one statement
- Condensed, more efficient, no undefined elements

```
var myCars=new Array("BMW","Volvo","Saab");
```

UNIVERSITY OF THE WEST of SCOTLAND
UWS

JS Arrays

- All JS variables are objects
- Array elements are objects
- You can have functions in an array
- You can have arrays within arrays
- Array object has predefined properties and methods
- Eg length property and sort() method

```
document.write(“I have  “+ myCars.length +” cars”);
document.write("Cars in ascending order: " +myCars.sort());
```

- Output

```
I have 3 cars
```

```
Cars in ascending order: BMW, Saab, Volvo
```

- Other methods reverse()

UNIVERSITY OF THE WEST of SCOTLAND
UWS

For in loop

- For/in loop is a simplified loop used in arrays
- Syntax

```
for (var i in arrayName){
        code
};
```

- Eg

```
var myCars=new Array("BMW","Volvo","Saab");
for (var i in myCars){
        document.write("The car is " +myCars[i]+"<br />");
}
```

- Output

```
The car is BMW
The car is Volvo
The car is Saab
```

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Date Object

- The Date object is used to work with dates and times.
- Date() method gets today's date
- getFullYear() gets the year

```
var d = new Date();
document.write("Current year is "+d.getYear());
```

- getTime() returns the number of milliseconds since 01.01.1970
- getDay() gets a number representing the weekday

```
var d = new Date();
document.write("Current day is "+d.getDay());
```

- Starts at 0 which represents “Sunday”
- Best to use an array to write a weekday, and not just a number

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Date Object

- Best to use an array to write a weekday, and not just a number
- Eg when button is clicked, the function is called to create an array which assigns the names of the weekday to each of the elements in the array
  The current weekday will then be the text within the <p> tag with an id=“demo”

```
<p id="demo">.</p>
<button onclick="myFunction()">Try it</button>
<script>
function myFunction(){
var d = new Date();
var weekday=new
Array("Sun“,"Mon”,”Tues”,"Wed“,”Thur“,”Fri“,"Sat“);
var x = document.getElementById("demo");
x.innerHTML=weekday[d.getDay()];
}
</script>
```

UNIVERSITY OF THE WEST of SCOTLAND
UWS

Lab

- Make sure that you have completed the JS exercises from last week
- More Javascript exercises have been added to Moodle for this week
  - Rollover effects
  - For Loop
  - Function
  - While Loop
  - Arrays
  - Date
- Try some JS demos and exercises from w3schools.com
- You should continue to work on the second assessment and consider how you may use Javascript to make it more interactive
- Continue to validate your HTML and CSS pages