Unit 3

Web Development II

1. Web Development

Web development is a specific field of software engineering that helps to build or design webpages. The major building blocks of web are HTML, CSS and JavaScript. The web development can be divided into two main categories i.e. front-end development and back-end development. Everything we see in web pages is considered as front-end. Front development can be done by using three codes i.e. HTML, CSS and JavaScript. Back-end development can be done by PHP, SQl etc.

2. HTML

HTML stands for Hyper Text Markup Language. It is a standard language that is used to create a webpage. It describes the structure of a webpage. It turns text into webpage and allows to put content like text, images, videos, audios, link etc. in a webpage.

3. CSS

CSS stands for Cascading style sheet. It is a language that is used to change the looks and appearance of web page. It describes how HTML elements are to be displayed on screen.

4. Client-Side Scripting

It is a code that runs on client computer in the Web Browser. The are used to create web pages and allows user to interact with the webpages through Buttons forms etc. They are completely browser dependent. Examples HTML, JavaScript, VB Script.

5. Server-side scripting

It is a code that runs on the web server. When any data is requested through client computer, it received by a server through these codes. It is used to connect the database that is on the web server. They are basically used to create dynamic pages. Examples PHP, JSP, ASP, Ruby.

6. JavaScript

JavaScript is a programming language also called client side scripting language that is specially used in websites to make any webpage interactive and attractive. This language is specially used to make any webpage dynamic. It was developed by LiveScript by Netscape in 1990 AD later renamed as JavaScript in 1995

7. understanding JavaScript Syntax

A JavaScript consist of JavaScript statements that are written within the <script> </script> HTML tags in the webpage or in external JavaScript file having .js extension. JavaScript codes consists of statements and semicolons.

Examples1: to display information in Brower's Window JavaScript:

```
<html>
<head><title>javascript</title></head>
<body>
<script>
	document.write("Hello JavaScript");
</script>
</body>
</html>
```

Output: Hello JavaScript

document.write( ); = is used to display information in webpage in JavaScript.

Examples2: to display information in JavaScript using alert box:

```
<html>
<head><title>javascript</title></head>
<body>
<script>
	alert("Hello JavaScript");
</script>
</body>
</html>
```

Output: Hello JavaScript

alert( ); = is used to display information in alertbox in JavaScript.

Examples3: to display information in browser's console:

```
<html>
<head><title>javascript</title></head>
<body>
<script>
	console.log("Hello JavaScript");
</script>
</body>
</html>
```

Output: Hello JavaScript

Console.log( ); = is used to display in Brower's console which shows detailed output (You can also find out error while JavaScript.

Examples4: to display information using getElementById() with the help of innerHTML property.

```
<html>
<head><title>javascript</title></head>
<body>
	<p id="call"></p>
<script>
	document.getElementById("call").innerHTML ="Hello JavaScript";
</script>
</body>
</html>
```

Output: Hello JavaScript

The getElementById() method returns the elements that have given an ID and .innerHTML is used to display the value.

8. Keywords in in JavaScript

Keywords are reserved words in JavaScript, which cannot be used as a variab le names or function names. Some Keywords are : *var, function, if, else, do, while, for, switch, break, default, return, case, class, void, continue, try, finally, debugger, this, true, false,in, instanceOf, typeOf, new, null, throw, width, delete.*

9. Assigning Data Types in JavaScript

```
<html>
<head><title>javascript</title></head>
<body>
<script>
var a = 5;	// var is a data type and a is a variable which holding numeric value 5
let b = 6;	// let is a data type and a is a variable which holding numeric value 6
var sum = a+b;
var h = " I am holding string";	// var is a data type and h is a variable which holding string.
document.write(sum);		// sum is a variable which is holding the value of a and b
document.write("<Br>");		// for breaking line
document.write("The sum is" + sum);		// “The sum is”  is a string
document.write("<Br>");		// for breaking line
document.write(h);
</script>
</body>
</html>
```

Output:	11
The sum is 11
I am holding string

10. Operators

Operators are symbols or keywords that tells java script perform task.

Arithmetic Operators: The operators that perform addition, subtraction, multiplication and division.

+ (Addition) - (Subraction) * (Multiplication) / (Division) % (Modules)

Assignment Operators: The operators that used to assign value to the variable.

=, +=, -=, *=, /=, %=

Increment and decrement Operators: It is used to increase or decrease the value

i++, i—

Logical Operators: It is used to combine conditional statement.

&& (And) || (Or) ! (Not)

Relational Operators / Conditional Operators: It is used to compare two values.

== (Equal to) != (Not equal to) < (Less than) > (Greater than) >= (Greater than or equal to), <= (less than or equal to).

11. Control Structure

Control structure controls the flow of execution of a program. Looping and selection structure are some control structures used in JavaScript.

a. Selection Structure : The selection statements are executed on the basis of the given conditions either true or false
   i. If …. else statement

      Syntax:

      ```
      If(condition)
      {
      Statement ;
      }
      ```

Example 5 :

```
<script>
    var a=40;
    if(a>=40)
    {
        document.write("you are old");
    }
</script>
```

Output: you are old

Example 6 :

```
<script>
var a=30;
if(a>=40)
{
        document.write("you are old");
}
else
{
        document.write("you are young");
}
</script>
```

Output: you are young

Example 7 :

```
<script>
var a=15;
if(a>=40)
{
        document.write("you are old");
}
else if(a>=20)
{
        document.write("you are young");
}
else
{
        document.write("you are teenaged");
}
</script>
```

Output: you are teenaged

ii. switch: The switch statements are executed on the basis of the given conditions either true or false. It works as if statement.

Syntax:

```
switch(condition)
{
case value 1:
statement;
```

```
break;
case value2:
statement2;
break;
……..
…….
default:
statement..n;
}
```

Example 8 :

```
<script>
    var a=1
    var x=50;
    var y=30;
    switch(a)
    {
case 1:
var sum = x+y;
document.write("The sum is " + sum);
break;
case 2:
var diff= x-y;
document.write("The difference is " + diff);
break;
default:
document.write("Error.. ");
    }
    </script>
```

Output: The sum is 80 (if case is 1)

Example 9 :

```
<script>
    var a="kathmandu"
          switch(a)
    {
case "palpa":
document.write("You are going to " + a);
break;
case "kathmandu":
document.write("You are going to " + a);
```

```
        break;
        default:
        document.write("Error.. ");
            }
            </script>
```

Output: You are going to Kathmandu (if your case is Kathmandu)

b. Looping Structure: It repeats the block of statement until certain condition is met.

i. For loop

Syntax:

```
for(initial_value; condition; increment/decrement)
{
Statement;
    }
```

Example 10:

```
    <body>
    <script>
    for (var i=1; i<=3; i++)
    {
        document.write("Hello World" + "<br>");
    }
    </script>
</body>
```

Output:

Hello World
Hello World
Hello World

Example 11:

```
<body>
    <script>
    for (var i=1; i<=3; i++)
    {
        document.write(i + "<br>");
    }
    </script>
</body>
```

Output:

1
2
3

ii. while loop : While loop repeats the block of statement. It is entry control loop it means the condition is checked initially and then block of code is executed.

Syntax:

```
initial_value ;
while (condition)
{
Statement;
Increment/decrement;
}
```

Example 12:

```
<body>
    <script>
        var a=1;
    while(a<=3)
    {
        document.write("javascript" + "<br>");
        a++;
    }
    </script>
</body>
```

Output:

javascript
javascript
javascript

iii. do while loop: do while loop repeats the block of statement. It execute the block of ststement once and then the condition is evaluated

syntax:

```
do
{
Statement;
Increment/decrement
}
While(condition);
```

Example 13:

```
<body>
    <script>
        var a=1;
```

```
        do
        {
                document.write("javascript" + "<br>");
                a++;
        }
        while(a<=3);
        </script>
</body>Output:
javascript
javascript
javascript
```

12. Functions

Function is a group of statements that perform specific task. It helps to divide huge programs into small functional task and makes code much easier to maintain.

a. Defining and calling function

Syntax:

```
Function functionName()
{
Codes to be excecuted;
}
Calling function;
```

Example 13:

```
<body>
        <script>
                        function fun()
        {

                document.write("hello");
        }
        fun();
        </script>
</body>
```

Output:hello

Or,

```
<body>
        <script>
```

```
            fun();
            function fun()
            {
                document.write("hello");
            }
        </script>
</body>
```

Output:hello

b. Adding parameters to functions

Example 14: With example to add two numbers using function.

```
<body>
        <script>
            var a=5;
            var b=6;
            sum(a,b);
                function sum(a,b)
                {
                var c= a+b;
                document.write(c);
                }
            </script>
</body>
```

c. Returning Values from function

Example 15: With example to add two numbers using function returning value.

```
<body>
<script>
            var a=5;
            var b=6;
            sum(a,b);
                function sum(a,b)
                {
                var c= a+b;
                return c;
                }
                document.write(sum(a,b));
            </script>
</body>
```

Note: Local and lobal variables:

If you define variable inside a function then it becomes local variable and can be accessed in that function only But, if you declare or define variable outside of the function it becomes global variable and can be accessed in any function.

Example 16: Example of Global Variable

```
<body>
	<script>
		var a=5;
		var b=6;
		sum(a,b);
		diff(a,b);
			function sum(a,b)
			{
			var c= a+b;
			document.write(c);
			}

			function diff(a,b)
			{
			var c= a-b;
			document.write(c);
			}
		</script>
</body>
```

13. Event Handling with JavaScript: Event handling is a manipulating webpage in any manner. Loading pages, clicking buttons, closing windows, resizing windows are some examples of even handling.

Example 16: Clicking button and display "Hello World"

```
<html>
<head><title>myjavascript</title>
</head>
<body>

<button type="button" onclick="document.write('Its me handling even with button')"
	>click</button>
</body>
</html>
```