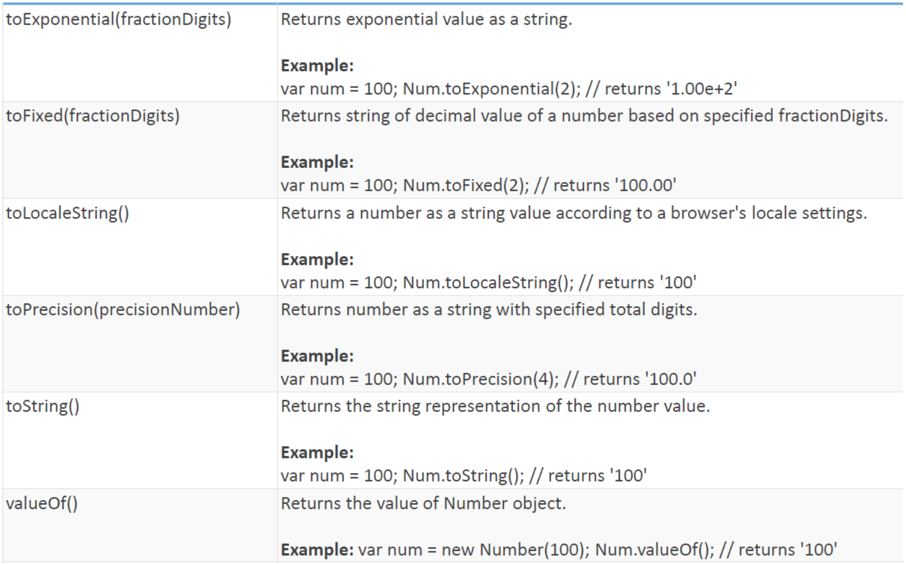
# MOD3 – JAVASCRIPT, AJAX, JQUERY

**MAX_VALUE Specifies the largest value a number can have.**
**MIN_VALUE Specifies the smallest value a number can have without being equal to 0.**
**NaN Stands for Not a Number. Represents a value that is not equal to any numeric value.**
**NEGATIVE_INFINITY A special value that represents a negative infinity value.**
**POSITIVE_INFINITY A special value that represents a positive infinity value.**
**prototype Represents the prototype for the number class.**
**toSource() Returns a string representation of the number object.**
**toString() Returns a string representing the specified number object.**
**valueOf() Returns the primitive value of a number object as a number data type.**



| | |
|---|---|
| toExponential(fractionDigits) | Returns exponential value as a string.<br>**Example:**<br>var num = 100; Num.toExponential(2); // returns '1.00e+2' |
| toFixed(fractionDigits) | Returns string of decimal value of a number based on specified fractionDigits.<br>**Example:**<br>var num = 100; Num.toFixed(2); // returns '100.00' |
| toLocaleString() | Returns a number as a string value according to a browser's locale settings.<br>**Example:**<br>var num = 100; Num.toLocaleString(); // returns '100' |
| toPrecision(precisionNumber) | Returns number as a string with specified total digits.<br>**Example:**<br>var num = 100; Num.toPrecision(4); // returns '100.0' |
| toString() | Returns the string representation of the number value.<br>**Example:**<br>var num = 100; Num.toString(); // returns '100' |
| valueOf() | Returns the value of Number object.<br>**Example:** var num = new Number(100); Num.valueOf(); // returns '100' |

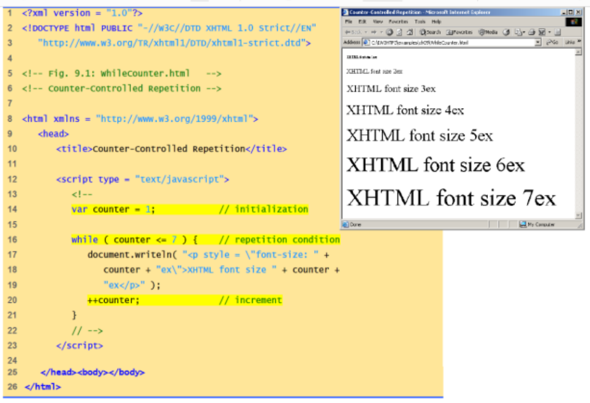
getDate() Returns the day of the month in int type from 1-31
getDay() Returns the day of the week from 0-6
getFullYear() Returns the year in four digits
getHours() Returns the hour and the value is from 0-23
getMilliseconds() Returns the milliseconds and the value is from 0-999
getMinutes() Returns the minutes and the value is from 0-59
getMonth() Returns the month whose value is from 0-11
getSeconds() Returns the seconds whose value is from 0-59
getTime() Returns the number of milliseconds since midnight Jan 1, 1970
getTimezoneOffset() Returns the time difference between UTC time and local time in minutes
getUTCDate() Returns the day of the month, according to universal time, value from 1-31
getUTCDay() Returns the day of the week, according to universal time, value from 0-6
getUTCFullYear() Returns the year, according to universal time in four digits
getUTCHours() Returns the hour, according to universal time, value from 0-23
getUTCMilliseconds() Returns the milliseconds, according to universal time, value from 0-999
getUTCMinutes() Returns the minutes, according to universal time, value from 0-59
getUTCMonth() Returns the month, according to universal time, value from 0-11
getUTCSeconds() Returns the seconds, according to universal time, value from 0-59
getYear() Deprecated.Use thegetFullYear()method instead
parse() Parses a date string and returns the number of milliseconds since January 1, 1970

setDate()
Sets the day of the month for a date object
setFullYear()
Sets the year in four digits for a date object

setHours()
Sets the hour of a date object
setMilliseconds()
Sets the milliseconds of a date object

setMinutes()
Set the minutes for a date object
setMonth()
Sets the month for a date object
setSeconds()
Sets the seconds for a date object
setTime()
Sets a date to a number of milliseconds after/before January 1, 1970
setUTCDate()
Sets the day of the month, according to universal time

setUTCFullYear()
Sets the year in four digits, according to universal time (four digits)
setUTCHours()
Sets the hour, according to universal time
setUTCMilliseconds()
Sets the milliseconds, according to universal time
setUTCMinutes()
Set the minutes, according to universal time
setUTCMonth()
Sets the month, according to universal time

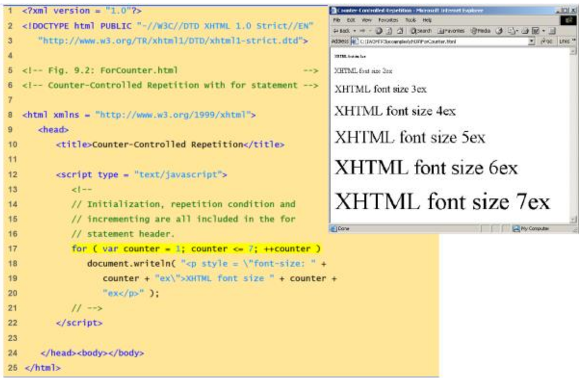


```
<?xml version = "1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 9.1: WhileCounter.html   -->
<!-- Counter-Controlled Repetition -->

<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>Counter-Controlled Repetition</title>

      <script type = "text/javascript">
         <!--
         var counter = 1;            // initialization

         while ( counter <= 7 ) {    // repetition condition
            document.writeln( "<p style = \"font-size: " +
               counter + "ex\">XHTML font size " + counter +
               "ex</p>" );
            ++counter;               // increment
         }
         // -->
      </script>

   </head><body></body>
</html>
```

```
XHTML font size 1ex
XHTML font size 2ex
XHTML font size 3ex
XHTML font size 4ex
XHTML font size 5ex
XHTML font size 6ex
XHTML font size 7ex
```

Counter-Controlled Repetition - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address
Done
My Computer

**While Statement**

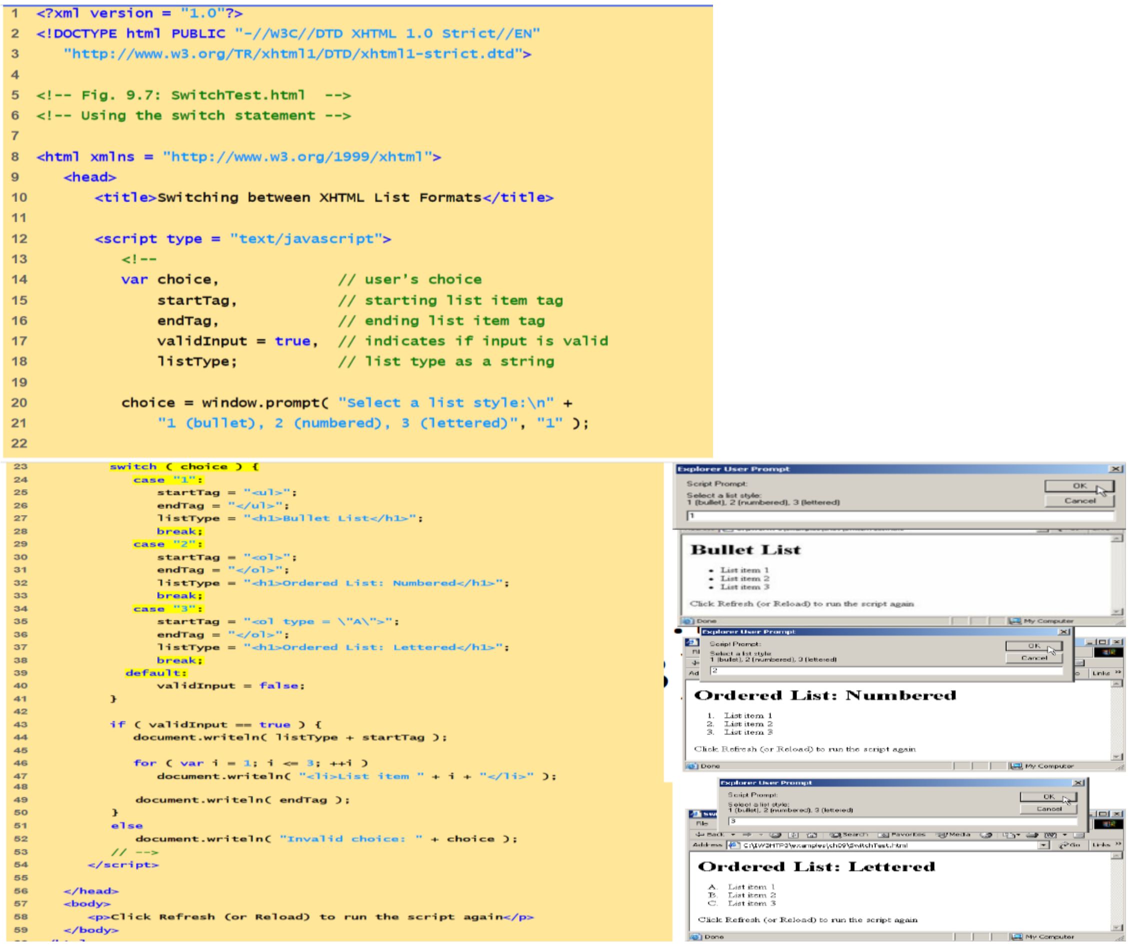


```
<?xml version = "1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 9.2: ForCounter.html                           -->
<!-- Counter-Controlled Repetition with for statement -->

<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>Counter-Controlled Repetition</title>

      <script type = "text/javascript">
         <!--
         // Initialization, repetition condition and
         // incrementing are all included in the for
         // statement header.
         for ( var counter = 1; counter <= 7; ++counter )
            document.writeln( "<p style = \"font-size: " +
               counter + "ex\">XHTML font size " + counter +
               "ex</p>" );
         // -->
      </script>

   </head><body></body>
</html>
```

```
XHTML font size 1ex
XHTML font size 2ex
XHTML font size 3ex
XHTML font size 4ex
XHTML font size 5ex
XHTML font size 6ex
XHTML font size 7ex
```

Counter-Controlled Repetition - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address
Done
My Computer

**For Loop**



```
<?xml version = "1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 9.7: SwitchTest.html  -->
<!-- Using the switch statement -->

<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>Switching between XHTML List Formats</title>

      <script type = "text/javascript">
         <!--
         var choice,              // user's choice
             startTag,            // starting list item tag
             endTag,              // ending list item tag
             validInput = true,   // indicates if input is valid
             listType;            // list type as a string

         choice = window.prompt( "Select a list style:\n" +
             "1 (bullet), 2 (numbered), 3 (lettered)", "1" );

         switch ( choice ) {
            case "1":
               startTag = "<ul>";
               endTag = "</ul>";
               listType = "<h1>Bullet List</h1>";
               break;
            case "2":
               startTag = "<ol>";
               endTag = "</ol>";
               listType = "<h1>Ordered List: Numbered</h1>";
               break;
            case "3":
               startTag = "<ol type = \"A\">";
               endTag = "</ol>";
               listType = "<h1>Ordered List: Lettered</h1>";
               break;
           default:
               validInput = false;
         }

         if ( validInput == true ) {
            document.writeln( listType + startTag );

            for ( var i = 1; i <= 3; ++i )
               document.writeln( "<li>List item " + i + "</li>" );

            document.writeln( endTag );
         }
         else
            document.writeln( "Invalid choice: " + choice );
         // -->
      </script>

   </head>
   <body>
      <p>Click Refresh (or Reload) to run the script again</p>
   </body>
```

```
Script Prompt:
Select a list style:
1 (bullet), 2 (numbered), 3 (lettered)
1
```

```
Bullet List
  • List item 1
  • List item 2
  • List item 3
Click Refresh (or Reload) to run the script again
```

```
Script Prompt:
Select a list style
1 (bullet), 2 (numbered), 3 (lettered)
2
```

```
Ordered List: Numbered
  1. List item 1
  2. List item 2
  3. List item 3
Click Refresh (or Reload) to run the script again
```

```
Script Prompt:
Select a list style:
1 (bullet), 2 (numbered), 3 (lettered)
3
```

```
Address C:\IW3HTP3\examples\ch09\SwitchTest.html
Ordered List: Lettered
  A. List item 1
  B. List item 2
  C. List item 3
Click Refresh (or Reload) to run the script again
```

## Roll Dice  js

```
<?xml version = "1.0"?>
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML
1.0 Strict//EN"
3 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">
6 <!-- Roll a Six-Sided Die 6000 Times -->
8 <html xmlns =
"http://www.w3.org/1999/xhtml">
9 <head>
10 <title>Roll a Six-Sided Die 6000 Times</title>
11
12 <script type = "text/javascript">
13 <!--
14 var face, frequency = [ , 0, 0, 0, 0, 0, 0 ];
17 for ( var roll = 1; roll <= 6000; ++roll ) {
18 face = Math.floor( 1 + Math.random() * 6 );
19 ++frequency[ face ];
20 }
document.writeln( "<table border = \"1\"" + 23
"width = \"100%\">" );
 24 document.writeln( "<thead><th width =
\"100\"" +
25 " align = \"left\">Face<th align = \"left\">" +
26 "Frequency</th></thead></tbody>" );
27
28 for ( face = 1; face < frequency.length; ++face )
29 document.writeln( "<tr><td>" + face +
"</td><td>" +
30 frequency[ face ] + "</td></tr>" ); 3
1
32 document.writeln( "</tbody></table>" );
33 // --> 34 </script>
35 36 </head>
37 <body>
38 <p>Click Refresh (or Reload) to run the script
again</p>
 39 </body>
 40 </html>
```

## Linear search

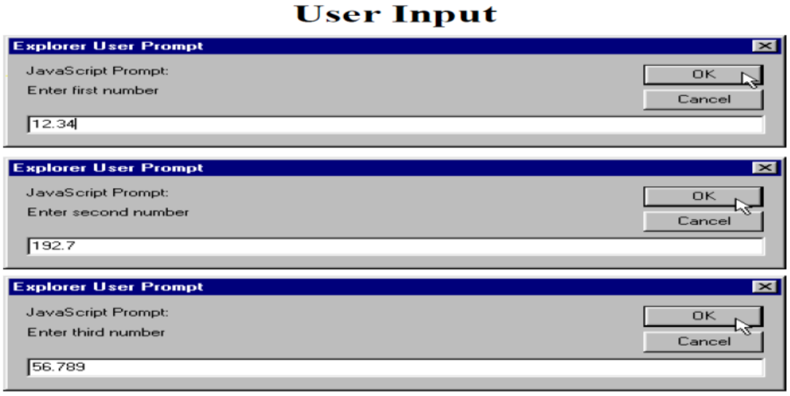
```
<?xml version = "1.0"?>
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML
1.0 Strict//EN"
3"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">
5 <!-- Fig. 11.10: LinearSearch.html -->
6 <!-- Linear Search of an Array -->
8 <html xmlns =
"http://www.w3.org/1999/xhtml">
9 <head>
10 <title>Linear Search of an Array</title>
12 <script type = "text/javascript">
13 <!--
14 var a = new Array( 100 );
16 // fill Array with even integer values from 0 to
198
17 for ( var i = 0; i < a.length; ++i )
18 a[ i ] = 2 * i;
// function called when "Search" button is pressed
21 function buttonPressed()
22 {
23 var searchKey = searchForm.inputVal.value;
24
25 // Array a is passed to linearSearch even though
it
26 // is a global variable. Normally an array will
27 // be passed to a method for searching.
28 var element = linearSearch( a, parseInt(
searchKey ) );
29
30 if ( element != -1 )
31 searchForm.result.value =
32 "Found value in element " + element;
33 else
34 searchForm.result.value = "Value not found";
35 }

function linearSearch( theArray, key )
39 {
40 for ( var n = 0; n < theArray.length; ++n )
41 if ( theArray[ n ] == key )
42 return n;
43
44 return -1;
45 }
46 // -->
47 </script>
48
49 </head>
50
51 <body>
52 <form name = "searchForm" action = "">
53 <p>Enter integer search key<br />
54 <input name = "inputVal" type = "text" />
55 <input name = "search" type = "button" value =
"Search"
56 onclick = "buttonPressed()" /><br /></p>
57
58 <p>Result<br />
59 <input name = "result" type = "text" size = "30"
/></p>
60 </form>
61 </body>
62 </html>
```

## BinarySearch.html

```
<?xml version = "1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<!-- Fig. 11.11 : BinarySearch.html -->
<!-- Binary search -->
<html xmlns = "http://www.w3.org/1999/xhtml">
<head>
<title>Binary Search</title>
<script type = "text/javascript">
<!--
var a = new Array( 15 );

for ( var i = 0; i < a.length; ++i )
a[ i ] = 2 * i;
  function buttonPressed()
{
var searchKey = searchForm.inputVal.value;
searchForm.result.value =
"Portions of array searched\n";
var element =
binarySearch( a, parseInt( searchKey ) );
if ( element != -1 )
searchForm.result.value +=
"\nFound value in element " + element;
else
searchForm.result.value += "\nValue not found";
}
function binarySearch( theArray, key )
{
var low = 0; // low subscript
var high = theArray.length - 1; // high subscript
var middle; // middle subscript
while ( low <= high ) {
middle = ( low + high ) / 2;
// The following line is used to display the
// part of theArray currently being manipulated
// during each iteration of the binary
// search loop.
buildOutput( theArray, low, middle, high );
if ( key == theArray[ middle ] ) // match
return middle;
else if ( key < theArray[ middle ] )
high = middle - 1; // search low end of array
else
low = middle + 1; // search high end of array
}
return -1; // searchKey not found
}
// Build one row of output showing the current
// part of the array being processed.
function buildOutput( theArray, low, mid, high )
{
for ( var i = 0; i < theArray.length; i++ ) {
if ( i < low || i > high )
searchForm.result.value += " ";
// mark middle element in output
else if ( i == mid )
searchForm.result.value += theArray[ i ] +
( theArray[ i ] < 10 ? "* " : "* " );
else
searchForm.result.value += theArray[ i ] +
( theArray[ i ] < 10 ? " " : " " );
}
searchForm.result.value += "\n";
}
// -->
</script>
</head>
<body>
 <form name = "searchForm" action = "">
 <p>Enter integer search key<br />
<input name = "inputVal" type = "text" />
<input name = "search" type = "button" value =
"Search" onclick = "buttonPressed()" /><br /></p>
<p>Result<br />
<textarea name = "result" rows = "7" cols = "60">
</textarea></p>
</form>
</body>
</html>
```

## InitArray3.html

```
<?xml version = "1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<!-- Fig. 11.13: InitArray3.html -->
<!-- Initializing Multidimensional Arrays -->
<html xmlns = "http://www.w3.org/1999/xhtml">
<head>
<title>Initializing Multidimensional Arrays</title>
<script type = "text/javascript">
<!--
function start()
{
var array1 = [ [ 1, 2, 3 ], // first row
[ 4, 5, 6 ] ]; // second row
var array2 = [ [ 1, 2 ], // first row
[ 3 ], // second row
20 [ 4, 5, 6 ] ]; // third row
22 outputArray( "Values in array1 by row", array1 );
23 outputArray( "Values in array2 by row", array2 );
24 }function outputArray( header, theArray )
27 {
28 document.writeln( "<h2>" + header + "</h2><tt>" );
30 for ( var i in theArray ) {
32 for ( var j in theArray[ i ] )
33 document.write( theArray[ i ][ j ] + " " );
35 document.writeln( "<br />" );
36 }
38 document.writeln( "</tt>" );}
40 // -->
41 </script>
42
43 </head><body onload = "start()"></body>
44 </html>
```


User Input

Explorer User Prompt
JavaScript Prompt:
Enter first number
12.34

Explorer User Prompt
JavaScript Prompt:
Enter second number
192.7

Explorer User Prompt
JavaScript Prompt:
Enter third number
56.789

```
1<!DOCTYPE html PUBLIC"-//W3C//DTD HTML 4.0 Transitional//EN">
2<HTML>
3<!--Fig. 16.3: maximum.html -->
5<HEAD>
6<TITLE>Finding the Maximum of Three Values</TITLE>
8<SCRIPT LANGUAGE = "JavaScript">
9varinput1 = window.prompt( "Enter first number", "0" );
10varinput2 = window.prompt( "Enter second number", "0" );
11varinput3 = window.prompt( "Enter third number", "0" );
13varvalue1 = parseFloat( input1 );
14varvalue2 = parseFloat( input2 );
15varvalue3 = parseFloat( input3 );
17varmaxValue = maximum( value1, value2, value3 );
19document.writeln( "First number: " + value1 +
20"<BR>Second number: " + value2 +
21"<BR>Third number: " + value3 +
22"<BR>Maximum is: " + maxValue );
24// maximum method definition (called from line 17)
25functionmaximum( x, y, z )
26{
27returnMath.max( x, Math.max( y, z ) );
28}
29</SCRIPT>
BODY>
33<P>Click Refresh (or Reload) to run the script again</P>
34</BODY>
35</HTML>
```

**Math.floor( a + Math.random() * b );**

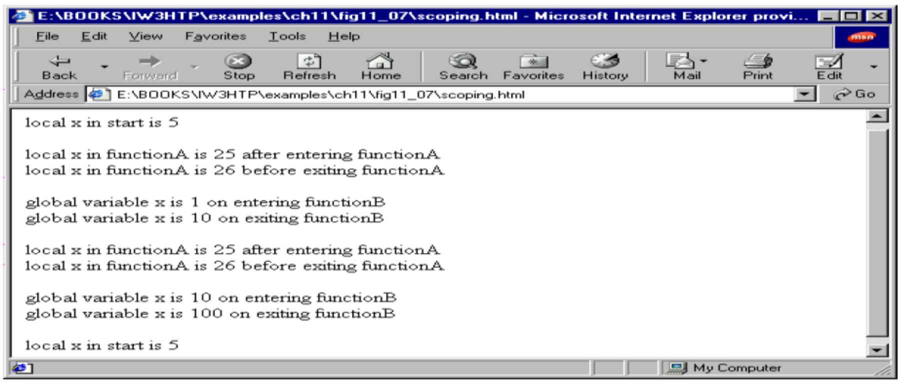
–a is the shifting value

•Equal to the first number in the desired range

–b is the scaling factor

•Equal to the width of the desired range

–Also possible to choose from sets of values other than ranges of consecutive integers


E:\BOOKS\IW3HTP\examples\ch11\fig11_07\scoping.html - Microsoft Internet Explorer provi...
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit
Address E:\BOOKS\IW3HTP\examples\ch11\fig11_07\scoping.html
Go
My Computer

```
local x in start is 5

local x in functionA is 25 after entering functionA
local x in functionA is 26 before exiting functionA

global variable x is 1 on entering functionB
global variable x is 10 on exiting functionB

local x in functionA is 25 after entering functionA
local x in functionA is 26 before exiting functionA

global variable x is 10 on entering functionB
global variable x is 100 on exiting functionB

local x in start is 5
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<!--Fig. 16.7: scoping.html -->
<HEAD>
<TITLE>A Scoping Example</TITLE>
<SCRIPT LANGUAGE ="JavaScript">
varx = 1; // global variable
functionstart()
{
varx = 5; // variable local to function start
document.writeln( "local x in start is " + x );
functionA(); // functionA has local x
functionB(); // functionB uses global variable x
functionA(); // functionA reinitializes local x
functionB(); // global variable x retains its value
document.writeln(
"<P>local x in start is " + x + "</P>" );
}
functionfunctionA()
{
varx = 25; // initialized each time functionA is called
document.writeln( "<P>local x in functionA is " + x +
" after entering functionA" );
++x;
document.writeln( "<BR>local x in functionA is " + x +
" before exiting functionA</P>" );
}
functionfunctionB()
{
document.writeln( "<P>global variable x is " + x +
" on entering functionB" );
x *= 10;
document.writeln( "<BR>global variable x is " + x +
" on exiting functionB</P>" );
}
</SCRIPT>
</HEAD>
<BODY ONLOAD = "start()"></BODY>
</HTML>
```

# BLOG PAGE

Home Page - Short Descriptions about travel diaries

Blog Page - Photos of the place visited and use image map / link tag for providing the information

Contact us – Use your intuitive approach and add the form fields for a contact us page

**Create a Javascript for the above travelblog to perform the functionalities given below**

**o Display the current day and time in the following format in any part of the home page**

** Today is : Friday**

** Time is : 11 AM : 30 : 38**

**o Use prompt box to get the name of the user when Main page is loaded**

**o Display a JavaScript Welcome Message Based on Time of Day with the name from prompt box**

**o Add a subscription Button for the Blog page. Once clicked get the E-Mail and mobile number by form fields.**

**o Use HTML 5 form field validation for E-Mail and Mobile number. Once submitted add a subscription confirmation message using Alert box with Customized message.**

** Add a Search field in the Blog to search a particular keyword**

 Validate the contact us page using JavaScript string methods (Length).

```
<!doctype html>
<html>
```

```
 <head>

 <title>My travel blog </title>
 <meta charset="utf-8" />
 <meta name="viewport" content="width=device-
width, initial-scale=1.0">
 <link rel="stylesheet" type="text/css"
href="../css/style2.css">
 <script src="mytravel.js"></script>

 </head>
 <body style="background-color:white;";>
 <header>
 <center>
 <h1> My travel blog </h1>
 </center>

 <div id="navbar">
 <ul>
 <li><a href="#">Home</a></li>
 <li><a href="#top">My blog </a></li>
 <li><a href="#about">About </a></li>
 <li><a href="#bottom">Contact Us </a></li>
 </ul>
 </div>
 </header>
 <script type="text/javascript">
 window.onload = function () {
 DisplayCurrentTime();
 };
 function DisplayCurrentTime() {
 var date = new Date();

 var hours = date.getHours() > 12 ? date.getHours()
- 12 : date.g
etHours();
 var am_pm = date.getHours() >= 12 ? "PM" :
"AM";
 hours = hours < 10 ? "0" + hours : hours;
 var minutes = date.getMinutes() < 10 ? "0" +
date.getMinutes() :
date.getMinutes();
var seconds = date.getSeconds() < 10 ? "0" +
date.getSeconds() :
date.getSeconds();
 time = "Time is "+ hours + " "+ am_pm + ":" +
minutes + ":" + s
econds + " " ;
 var lblTime =
document.getElementById("lblTime");
 lblTime.innerHTML = time;

 //var day = ["Monday", "Tuesday", "Wednesday",
"Thrusday", "Frid
ay","Saturday","Sunday"];
 //var myDay = myDate.getDay();
 day1 = "Today day is ";
```

```
 var lblDay=document.getElementById("lblDay");
 lblDay.innerHTML= day1;
 };
 </script>

 <section class="intro">

 <img src="images/mount.jpg" width="100%"
height="100%" alt=" 2 cc
">
 <div class="welcome">

 <h3>
 “A mind that is stretched by a new experience can
never go
back to its old dimensions.”
 </h3>>
 </div>
 <div class="time">
 <center><span id="lblDay"></span></center>
 <br>
 <center><span id="lblTime"></span></center>
</div>
 </section>
 <a name="about"></a>
 <br>
 <section class="about">
 <div class="welcome">
 <h2>
 About
 </h2>

 <img src="images/me.JPG" width="100%"
height="100%" alt=" 2 cc ">
 <h3>
 K V SHANMUKHASAI
 <br>
 Btech CSE<br>
 Vellore Institute of Technology<br>
 Vellore, Tamilnadu<br>

 </h3>
 </div>
 </section>
 <br>
 <br>
 <br>
 <a name="top"> </a>
<main class="container">

 <h2><center><span>MY BLOGS</span>
</center></h2>
 <div class="search-container">
 <form action="/action_page.php">
 <input type="text" size="20"
placeholder="Search.." name="sear
ch">
```

```
 </form>
 </div>
 </div>
 <section class="card">
 <img src="images/chennai.jpg" alt=""
width="200" height="200">
 <div>
 <h3>Trip to Mahabalipuram </h3>
 <p>
 Mamallapuram,
 </p>
 <a
href="https://www.mapquest.com/india/chennai2
82471390" class="btn">Directions</a>
 </div>
 </section>

 <a name="bottom"> </a>
 <div class="wrapper">
 <div class="title">
 <h1> Reach me </h1>
 </div>
 <div class="contact-form">
 <form action="mytravelblog.html" name="form "
onsubmit="returnva
lidatemail2();">
 <div class="input-fields">
 <input type="text" class="input"
placeholder="Name">
 <input type="text" class="input"
placeholder="Email Address">
 <input type="text" class="input"
placeholder="Phone">

 </div>
 <div class="msg">
 <textarea placeholder="Message"></textarea>
 <div class="btn">send</div>
 </div>
 </form>
 </div>
 </div></body>
 <script>
 AOS.init({
 offset: 400, // offset (in px) from the original
trigger point
 delay: 0, // values from 0 to 3000, with step 50ms
 duration: 1000 // values from 0 to 3000, with step
50ms
 });
 </script>
</html>

header{
 background-color: lightgray ;
 padding: 20px;
 text-align: center;
```

```
 position: fixed;
 width: 100%;
 z-index: 1;
 top: 0px;
 left: 0px;
 }
 h1{
 color:black;
 background-color: fff;
 font-size: 30px;
 text-transform: uppercase;
 font-family: Arial, Helvetica, sans-serif;
 text-align: center;


 border: 8px solid crimson;
 padding: 6px 12px;
 display: inline-block;
 border-radius: 24px;

 }
 #navbar ul {
 width: 90%;
 position: fixed;
 margin: 0;
 padding: 5px;
 list-style-type: none;
 text-align: right;
 background-color: #000;
 }

 #navbar ul li {
 display: inline;
 }

 #navbar ul li a {
 text-decoration: none;
font-size: 30px;
 padding: 0.2em 1em;
 color: #fff;
 background-color: #000;
 }

 #navbar ul li a:hover {
 color: #000;
 background-color: #fff;
 }
 .intro{
 position: relative;


 }
 .intro .welcome{
 color: whitesmoke;

 position: absolute;
 padding: 10px;
 left: 80px;
 top: 150px;

 }
function validateemail()
{
var x=document.myform.email.value;
var y=document.myform.name1.value;
var atposition=x.indexOf("@");
var dotposition=x.lastIndexOf(".");
if(y.length< 4){
 alert("Name should be atleast 3 characters long.");
 return false;
 }
if (atposition<1 || dotposition<atposition+2 || dotposition+2>=x.length){
 alert("Please enter a valid e-mail address ");
 return false;
 }
}
function validateemail2()
{
var x=document.myform2.email2.value; var y=document.myform2.number.value;
var atposition=x.indexOf("@");
var dotposition=x.lastIndexOf(".");
if(y.length> 10){
 alert("Enter valid number");
 return false;
 }
else if (atposition<1 || dotposition<atposition+2 || dotposition+2>=x.length){

 alert("Please enter a valid e-mail address ");
 return false;
 }
 else{
 alert("Subscribed successfully");
 return true;
 }}
var txt;
var person = prompt("Please enter your name:", "Your name ");
if (person == null || person == "") {
 txt = " ";
} else {
 window.alert("Welcome to the website: "+ person);
 txt = "Hello " + person + "! How are you today?";
}
document.getElementById("demo").innerHTML = txt;
```

**Write a JavaScript to Validate an HTML form which will collect book information that includes the following:**
**• Title of Book (Maximum string length accepted : 75)**
**• Author's Names (minimum 1 author and maximum 5, use input fields)**
**• ISBN Number( Format OF EXAMPLE USING REGULAR EXPRESSION 978-3-16-148410-0)**
**• Publisher (Maximum string length accepted : 30) • Edition (string type only) • Price (Validation for number type ) • Date of Publication**

```
<!doctype html>
<html>
   <head>
      <title>Book Entry  </title>
      <meta charset="utf-8" />
      <meta name="viewport" content="width=device-width, initial-scale=1.0">

      <link rel="stylesheet" type="text/css" href="../css/book.css">
      <script src="book.js"></script>

   </head>
   <body>
      <div class="wrapper">
         <div class="title">
           Book Registration
         </div>

         <form name="myForm" onsubmit="return validateForm()" action="trail.php" method="post">
         <div class="form">
           <div class="inputfield">
             <label>Title of the book </label>
             <input type="text" class="input" name="title" required >
           </div>

           <div class="inputfield">
             <label>Author 1</label>
             <input type="text" class="input"
placeholder="Required" required>
           </div>
           <div class="inputfield">
            <label>Author 2</label>
            <input type="text" class="input" >
          </div>
          <div class="inputfield">
            <label>Author 3</label>
            <input type="text" class="input" >
          </div>
          <div class="inputfield">
            <label>Author 4</label>
            <input type="text" class="input" >
          </div>
          <div class="inputfield">
            <label>Author 5</label>
            <input type="text" class="input" >
          </div>
           <div class="inputfield">
             <label> ISBN Number</label>
             <input type="text" class="input"
name="isbn" required>
           </div>
            <div class="inputfield">
             <label>Publisher </label>
             <input type="text" class="input"
name="publisher" required>
           </div>
           <div class="inputfield">
             <label>Edition</label>
             <input type="text" class="input"
name="edition" required>
           </div>

           <div class="inputfield">
             <label>Cost of the book</label>
             <input type="number" class="input"
name="price" required>
           </div>

           <div class="inputfield">
            <label>Date of Publication</label>
            <input type="date" class="input"
required>
          </div>

           <div class="inputfield terms">
             <label class="check">
               <input type="checkbox" required>
               <span class="checkmark"></span>
             </label>
             <p>Agreed to terms and
conditions</p>
           </div>

          <div class="inputfield">
            <input type="submit"
value="Register" class="btn">
          </div>
         </div>
         </form>
      </div>
   </body>
</html>
```

**Javascript**

```
function validateForm(input) {

  var y =
document.forms["myForm"]["publisher"].valu
e;
  var x =
document.forms["myForm"]["title"].value;
  var a =
document.forms["myForm"]["edition"].value;
  var b =
document.forms["myForm"]["price"].value;
  var c =
document.forms["myForm"]["isbn"].value;
  var v =
document.forms["myForm"]["price"].value;
  var r = Math.floor( Math.random() *v
*10000000 ) ;
  var d = /^\d\d\d-\d-\d\d-\d\d\d\d\d\d-\d$/;
  var letters = /^[0-9]+$/;
  var letter = /^[a-zA-Z]+$/;

  if (x.length > 75 || y.length > 30 ||
(a).match(letters) || (b.match.letter) ||
(!c.match(d)) ) {

    if(x.length > 75){
    alert("Title is too long, please make it
short");
    return false;
    }
```

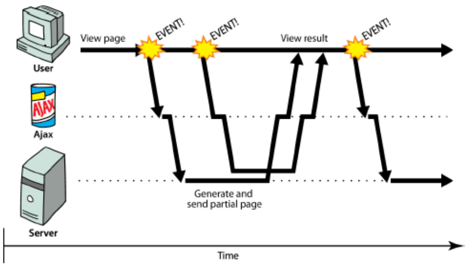
```
   if(y.length > 30 ){
     alert("Publisher name is too long, please
make it short");
     return false;
   }

   if((a).match(letters))
     {
     alert('For edition, Only String  is accepted :
you can try another');
     return false;
     }


     if((b).match(letter))
     {
     alert('For Cost, Only number  is accepted :
you can try another');
     return false;
     }

   if (!c.match(d))
   {
     alert( "The ISBN number entered is
invalid!" );
     return false;
   }
  }
  else{

   localStorage.setItem("textvalue",x);
   localStorage.setItem("isbn",c);
   localStorage.setItem("random",r);

   return true;

  }

}

<html>
   <head>
      <title>
         Check out
      </title>
      <meta charset="utf-8" />
      <meta name="viewport"
content="width=device-width, initial-
scale=1.0">
```

```
      <link rel="stylesheet" type="text/css"
href="../css/result.css">
   </head>
   <body>
      <center>
      <h1>
         Check-out
      </h1>
      <h2>

         <form name="myForm"
onsubmit="return validateForm()"
action="submit.html" >
         <table id="confirm">

            <tr>

               <th> Bill Id   </th>
               <th> <span id="random"></span>
</th>

             </tr>
            <tr>

               <td>ISBN number </td>
               <td> <span id="isbn"></span>
</td>

             </tr>
             <tr>

               <th> Title of the book  </th>
               <th> <span id="result"></span>
</th>

             </tr>
             <tr>

               <td>Date of Issue </td>
               <td> <span id="dd"></span>-<span
id="ddd"></span>-<span id="dddd"></span>
</td>

             </tr>
             <tr>

               <th>Date of Return </th>
               <th> <span id="rd"></span>-<span
id="rdd"></span>-<span id="rddd"></span>
</th>
```

```
             </tr>

           </table>
           <br>
           <input type="submit"  class= "button"
value="Confirm" class="btn">
          </form>

          <script>

document.getElementById("result").innerHT
ML= localStorage.getItem("textvalue");

document.getElementById("isbn").innerHTML
= localStorage.getItem("isbn");

document.getElementById("random").innerH
TML= localStorage.getItem("random");
             var d = new Date();
             var dd = d.getDate();
             var rd = d.getDate();
             var ddd = d.getMonth()+1;
             var rdd = d.getMonth()+2;
              var dddd = d.getFullYear();
              var rddd = d.getFullYear();

document.getElementById("dd").innerHTML =
dd;

document.getElementById("ddd").innerHTML
= ddd;

document.getElementById("dddd").innerHTM
L = dddd;

document.getElementById("rd").innerHTML =
rd;

document.getElementById("rddd").innerHTM
L = rddd;

document.getElementById("rdd").innerHTML
= rdd;



          </script>
       </h2>
    </center>
    </body>
</html>
```

# AJAX & J QUERY

•Not all AJAX apps involve XML
•Combination of technologies
•XHTML, CSS, DOM
•XML, XSLT, XMLHttp, JavaScript
•Some server scripting language
•A method for building more responsive and interactive applications

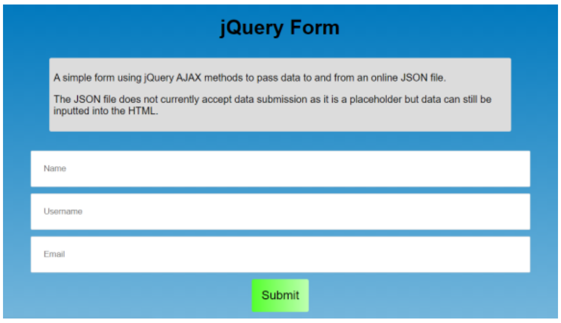


method
how to fetch the request from the server (default "post")
parameters
query parameters to pass to the server, if any
asynchronous (default true), contentType, encoding, requestHeaders
status the request's HTTP error code (200 = OK, etc.)
statusText HTTP error code text
responseText the entire text of the fetched page, as a String
responseXML the entire contents of the fetched page, as an XML DOM tree (seen later)

**What jQuery Does**
“Unobtrusive” JavaScript –separation of behaviorfrom structure
CSS –separation of stylefrom structure
Allows adding JavaScript to your web pages
Advantages over justJavaScript
Much easier to use
Eliminates cross-browser problems
HTML to CSS to DHTML

.each()iterate over the set
.size()number of elements in set
.end()reverts to the previous set
.get(n)get just the nth element (0 based)
.eq(n)get just the nth element (0 based) also .lt(n) & .gt(n)
.slice(n,m)gets only nth to (m-1)th elements
.not(‘p’)don’t include ‘p’ elements in set
.add(‘p’)add <p> elements to set
.remove() removes all the elements from the page DOM
.empty()removes the contents of all the elements
.filter(fn/sel)selects elements where the func returns true or sel
.find(selector)selects elements meeting the selector criteria
.parent()returns the parent of each element in set
.children()returns all the children of each element in set
.next()gets next element of each element in set
.prev()gets previous element of each element in set
.siblings()gets all the siblings of the current element

**Jquery-Ajax form validation:**



A simple form using jQuery AJAX methods to pass data to and from an online JSON file.

The JSON file does not currently accept data submission as it is a placeholder but data can still be inputted into the HTML.

```
  <h1 class="header">jQuery Form</h1>

  <div class="intro-text">
    <p>
      A simple form using jQuery AJAX methods to pass data to and from an
      online JSON file.
    </p>
    <p>
      The JSON file does not currently accept data submission as it is a
      placeholder but data can still be inputted into the HTML.
    </p>
  </div>
  <form
    class="users-form"

action="https://jsonplaceholder.typicode.com/users"
  >
    <label for="name">Name</label>
    <p class="validation-error-text"></p>
    <input class="form-input" type="text" placeholder="Name" id="name" />
    <label for="username">Username</label>
    <input
      class="form-input"
      type="text"
      placeholder="Username"
      id="username"
    />
    <label for="email">Email</label>
    <input class="form-input" type="text" placeholder="Email" id="email" />
    <label for="submitButton"></label>
    <button class="submit-button" type="submit" id="submitButton">
      Submit
    </button>
  </form>

  <h2 class="header">Users</h2>

  <div class="users"></div>
```

Jquery- AJax

```
/* Using .ajax method to retrieve data from external JSON url,
.map method to create a new array of items and .append method to input data into the DOM
*/

// .ajax 'get' used to grab data from URL
$(document).ready(function() {
 $.ajax({
  method: "get",
  url: "https://jsonplaceholder.typicode.com/users"
,
  dataType: "json"
 }).done(function(data) {
  // Using .map method to retrieve data from array
  $.map(data, function(user, i) {
   // Using .append to insert data in the DOM
   $(".users").append(
    "<div>" +
     "<p>" +
     "Name: " +
     user.name +
     "</p>" +
     "<p>" +
     "Username: " +
     user.username +
     "</p>" +
     "<p>" +
     "Email: " +
     user.email +
     "</p>" +
     "</div>"
   );
  });
 });

 /* Checking if an item with the attribute "submit" is clicked
  with the submit method and passing the event (e)*/
  $(".users-form").submit(function(e) {
    e.preventDefault();
    var name = $("#name").val(); // Grabbing values of form input fields
    var username = $("#username").val();
    var email = $("#email").val();
    // (this) is the form element and the action attribute is checked to grab the URL
    var url = $(this).attr("action");

    // Validation preventing form from being submitted if values are missing

    var input = $(".form-input");

    /* .each method used to iterate through each input
   with class ".form-input" */

    $(input).each(function(index) {
      var inputValue = $(this).val(); // this being used to grab current element
      var inputLength = inputValue.length;
      if (inputLength < 1) {
        // Adding class with .add method to highlight empty input fields
        input[index].classList.add("validation-failed");
        $(".validation-error-text").html(
          "Please enter a value in the highlighted fields."
        );
      } else {
        // Removing class and error message if value is entered on resubmission
        input[index].classList.remove("validation-failed");
        $(".validation-error-text").html("");
      }
    });

    /* Checking if the validation-failed class is present,
    if it is then return false will be used to stop the form
    from being submitted. Return false couldn't be used in the
    .each method as it would break the loop.*/

    if ($("input").hasClass("validation-failed")) {
      return false;
    }
    // .post method used to send data to URL
    $.post(url, {
      name: name, // Determining the data which is sent
      username: username,
      email: email
      // .done method runs this function when .post method is completed
    }).done(function(data) {
      // .prepend method used to insert data at the beginning of the users div

      $(".users").prepend(
        "<div>" +
          "<p>" +
          "Name: " +
          name +
          "</p>" +
          "<p>" +
          "Username: " +
          username +
          "</p>" +
          "<p>" +
          "Email: " +
          email +
          "</p>" +
          "</div>"
      );
    });
  });
});
```

# MOD4&5 – PHP, MYSQL

**strlen length**
**strpos indexOf**
**substr substring**
**strtolower, strtoupper toLowerCase, toUpperCase**
**trim trim**
**explode, implode split, join**
**strcmp compareTo**

**Array functions**
**function name(s) description**
**count number of elements in the array**
**print_r print array's contents**
**array_pop, array_push, array_shift, array_unshift using array as a stack/queue**
**in_array, array_search, array_reverse,**
**sort, rsort, shuffle**
**searching and reordering**
**array_fill, array_merge, array_intersect,**
**array_diff, array_slice, range**
**creating, filling, filtering**
**array_sum, array_product, array_unique,**
**array_filter, array_reduce**
**processing elements**

```
<?php$file=fopen("welcome.txt","r");
fclose($file);
?>
```

The fread() reads from an open file.
•The function will stop at the end of the file or when it reaches the specified length, whichever comes first.
•This function returns the read string, or FALSE on failure.
fread(file,length)
Parameter
Description
file
Required. Specifies the open file to read from
length
Required. Specifies the maximum number of bytes to read

Check End-of-file
•The feof() function checks if the "end-of-file" (EOF) has been reached.
•The feof() function is useful for looping through data of unknown length.
•Note: You cannot read from files opened in w, a, and x mode!

```
<?php
$file = “name.txt”;
if (feof($file)) echo "End of file";
?>
```

Reading a File Line by Line

```
<?php$file = fopen("welcome.txt", "r") or exit("Unable to open file!");
//Output a line of the file until the end is reached
while(!feof($file))
{echo fgets($file). "<br>";}
fclose($file);?>
```

Reading a File Character by Character

```
<?php$file=fopen("welcome.txt","r") or exit("Unable to open file!");
while (!feof($file)){echo fgetc($file);}fclose($file);
?>
```

Writing file

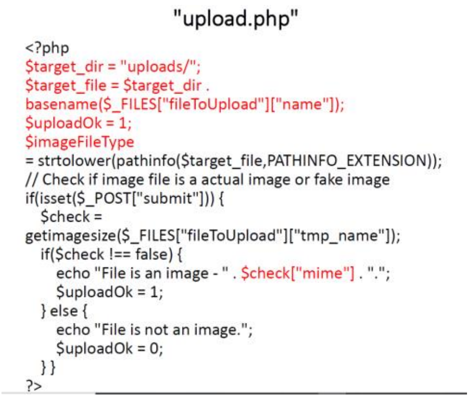
```
<?php
$fileName= "/doc/myFile.txt";
$fp= fopen($fileName,"w");
if( $fp== false )
{
echo ( "Error in opening file" );
exit();
}
fwrite( $fp, "This is a sample text to write\n" );
?>
```

mkdir(path,mode,recursive,context)

PHP Remove Directory Functions

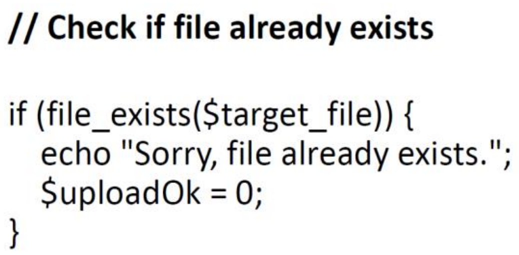
rmdir(dir,context)

- The rmdir() function removes an empty directory.
- This function returns TRUE on success, or FALSE on failure.

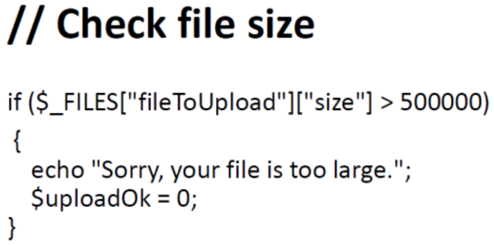


"upload.php"

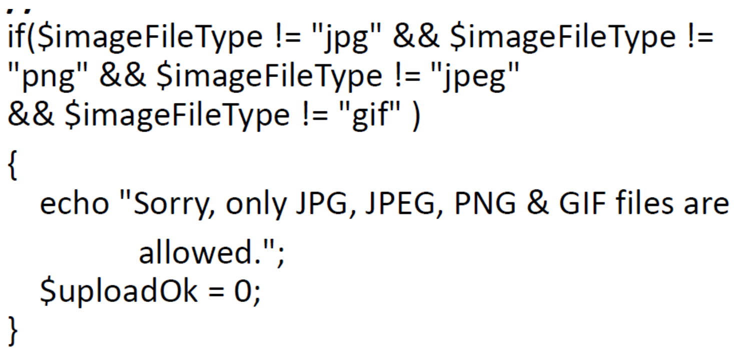
```
<?php
$target_dir = "uploads/";
$target_file = $target_dir .
basename($_FILES["fileToUpload"]["name"]);
$uploadOk = 1;
$imageFileType
= strtolower(pathinfo($target_file,PATHINFO_EXTENSION));
// Check if image file is a actual image or fake image
if(isset($_POST["submit"])) {
   $check =
getimagesize($_FILES["fileToUpload"]["tmp_name"]);
   if($check !== false) {
      echo "File is an image - " . $check["mime"] . ".";
      $uploadOk = 1;
   } else {
      echo "File is not an image.";
      $uploadOk = 0;
   }}
?>
```



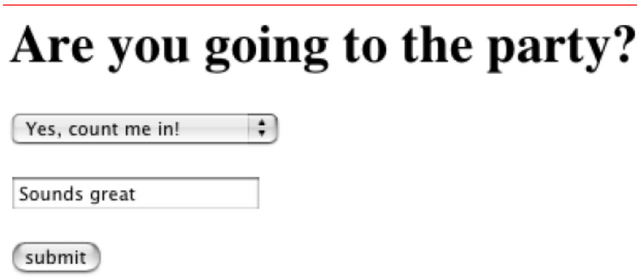
```
// Check if file already exists

if (file_exists($target_file)) {
   echo "Sorry, file already exists.";
   $uploadOk = 0;
}
```



```
// Check file size

if ($_FILES["fileToUpload"]["size"] > 500000)
 {
   echo "Sorry, your file is too large.";
   $uploadOk = 0;
}
```



```
if($imageFileType != "jpg" && $imageFileType !=
"png" && $imageFileType != "jpeg"
&& $imageFileType != "gif" )
{
   echo "Sorry, only JPG, JPEG, PNG & GIF files are
            allowed.";
   $uploadOk = 0;
}
```



Are you going to the party?

Yes, count me in!

Sounds great

Sending e-mail with PHP
You may need to specify the location of your mail server in php.ini

```
<html>
<head>
<title>Invitation</title>
<body>
<h1> Are you going to the party? </h1>
<form
method="POST"action="response.php">
<p>
<select name="attend">
<option selected value="Y"> Yes, count me in!
</option>
<option value="N"> Sorry, can't be bothered
</option>
</select>
</p>
<p><input
name="comment"type=textvalue="*type
comments in here*"/></p>
<p><input
type="submit"value="submit"></p>
</form>
</body></html>
```

response.php

```
<?php
$mailto= "a_______@localhost";
$subject= "Party";
$message= "";
$comment= $_POST['comment'];
if ($comment== "*type comments in here*") {
$comment= "I have no comment";
}
$willgo= $_POST['attend'];
if ($willgo== "Y") {
$message.= "Yes I am going\n";
}
elseif ($willgo== "N") {
$message.= "No!\n";
}
$message.= "$comment\n";
if (mail($mailto, $subject, $message)){
print "<h3>Mail was sent
successfully</h3><br/>";
}
else {
print "<h3>Could not send mail</h3><br/>";
}
?>
```

Multipurpose Internet Mail Extensions
MIME is an addition to the standard protocols that just sends simple text messages

Each data segment in a complex email is preceded by a number of content specification headers:Content-Type: image/jpeg; name=“abc.jpg”Content-Transfer-Encoding:7bit
PHP itself does not have support for sending emails with attachments

# HTTP Cookies

a cookie is a packet of information sent from the server to client, and then sent back to the server each time it is accessed by the client.
Introduces state into HTTP (remember: HTTP is stateless)
Cookiesare transferredbetween serverand clientaccording to http.Cookies can also be thought of as tickets used to identify clients and their orders

```
setcookie(name,value,expire,path,domain,secure)
<?php
header(“Set-Cookie:mycookie=myvalue;
path=/; domain=.xyz.org”);
?>
```

Creating cookies with setcookie()
Setcookie(name,value,expire, path, domain, secure)
e.g.

```
<?php
setcookie("MyCookie", $value,
time()+3600*24);
setcookie("AnotherCookie", $value,
time()+3600);
?>
```

Name: name of the file
Value: data stored in the file
Expire: data string defining the life time
Path: subset of URLs in a domain where it is valid
Domain: domain for which the cookie is valid
Secure: set to '1' to transmit in HTTPS

Reading cookies

```
<?php
foreach ($_COOKIEas $key=>$val) {
print $key . " => ". $val ."<br/>";
}
?>
```

Creating and using cookies example

```
<?php
setcookie("MyCookie", $value, time()+7200);
setcookie("AnotherCookie", $value, time()+7);
?>
<?php
foreach ($_COOKIEas $key=>$val) {
print $key . " => ". $val ."<br/>";
}
?>
```

Using headers

```
<?php
$strValue= "This is my first cookie";
setcookie ("mycookie", $strValue);
echo "Cookie set<br>";
?>
<!DOCTYPE html PUBLIC "=//W3C//DTD
XHMTL 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html
xmlns="http://www.w3.org/1999/xhmtl"xml:lang="en">
<head><title>PHP Script using Cookies</title>
</head>
<body>
<?php
echo “<p> A cookie has been set. </p>”;
?>
</body>
</html>
```

Multiple data items
Use explode() e.g.

```
<?php
$strAddress= $_SERVER['REMOTE_ADDR'];
$strBrowser=
$_SERVER['HTTP_USER_AGENT'];
$strOperatingSystem= $_ENV['OS'];
$strInfo="$strAddress::$strBrowser::$strOperatingSystem";
setcookie ("somecookie4",$strInfo,
time()+7200);
?>
<?php
$strReadCookie= $_COOKIE["somecookie4"];
$arrListOfStrings= explode ("::",
$strReadCookie);
```

```
echo "<p>$strInfo</p>";
echo "<p>Your IP address is:
$arrListOfStrings[0] </p>";
echo "<p>Client Browser is:
$arrListOfStrings[1] </p>";
echo "<p>Your OS is: $arrListOfStrings[2]
</p>";
?>
```

## PHP COOKIES CRUD OPERATIONS

**PHP Create/Retrieve a Cookie**

```
<?php
$cookie_name ="user";
$cookie_value =“XYZ";
setcookie($cookie_name, $cookie_value,
time() + (86400*30),"/");// 86400 = 1 day?>
<html>
<body>
<?php
if(!isset($_COOKIE[$cookie_name]))
{echo"Cookie named '". $cookie_name ."' is
not set!";}
else
{echo"Cookie '". $cookie_name ."' is
set!<br>";echo"Value is:
".$_COOKIE[$cookie_name];}
?>
</body>
 </html>
```

**Modify a Cookie Value**

```
<?php
$cookie_name ="user";
$cookie_value =“XYZ";
setcookie($cookie_name, $cookie_value,
time() + (86400*30),"/");
?>
<html>
<body>
<?phpif(!isset($_COOKIE[$cookie_name]))
{echo"Cookie named '". $cookie_name ."' is
not set!";}
else{echo"Cookie '". $cookie_name ."' is
set!<br>";
echo"Value is: ".$_COOKIE[$cookie_name];}
?>
</body>
</html>
```

**Delete a Cookie**

```
<?php
// set the expiration date to one hour ago
setcookie("user","", time() -3600);?>
<html>
<body>
<?php
echo"Cookie 'user' is deleted.";
?>
</body>
</html>
```

Check if Cookies are Enabled

```
<?php
setcookie("test_cookie","test", time()
+3600,'/');?>
<html>
<body>
<?phpif(count($_COOKIE) >0)
{echo"Cookies are enabled.";}
else{echo"Cookies are disabled.";}?>
</body>
</html>
```

## RESUME MAKER ONLINE (Part-1)

Create a Login form to authenticate the user. User Regular Expression to validate the login details

• After successful login , use a HTML form with input fields for a resume (Minimum 8) (Details like Name, Birth date, Nationality, Address, Education, E-Mail and Mobile Number etc)

• Clicking on the submit button the script should verify data entered in the fields in server side.

• Use String Objects to convert all the test data entry to Upper case. Age calculation based on the Birth date selection.

• After successful validation, Display the resume on the output.php page all the fields and a Send button.

**Login.html**

```
<html>
<head>
<title> Login Details       </title>
<meta http-equiv="Content-Type"
content="text/html; charset=UTF-8">
<link rel="stylesheet" type="text/css"
href="login.css">
<script type="text/javascript">
function validate()
{
var h=document.myform.id.value;
var d = /^[0-9][0-9][A-Z][A-Z][A-Z][0-9][0-9][0-
9][0-9]$/;
if (!h.match(d))
{
alert( "The Registartion  Number             is
invalid!" ); return false;
}
}
</script>
</head>
<body>
<div class="wrapper">
<h3 align="center"> Student Login </h3>
<form name="myform" onsubmit="return
validate()" action="cat.html" method="pos t">
<table border=0 align="center"
cellpadding="20" cellspacing=0 height=400
width
=450>


<div class="inputfield">
<tr><td>Registration Number     :</td><td>
<input type="text"         name="id" required
></td> </div>
<div class="inputfield">
<tr><td>Password :</td><td><input
type="password" size="20" maxlength="40"
nam e="tdate" required></td></tr> </div>
<div class="inputfield">
<tr><td><input type="submit"
value="Register" class="btw" ></td>
<td><input type="reset" value="Reset"
class="btw"></td></tr></form></div>
</table>
</div>
</body></html>
```

**CAT.HTML**

```
<html>
<head>
<title>   </title>
<meta http-equiv="Content-Type"
content="text/html; charset=UTF-8">
<link rel="stylesheet" type="text/css"
href="login.css">
<script type="text/javascript"> function
validate()
{
var letters = /^[0-9]+$/;

var  name=document.myform.name.value;
var email = document.myform.email.value;
var dob = document.myform.bday.value; var
mob =  document.myform.mob.value; var add
= document.myform.address.value;
var interest = document.myform.int.value; var
nation = document.myform.nation.value; var
gender = document.myform.gender.value;

if (name == "") {
alert(" Enter your name") return false;
}
if((name).match(letters)){
alert(" String only Accepted "); return false;
}
if (email == "") {
alert(" Enter the email address"); return false;
} else if (/^([a-zA-Z0-9_\-\.]+)@([a-zA-Z0-9_\-
\.]+)\.([a- zAZ]{2,5})$/.test(email) == false) {
alert("Invalid Email address "); return false;
}
if (Mob <= 0) {
alert("Only Number are accepted "); return
false;
}
if (!mob.match(letters))
{
 alert(only numbers)
retrun false;
}
if (dob == "") {
alert("Add Dob"); return false;
}
if (interest == "") { alert("Add interest");
return false;
}
if (nation == "") { alert("Add Education");
return false;
}

if((nation).match(letters)){
alert("   Nationality Only String!"); return
false;
}
if (ed == "") { alert("Add Education"); return
false;
}
if (add == "") { alert("Add Address"); return
false;
}
if (gender == "") { alert("Add Gender"); return
false;
}
if(!(gender).match(letters)){
alert("   Gender Only String "); return false;
}

}
</script>

</head>
<body>
<div class="wrapper">
<h3 align="center"> Your Resume          </h3>
<form name="myform" onsubmit="return
validate()" action="cat.php" method="post ">
<table border=0 align="center"
cellpadding="20" cellspacing=0 height=400
width
=450>


<div class="inputfield">
<tr><td>Your Name       :</td><td><input
type="text"       name="name" id="name"
required></td> </div>

<div class="inputfield">
<tr><td>Email   :</td><td><input type="text"
            name="email" id="email" requi
red></td> </div>

<div class="inputfield">
<tr><td>Date of Birth :</td><td><input
type="date"       name="bday" id
="bday" required></td></tr>
<tr><td>Nationality       :</td><td><input
type="text"       name="nation" id="nation"
required></td> </tr>
<tr><td> Your Address   :</td><td><input
type="text"       name="add ress"
id="address" required></td> </tr>
<tr><td>Gender            :</td><td><input
type="text"       name="gender" i d="gender"
required></td> </tr>
<tr><td>Your Interests  :</td><td><input
type="text"       na me="int" id="int"
required></td></tr>
<tr><td>Phone Number :</td><td> <input
type="numbe r" name="mob" id="mob"
required></td></tr>
</div>
<div class="inputfield">
<tr><td><input type="submit"
value="Register" class="btw" ></td>

<td><input type="reset" value="Reset"
class="btw"></td></tr></form></div>
</table>
</div>
</body></html>
```

**CAT.PHP**

```
<div class="wrapper">
<?php header('Content-Type: text/html;
charset=UTF-8'); ?>
```

```
<center>
<?php


$name=$_POST['name'];
$email=$_POST['email'];
$bday = $_POST['bday'];
$nation=$_POST['nation'];
$address=$_POST['address'];
$gender = $_POST['gender'];
$int=$_POST['int'];
$mob=$_POST['mob'];
$date=date('Y')-date('Y',strtotime($bday));


echo " <h1> Your Profile            </h1> " ; echo
"<br>";
echo strtoupper($name); echo "<br>";
echo "<br>";
echo strtoupper($email); echo "<br>";
echo "<br>"; echo $bday; echo "<br>"; echo
"<br>";
echo strtoupper($nation); echo "<br>";
echo "<br>";
echo strtoupper($address); echo "<br>";
echo "<br>";
echo strtoupper($gender); echo "<br>";
echo "<br>";
echo strtoupper($int); echo "<br>";
echo "<br>"; echo $mob; echo "<br>"; echo
"<br>";

echo $date;
?>
</center>

</div>
<html>
<head>
<title> Your Resume      </title>

<link rel="stylesheet" type="text/css"
href="login.css">



<body>
<center>
<br>
<br>
<br>
<a href="login.html" target="_parent"
style="color:red;font- size:140%;">Back </a>

</center>
</body>
</html>
```

An LPG supply company wishes to automate the operations related to booking and delivery of LPG cylinders.
Develop a web application which serves this system following below guidelines

**Add this Delivery Status to the booking table as : DEY (DELAYED)**
**Functional Requirements:**
1. Add records into Consumer table (Use Web Form)
2. Add records into booking table (Use Web Form)
3. Display all the consumers based on the connection type
4. Update the booking based on the delivery status
5. Display the total number of booking based on the delivery status.
6. Display the Booking details by using consumer id (Consumer name, connection type, booking id and Delivery status).

Create HTML forms with PHP to perform the functional requirements. Use appropriate forms with your customized names to display the output.

**FUNCTIONS.PHP**

```
<?php
define('DBINFO','mysql:host=localhost;dbnam
e=assignment'); define('DBUSER','root');
```

```
define('DBPASS','');

function performQuery($query){
$con = new PDO(DBINFO,DBUSER,DBPASS);
$stmt = $con->prepare($query); if($stmt->execute()){
return true;
}else{
return false;
}
}
function fetchAll($query){
$con = new PDO(DBINFO, DBUSER, DBPASS);
$stmt = $con->query($query); return $stmt->fetchAll();
}
?>
```

**BOOKTABLE.PHP**

```
<?php
session_start(); include("functions.php");
?>
<html>
<head>
<title>Login </title>
<meta charset="utf-8" />
<meta name="viewport"
content="width=device-width, initial-scale=1.0">

<link rel="stylesheet" href="assignment.css"/>

</head>
<body>
<?php

if(isset($_POST['myForm'])){

$bid = $_POST['bid'];

$cid = $_POST['cid'];
$dstatus = $_POST['dstatus'];

$query = "INSERT INTO `book_table` (
'bid','cid','dstatus' ) VALUES ( '$bid', '$cid',
'$dstatus')";
if(performQuery($query)){
echo "<script>alert('Successfully Registred
')</script>";
}else{
echo "<script>alert('Unknown error occured
or Bill Id is already registred')</script>";
}
}
?>
<center><h1>Enter the Booking
details</h1></center><br><br>

<div class="wrapper">
<div class="title"> Booking Status
</div>
<div class="container" id="container">
<div class="form-container sign-up-container">
</div>
</div>
<form name="myForm" method="POST" >
<div class="form">
<div class="inputfield">
<label>Booking ID </label>
<input type="text" class="input" id="bid"
name="bid" required >
</div>
<div class="inputfield">
<label> Customer Id </label>
<input type="text" class="input" id="cid"
name="cid" required>
</div> <div class="inputfield">
<label>Status</label>
<select name="dstatus" id="dstatus">
<option value="DEL">Delivered</option>
<option value="NYD">Not Yet Delivered
</option>
<option value="DEY">Delayed </option>
</select> </div>
<div class="inputfield">
<input type="submit" value="Submit"
name="myForm" class="btn">
</div>
</div>
</form>
</div>

</body>
</html>
```

**CONSUMER.PHP**

```
<?php
session_start(); include("functions.php");
?>
<html>
```

```
<head>
<title>Login </title>
<meta charset="utf-8" />
<meta name="viewport"
content="width=device-width, initial-
scale=1.0">
<link rel="stylesheet" href="assignment.css"/>
</head>
<body>
<?php
if(isset($_POST['myForm'])){
$cid = $_POST['cid'];
$cname = $_POST['cname'];
$ctype = $_POST['ctype'];
$query.= "INSERT INTO `consumer_table` (
`cid`, `cname`, `ctype`) VALUES ( '$cid',
'$cname', '$ctype')";
if(performQuery($query)){
echo "<script>alert('Successfully Registred
')</script>";
}else{
echo "<script>alert('Unknown error occured
or Customer Id is already registred')</script>";
}
}
?>
<center><h1>Enter the Consumer
details</h1></center><br><br>
<div class="wrapper">
 <div class="title"> Status
</div>
<div class="container" id="container">
<div class="form-container sign-up-
container">
</div>
</div>
<form name="myForm" method="POST" >
<div class="form">
<div class="inputfield">
<label> Customer Id </label>
<input type="text" class="input" id="cid"
name="cid" required>
</div>
<div class="inputfield">
<label> Customer Name </label>
<input type="text" class="input" id="cname"
name="cname" required>
</div> <div class="inputfield">
<label>Type</label>
<select name="ctype" id="ctype">
<option value="DOM">Domestic </option>
<option value="COM">Commercial </option>
</select> </div>

<div class="inputfield">
<input type="submit" value="Submit"
name="myForm" class="btn">
</div>
</div>
</form>
</div>
</body>
</html>
```

**CUSTLOGIN.PHP**

```
<?php
session_start();
include("functions.php");
?>
<html>
<head>
<title>Login </title>
<meta charset="utf-8" />
<meta name="viewport"
content="width=device-width, initial-
scale=1.0">
<link rel="stylesheet" href="assignment.css"/>
</head>
<body>
<?php
if(isset($_POST['myForm'])){

$cid = $_POST['cid'];
$_SESSION['cid'] = $_POST['cid'];

$query .= "SELECT * from `consumer_table`;";
if(count(fetchAll($query)) > 0){ //this is to
catch unknown error.
foreach(fetchAll($query) as $row){
if($row['cid']==$cid){
$_SESSION['custdisplay'] = true;
header('location:custdisplay.php');

}else{
echo "<script>alert('Wrong login
details.')</script>";
}
}
}
}
```

```
?>
<center><h1>Enter the Customer Id
details</h1></center><br><br>


<div class="wrapper">




<div class="title"> Booking Status
</div>
<div class="container" id="container">
<div class="form-container sign-up-
container">
</div>
</div>
<form name="myForm" method="POST" >
<div class="form">
<div class="inputfield">
<label>Customer ID </label>
<input type="text" class="input" id="cid"
name="cid" required >
</div>

<div class="inputfield">
<input type="submit" value="Submit"
name="myForm" class="btn">
</div>
</form>
</div>
</body>
</html>
```

**CUSTDISPLAY.PHP**

```
<?php
session_start(); include("functions.php");
$cid = $_SESSION['cid'];
?>
<html>
<head>
<title>Your details </title>
<meta charset="utf-8" />
<meta name="viewport"
content="width=device-width, initial-
scale=1.0">
<link rel="stylesheet" href="assignment.css"/>
</head>
<body>
<?php
$query = "select * from `consumer_table`
where cid= '$cid'            ";
if(count(fetchAll($query))>0){
foreach(fetchAll($query) as $row){
?>
<center>

<div id="req">



<div class="row">
<div class="column">
<div class="card">
<br><br>
<h2>Customer Name:<b><?php echo
$row['cname'] ?></b></h4>
<p>Connection Type:<?php echo
$row['ctype'] ?></p>
<p>Connection Id:<?php echo $row['cid']
?></p>

</div>
</div>
</div>
</div>

<?php
}

}
?>

<?php


$query1 = "select * from `book_table`  where
cid= '$cid'          ";


if(count(fetchAll($query1))>0){
foreach(fetchAll($query1) as $row){

?>
<div id="req">
<div class="row">
<div class="column">
<div class="card">
<p>Booking Id: <?php echo $row['bid'] ?></p>
<p>Delivery Status:<?php echo
$row['dstatus'] ?></p>
</div>
</div>

</div>
</div>

<?php

}
}
?>
</center>
</body>
</html>
```

**DISPLAY.PHP**

```
<?php
session_start(); include("functions.php");
?>
<html>
<head>


<title>Login </title>
<meta charset="utf-8" />
<meta name="viewport"
content="width=device-width, initial-scale=1.0">


<link rel="stylesheet" href="assignment.css"/>


</head>
<body>

<div class="wrapper">

<div class="title"> Customer Details
</div>
<div class="container" id="container">
<div class="form-container sign-up-container">
<?php
$count1 = 0;
$count2 = 0;
$c = "DOM";

$query = "select * from `consumer_table`
            ";


if(count(fetchAll($query))>0){
foreach(fetchAll($query) as $row){

if($row['ctype']==$c){
$count1 = $count1 + 1;

}
else{
$count2 = $count2 + 1;

}
?>
<div id="req">
<div class="row">
<div class="column">
<div class="card">
<h4 id="pleft"><?php echo $row['cid'] ?>
&nbsp; <?php echo
$row['ctype'] ?> &nbsp;<?php echo
$row['cname'] ?></h4>

</div>
</div>
</div>
</div>
<?php
}
}else{
echo "No Previous Records ";
}
?>
</div>
</div>
<br><br>
<h4>Total Present Domestic Users <?php
echo $count1 ?></h4>
<h4> Total Present Commercial Users <?php
echo $count2 ?></h4>
</div>
</body>
</html>
```

**DELIVERYSTATUS.PHP**

```
<?php
session_start(); include("functions.php");
?>
<html>
<head>
<title>Login </title>
<meta charset="utf-8" />
<meta name="viewport"
content="width=device-width, initial-
scale=1.0">

<link rel="stylesheet" href="assignment.css"/>
</head>
<body>
<?php

$count1 = 0;
$count2 = 0;
$c = "DEL";
$query = "select * from `book_table`      ";
if(count(fetchAll($query))>0){
foreach(fetchAll($query) as $row){
if($row['dstatus']==$c){
$count1 = $count1 + 1;
}
else{
$count2 = $count2 + 1
}


}
?>
<div id="req">
<div class="row">
<div class="column">
<div class="card">
<h4 id="pleft"><?php echo $row['bid'] ?>
&nbsp; <?php echo
$row['cid'] ?> &nbsp;<?php echo
$row['dstatus'] ?></h4>
</div>
</div>
</div>
</div>
<?php
}
}else{
echo "No Previous Records ";
}
?>
</div>
</div>
<br><br>
<h4>Total Connections Delivered <?php echo
$count1 ?></h4>
<h4> Total Connections not yet delivered
<?php echo $count2 ?></h4>
</div>
</body>
</html>
```

# MOD6 - XML XSL

HTML
-uses tags and attributes
-content and formatting can be placed together
<p><font=”Arial”>text</font>
-tags and attributes are pre-determined and rigid
•XML
-uses tags and attributes
-content and format are separate; formatting is contained in a stylesheet
-allows user to specify what each tag and attribute means
XML Advantages(1)
•XML Separates Data from HTML
•XML Simplifies Data Sharing
•XML Simplifies Data Transport
•XML Simplifies Platform Changes
•XML Makes Your Data More Available
•XML is Used to Create New Internet Languages
DTD versus Schema
Limitations of DTD
•No constraints on character data
•Not using XML syntax
•No support for namespace
•Very limited for reusability and extensibility
Advantages of Schema
•Syntax in XML Style
•Supporting Namespace and import/include
•More data types
•Able to create complex data type by inheritance
•Inheritance by extension or restriction

Two Ways of Interfacing XML Documents with XML Applications
•Object-based: DOM(Document Object Model)
•Specified by W3C
•Tree is built
•The parser loads the XML doc into computer memory and builds a tree of objects for all elements & attributes
•Event-based: SAX(Simple API for XML)
•Originally a Java-only API.
•Developed by XML-DEV mailing list community
•No tree is built
•The parser reads the file and triggers events as it finds elements/attribute/text in the XML doc

Uses of JSON
•ItisusedwhilewritingJavaScriptbasedapplicationsthatincludesbrowserextensionsandwebsites.
•JSONformatisusedforserializingandtransmittingstructureddataovernetworkconnection.
•Itisprimarilyusedtotransmitdatabetweenaserverandwebapplications.
•WebservicesandAPIsuseJSONformattoprovidepublicdata.
•Itcanbeusedwithmodernprogramminglanguages.

Sending Data

•If you have data stored in a JavaScript object, you can convert the object into JSON, and send it to a server:

```
<!DOCTYPE html>
<html>
<body>
<h2>Convert a JavaScript object into a JSON string, and send it to the server</h2>
<script>
varmyObj= { name: "John", age: 31, city: "New York" };
varmyJSON= JSON.stringify(myObj);
window.location= "demo_json.php?x=" + myJSON;
</script>
</body>
</html>
```

Receiving Data

•If you receive data in JSON format, you can convert it into a JavaScript object:

```
varmyJSON='{"name":"John", "age":31, "city":"NewYork"}';
varmyObj=JSON.parse(myJSON);d
ocument.getElementById("demo").innerHTML= myObj.name;
```

Storing Data

•JSONmakesitpossibletostoreJavaScriptobjectsastext.

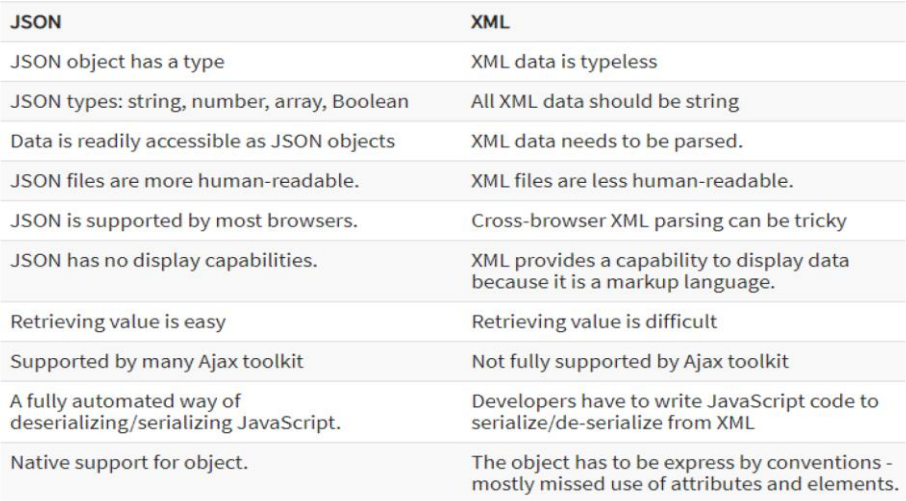
```
// Storing data:
myObj= {name:"John",age:31, city:"New York"};
myJSON=JSON.stringify(myObj);
localStorage.setItem("testJSON", myJSON);
// Retrieving data:
text = localStorage.getItem("testJSON");
obj=JSON.parse(text);
document.getElementById("demo").innerHTML= obj.name;
```

•{"name":"John"}

**Why JSON is Better Than XML?**

**•XML is much more difficult to parse than JSON.**

**•JSON is parsed into a ready-to-use JavaScript object.**



| JSON | XML |
|---|---|
| JSON object has a type | XML data is typeless |
| JSON types: string, number, array, Boolean | All XML data should be string |
| Data is readily accessible as JSON objects | XML data needs to be parsed. |
| JSON files are more human-readable. | XML files are less human-readable. |
| JSON is supported by most browsers. | Cross-browser XML parsing can be tricky |
| JSON has no display capabilities. | XML provides a capability to display data because it is a markup language. |
| Retrieving value is easy | Retrieving value is difficult |
| Supported by many Ajax toolkit | Not fully supported by Ajax toolkit |
| A fully automated way of deserializing/serializing JavaScript. | Developers have to write JavaScript code to serialize/de-serialize from XML |
| Native support for object. | The object has to be express by conventions - mostly missed use of attributes and elements. |

**basicxml**

```
<?xml version='1.0'?>
<bookstore>
  <book genre=‘autobiography’ publicationdate=‘1981’
        ISBN=‘1-861003-11-0’>
    <title>The Autobiography of Benjamin Franklin</title>
    <author>
      <first-name>Benjamin</first-name>
      <last-name>Franklin</last-name>
    </author>
    <price>8.99</price>
  </book>
  <book genre=‘novel’ publicationdate=‘1967’ ISBN=‘0-201-63361-2’>
    <title>The Confidence Man</title>
    <author>
      <first-name>Herman</first-name>
      <last-name>Melville</last-name>
    </author>
    <price>11.99</price>
  </book>
</bookstore>
```

**namespace**

```
<bk:BOOK xmlns:bk=“http://www.bookstuff.org/bookinfo”
            xmlns:money=“urn:finance:money”>
   <bk:TITLE>All About XML</bk:TITLE>
   <bk:AUTHOR>Joe Developer</bk:AUTHOR>
   <bk:PRICE money:currency=‘US Dollar’>
       19.99</bk:PRICE>
```

**dtd**

```
<!DOCTYPE BOOK [
<!ELEMENT BOOK (TITLE+, AUTHOR) >
<!ELEMENT TITLE (#PCDATA) >
<!ELEMENT AUTHOR        (#PCDATA) >
]>
```

**Xsd**

```
<xsd:schema id="NewDataSet“ targetNamespace="http://tempuri.org/schema1.xsd"
   xmlns="http://tempuri.org/schema1.xsd"
   xmlns:xsd="http://www.w3.org/1999/XMLSchema"
   xmlns:msdata="urn:schemas-microsoft-com:xml-msdata">
  <xsd:element name="book">
    <xsd:complexType content="elementOnly">
      <xsd:all>
        <xsd:element name="title" minOccurs="0" type="xsd:string"/>
        <xsd:element name="author" minOccurs="0" type="xsd:string"/>
      </xsd:all>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name=“Catalog" msdata:IsDataSet="True">
    <xsd:complexType>
      <xsd:choice maxOccurs="unbounded">
        <xsd:element ref="book"/>
      </xsd:choice>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

**embedded in an HTML SCRIPT**

```
<SCRIPT language=“xml” id=“XMLID”>
<SCRIPT type=“text/xml” id=“XMLID”>
<SCRIPT language=“xml” id=“XMLID”
        src=“mydocument.xml”>
```

Products.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="product.xsl"?>
<PRODUCTDATA>
<PRODUCT>
<PRODID id="P001" ></PRODID>
<PRODUCTNAME>Barbie Doll</PRODUCTNAME>
<DESCRIPTION>This is a toy for children in the age group below 5
years </DESCRIPTION>
<PRICE>240.00</PRICE>
<QUANTITY>12</QUANTITY>
</PRODUCT>
<PRODUCT>
<PRODID id="P002" ></PRODID>
<PRODUCTNAME>Mini Bus</PRODUCTNAME>
<DESCRIPTION>This is a toy for children in the age group of 5-10
years </DESCRIPTION>
<PRICE>420.00</PRICE>
<QUANTITY>6</QUANTITY>
</PRODUCT>
<PRODUCT>
<PRODID id="P003" ></PRODID>
<PRODUCTNAME>Car</PRODUCTNAME>
<DESCRIPTION>This is a toy for children in the age group of 10-15
years </DESCRIPTION>
<PRICE>600.00</PRICE>
<QUANTITY>21</QUANTITY>
</PRODUCT>
<PRODUCT>
<PRODID id="P004" ></PRODID>
<PRODUCTNAME>Air Plane</PRODUCTNAME>
<DESCRIPTION>This is a toy for children in the age group of 08-15
```

```
years </DESCRIPTION>
<PRICE>700.00</PRICE>
<QUANTITY>25</QUANTITY>
</PRODUCT>
</PRODUCTDATA>
```

Products.xsl

```
<?xml version="1.0" encoding="ISO-
8859-1"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/
Transform">
<xsl:template match="/">
<html>
<body>
<h2>Product Details</h2>
<table border="1">
<tr bgcolor="#CCCFFF">
<th align="left">Product Name</th>
<th align="left">Quantity</th>
<th align="left">Price</th>
</tr>
<xsl:for-each
select="PRODUCTDATA/PRODUCT">
<tr>
<td><xsl:value-of
select="PRODUCTNAME"/></td>
<td><xsl:value-of
select="QUANTITY"/></td>
<td><xsl:value-of select="PRICE"/></td>
</tr>
</xsl:for-each>
</table>
</body>
</html>
</xsl:template>
</xsl:stylesheet>
```

## Example 2:

## First.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl"
href="first.xsl"?>
<studentdata>
<student id="18BCE0971">
<REGNUM id="18BCE0971"
>18BCE0971</REGNUM>
<FNAME> SHANMUKHASAI </FNAME>
<LNAME> KALASAMUDRAM V </LNAME>
<DEGREE> BTECH </DEGREE>
<AYEAR> 2018 </AYEAR>
<CGPA> 8.59 </CGPA>
<AREAS> DRAWING, READING,PLAYING
</AREAS>
<EVENTS> <ONE>Hackothon</ONE>
<TWO>GAMEOTHON</TWO>
<THREE>IDEATHON</THREE>
</EVENTS>
<ADDRESS> <DNO>711</DNO>
<BLOCK>J </BLOCK>
<AREA>VIT UNIVERSITY</AREA>
</ADDRESS>
<EMAIL>
shanmukhkalasamudram@gmail.com
</EMAIL>
<MOBILE> 9494472861</MOBILE>
</student>
<student id="18BCVB">
<REGNUM id="18BDFG"
>18BCE0114</REGNUM>
<FNAME> CVB </FNAME>
<LNAME>J </LNAME>
<DEGREE> BTECH </DEGREE>
<AYEAR> 2018 </AYEAR>
<CGPA> 8.2 </CGPA>
<AREAS> DRAWING, READING,PLAYING
</AREAS>
<EVENTS> <ONE>Hackothon</ONE>
<TWO>GAMEOTHON</TWO>
<THREE>IDEATHON</THREE>
</EVENTS>
<ADDRESS> <DNO>711</DNO>
<BLOCK>J </BLOCK>
<AREA>VIT UNIVERSITY</AREA>
</ADDRESS>
<EMAIL> @gmail.com </EMAIL>
<MOBILE> 9449</MOBILE>
```

```
</student>
</studentdata>
```

# FIRST.XSL

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
<html>
<body>
<h2>Student Details</h2>
<table border="1">
<xsl:for-each select="studentdata/student">
<tr bgcolor="#CCCFFF">
<td align="left">Reg Num </td>
<td><xsl:value-of select="REGNUM"/></td>
</tr>
<tr >
<td align="left">First Name </td>
<td><xsl:value-of select="FNAME"/></td>
</tr>
<tr>
<td align="left">Last Name </td>
<td><xsl:value-of select="LNAME"/></td>
</tr>
<tr >
<td align="left">DEGREE </td>
<td><xsl:value-of select="DEGREE"/></td>
</tr>
<tr >
<td align="left">ADMISSION YEAR </td>
<td><xsl:value-of select="AYEAR"/></td>
</tr>
<tr >
<td align="left">CGPA </td>
<td><xsl:value-of select="CGPA"/></td>
</tr>
<tr >
<td align="left">AREAS OF INTERESTS </td>
<td><xsl:value-of select="AREAS"/></td>
</tr>
<tr >
<td align="left">EVENTS ATTENDED </td>
18BCE0971
</tr><tr>
<td align="left">1 </td>
<td><xsl:value-of
select="EVENTS/ONE"/></td>
</tr>
<tr>
<td align="left">2 </td>
<td><xsl:value-of
select="EVENTS/TWO"/></td>
</tr>
<tr>
<td align="left">3 </td>
<td><xsl:value-of
select="EVENTS/THREE"/></td>
</tr>
<tr >
<td align="left">ADDRESS </td>
</tr><tr>
<td><xsl:value-of
select="ADDRESS/DNO"/></td>
</tr>
<tr>
<td><xsl:value-of
select="ADDRESS/BLOCK"/></td>
</tr>
<tr>
<td><xsl:value-of
select="ADDRESS/AREA"/></td>
</tr>
<tr>
<td align="left">EMAIL </td>
<td><xsl:value-of select="EMAIL"/></td>
</tr><tr>
<td align="left">MOBILE </td>
<td><xsl:value-of select="MOBILE"/></td>
</tr>
</xsl:for-each>
</table>
</body>
</html>
</xsl:template>
</xsl:stylesheet>
```

# academiccalendar.dtd

**Code:**

```
<?xml encoding="UTF-8"?>

<!ELEMENT Academic_Details
(Reg_No,FirstName,LastName,Degree_Name,
Admission_Year,CGPA,Areaof_Interests,
List_of_Events_Attended,Address_Details, E-
Mail,Mobile_Number)>
<!ATTLIST Academic_Details
xmlns CDATA #FIXED ''
xmlns:xsi CDATA #FIXED
'http://www.w3.org/2001/XMLSchema-
instance' xsi:noNamespaceSchemaLocation
NMTOKEN #REQUIRED>
<!ELEMENT Reg_No (#PCDATA)> <!ATTLIST
Reg_No
xmlns CDATA #FIXED ''>
<!ELEMENT FirstName (#PCDATA)> <!ATTLIST
FirstName
xmlns CDATA #FIXED ''>
<!ELEMENT LastName (#PCDATA)> <!ATTLIST
LastName
xmlns CDATA #FIXED ''>
<!ELEMENT Degree_Name (#PCDATA)>
<!ATTLIST Degree_Name
xmlns CDATA #FIXED ''>
<!ELEMENT Admission_Year (#PCDATA)>
<!ATTLIST Admission_Year
xmlns CDATA #FIXED ''>

<!ELEMENT CGPA (#PCDATA)> <!ATTLIST
CGPA
xmlns CDATA #FIXED ''>
<!ELEMENT Areaof_Interests
(Areaof_Interests1,Areaof_Interests2,
Areaof_Interests3)>
<!ATTLIST Areaof_Interests xmlns CDATA
#FIXED ''>
<!ELEMENT List_of_Events_Attended
(Events_Attended1,Events_Attended2,
Events_Attended3)>
<!ATTLIST List_of_Events_Attended xmlns
CDATA #FIXED ''>
<!ELEMENT Address_Details
(Flatno,Street,City,State)> <!ATTLIST
Address_Details
xmlns CDATA #FIXED ''>
<!ELEMENT E-Mail (#PCDATA)> <!ATTLIST E-
Mail
xmlns CDATA #FIXED ''>
<!ELEMENT Mobile_Number (#PCDATA)>
<!ATTLIST Mobile_Number
xmlns CDATA #FIXED ''>
<!ELEMENT Areaof_Interests1 (#PCDATA)>
<!ATTLIST Areaof_Interests1
xmlns CDATA #FIXED ''>
<!ELEMENT Areaof_Interests2 (#PCDATA)>
<!ATTLIST Areaof_Interests2
xmlns CDATA #FIXED ''>
<!ELEMENT Areaof_Interests3 (#PCDATA)>
<!ATTLIST Areaof_Interests3
xmlns CDATA #FIXED ''>
<!ELEMENT Events_Attended1 (#PCDATA)>
<!ATTLIST Events_Attended1
xmlns CDATA #FIXED ''>
<!ELEMENT Events_Attended2 (#PCDATA)>
<!ATTLIST Events_Attended2 xmlns CDATA
#FIXED ''>
<!ELEMENT Events_Attended3 (#PCDATA)>
<!ATTLIST Events_Attended3
xmlns CDATA #FIXED ''>
<!ELEMENT Flatno (#PCDATA)> <!ATTLIST
Flatno
xmlns CDATA #FIXED ''>
<!ELEMENT Street (#PCDATA)> <!ATTLIST
Street
xmlns CDATA #FIXED ''>
<!ELEMENT City (#PCDATA)> <!ATTLIST City
xmlns CDATA #FIXED ''>
<!ELEMENT State (#PCDATA)> <!ATTLIST State
xmlns CDATA #FIXED ''><!ELEMENT E-Mail
(#PCDATA)> <!ELEMENT Mobile_Number
(#PCDATA)>
```

# acad_d.xsd

**Code:**

```
<xs:schema
attributeFormDefault="unqualified"
elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSche
ma">
<xs:element name="Academic_Details">
<xs:complexType>
<xs:sequence> <xs:element <xs:element
<xs:element <xs:element <xs:element
<xs:element <xs:element
type="xs:string" name="Reg_No"/>
type="xs:string" name="FirstName"/>
type="xs:string" name="LastName"/>
type="xs:string" name="Degree_Name"/>
type="xs:short" name="Admission_Year"/>
type="xs:float" name="CGPA"/>
name="Areaof_Interests">
<xs:complexType> <xs:sequence>
<xs:element type="xs:string"
name="Areaof_Interests1"/>
```

```
<xs:element type="xs:string"
name="Areaof_Interests2"/>
<xs:element type="xs:string"
name="Areaof_Interests3"/>
</xs:sequence> </xs:complexType>
</xs:element>
<xs:element
name="List_of_Events_Attended">
<xs:complexType> <xs:sequence>
<xs:element type="xs:string"
name="Events_Attended1"/>
<xs:element type="xs:string"
name="Events_Attended2"/>
<xs:element type="xs:string"
name="Events_Attended3"/>
</xs:sequence> </xs:complexType>
</xs:element>
<xs:element name="Address_Details">
<xs:complexType> <xs:sequence>
<xs:element type="xs:string"
name="Flatno"/> <xs:element
type="xs:string" name="Street"/> <xs:element
type="xs:string" name="City"/> <xs:element
type="xs:string" name="State"/>
</xs:sequence> </xs:complexType>
</xs:element>
<xs:element type="xs:string" name="E-
Mail"/>
<xs:element type="xs:long"
name="Mobile_Number"/> </xs:sequence>
</xs:complexType> </xs:element>
</xs:schema>
```

## academiccal_output.html

**Code:**

```
<html> <head> <META http-equiv="Content-
Type" content="text/html; charset=UTF-8">
<title>18BCE0741</title>
<style>
table, th, td {
border: 2px solid black; border-collapse:
collapse; padding:1%;
}
tr:hover {background-color: #99FFFF;} tr:nth-
child(even) {background-color: #66B2FF;}
</style> </head>
<body style="background-color:#CCE5FF">
<center>
<h1>Devina's Details</h1> <div>
<table border="1"> <tr>
<td> <p>
<b>REGISTRATION NUMBER</b> </p>
</td><td> <p>18BCE0741</p>
</td> </tr>
<tr> <td>
<p> <b>NAME</b>
</p> </td><td>
<p>DevinaV arshney</p> </td>
</tr> <tr>
<td> <p>
<b>BRANCH</b> </p>
</td><td>
<p>B.Tech in Computer Science and
Engineering</p>
</td> </tr>
<tr> <td> <p>
<b>YEAR OF ADMISSION</b>
</p> </td><td>
<p>2018</p> </td>
</tr> <tr>
<td> <p>
<b>CGPA</b>
</p> </td><td>
<p>9.60</p> </td>
</tr> <tr>
<td> <p>
<b>INTERESTS</b> </p>
</td><td>
<p>Web Development,
Machine Learning,
Dancing</p> </td>
</tr> <tr>
<td> <p>
<b>EVENTS</b> </p>
</td><td> <p>Gravitas'19,
</td> </tr>
<tr> <td>
HackGrid'19,
iOS Fusion'19</p>
<p> <b>ADDRESS</b>
</p> </td><td>
<p>C301 Sangath Platina, Motera,
Ahmedabad,
Gujarat</p> </td>
</tr> <tr>
<td> <p> <b>E-MAIL</b>
</p> </td><td>
<p>devinavarshney@gmail.com</p> </td>
</tr> <tr> <td> <p>
<b>PHONE</b> </p> </td><td>
<p>9662035700</p> </td> </tr> </table>
</div> </center> </body>
</html>
```

# NODE JS

Node.js runs single-threaded,non-blocking,asynchronously programming,which is very memory efficient.

Here is how PHP or ASP handles a file request:
•Sends the task to the computer's file system.
•Waits while the file system opens and reads the file.
•Returns the content to the client.
•Ready to handle the next request.
Here is how Node.js handles a file request:
•Sends the task to the computer's file system.
•Ready to handle the next request.
•When the file system has opened and read the file, the server returns the content to the client.

Module - Description
assert - Provides a set of assertion tests
buffer - To handle binary data
child_process - To run a child process
cluster - To split a single Node process into multiple processes
crypto - To handle OpenSSL cryptographic functions
dgram - Provides implementation of UDP datagram sockets
os - Provides information about the operation system
dns - To do DNS lookups and name resolution functions
domain - Deprecated. To handle unhandled errors
events - To handle events
fs - To handle the file system
http - To make Node.js act as an HTTP server
https - To make Node.js act as an HTTPS server.
net - To create servers and clients

emitter.addListener(event, listener)
Adds a listener to the end of the listeners array for the specified event. No checks are made to see if the listener has already been added.
emitter.on(event, listener)
Adds a listener to the end of the listeners array for the specified event. No checks are made to see if the listener has already been added. It can also be called as an alias of emitter.addListener()
emitter.once(event, listener)
Adds a one time listener for the event. This listener is invoked only the next time the event is fired, after which it is removed.
emitter.removeListener(event, listener)
Removes a listener from the listener array for the specified event. Caution: changes array indices in the listener array behind the listener.
emitter.removeAllListeners([event])
Removes all listeners, or those of the specified event.
emitter.setMaxListeners(n)
By default EventEmitters will print a warning if more than 10 listeners are added for a particular event.
emitter.getMaxListeners()
Returns the current maximum listener value for the emitter which is either set by emitter.setMaxListeners(n) or defaults to EventEmitter.defaultMaxListeners.
emitter.listeners(event)
Returns a copy of the array of listeners for the specified event.
emitter.emit(event[, arg1][, arg2][, ...])

Raise the specified events with the supplied arguments.
emitter.listenerCount(type)
Returns the number of listeners listening to the type of event

**Events in Node.js**

```
var fs = require('fs');
var rs = fs.createReadStream('./demofile.txt');
rs.on('open', function () {
  console.log('The file is open');
});
```

In the example below we have created a function that will be executed when a "scream" event is fired.

To fire an event, use the emit() method.

```
var events = require('events');
var eventEmitter = new events.EventEmitter();
//Create an event handler:
var myEventHandler = function () {
  console.log('I hear a scream!');
}
//Assign the event handler to an event:
eventEmitter.on('scream', myEventHandler);
//Fire the 'scream' event:
eventEmitter.emit('scream');
```

**Listeners**

emitter.addListener(event, listener)

```
var emitter = require('events').EventEmitter;
var em = new emitter();
//Subscribe FirstEvent
em.addListener('FirstEvent', function (data) {
    console.log('First subscriber: ' + data);
});
// Raising FirstEvent
em.emit('FirstEvent', 'This is my first Node.js event emitter example.');
```

**What are callbacks?**

A callback function is a function that is passed to another function as a parameter, and the callback function is called (executed) inside the second function.

Callbacks functions execute asynchronously

```
var fs = require('fs');
var fcontent ;
function readingfile(callback){
fs.readFile("readme.txt", "utf8", function(err, content) {
fcontent=content;
 if (err)
 {
 return console.error(err.stack);
  }
callback(content);
})
};
function mycontent() {
  console.log(fcontent);
}
readingfile(mycontent);
Console.log('Reading files....');
```

**mycontent() function can get passed in an argument that will become the callback variable inside the readingfile() function. After file reading is completed through readFile() the callback variable ( callback()) will be invocked. Only function can be invoked**

**Querying for data in a MongoDB database**

```
varMongoClient=
require('mongodb').MongoClient;
varurl= 'mongodb://localhost/EmployeeDB';
MongoClient.connect(url, function(err, db) {
varcursor = db.collection('Employee').find();
cursor.each(function(err, doc) {
console.log(doc);
});
});
```

**Inserting documents in a collection**

```
varMongoClient=
require('mongodb').MongoClient;
varurl= 'mongodb://localhost/EmployeeDB';
MongoClient.connect(url, function(err, db) {
db.collection('Employee').insertOne({
Employeeid: 4,
EmployeeName: "NewEmployee"
});
});
```

To check that the data has been properly inserted in the database, execute the following commands in MongoDB

- Use EmployeeDB
- db.Employee.find({Employeeid:4 })

**Updating documents in a collection**

```
varMongoClient=
require('mongodb').MongoClient;
varurl= 'mongodb://localhost/EmployeeDB';
MongoClient.connect(url, function(err, db) {
db.collection('Employee').updateOne({
"EmployeeName": "NewEmployee"
}, {
$set: {
"EmployeeName": "Mohan"
}
});
});
```

**Deleting documents in a collection**

```
varMongoClient=
require('mongodb').MongoClient;
varurl= 'mongodb://localhost/EmployeeDB';
MongoClient.connect(url, function(err, db) {
db.collection('Employee').deleteOne(
{
"EmployeeName": "Mohan"
}
);
});
```