1. <button onclick=’document.getElementById(“text-1”).innerHTML=”Hello JS”’>Click Me! </button>

2. <img id=”img-1” src=”pic\_bulb\_off.jpg” style=”width:100px”>

<button onclick=”document.getElementById(“img-1”).src=’pic\_bulb\_on.jpg’”>Turn on the light</button>

<button onclick=”document.getElementById(“img-1”).src=’pic\_bulb\_off.jpg’”>Turn OFF the light</button>

3. document.getElementById(“demo”).style.fontSize=”35px”;

document.getElementById(“demo”).style.display=”none”;

document.getElementById(“demo”).style.display=”block”;

4. <script>

function click\_1() {

document.getElementById(“demo”).innerHTML=”paragraph changed”;

}

</script>

<button onclick=”click\_1()”> Try it </button>

5. <p id=”demo” > This paragraph will be changed </p>

<script>

document.getElementById(“demo”).innerHTML=5+6;

</script>

6. <button onclick=”document.write(5+6)”>Try it</button>; // it will delete all the HTML content and simply display “11”

7. window.alert(5+6); OR alert(5+6);

8. console.log(5+6);

9. window.print() // search about it –

10. document.write("<h1>This is a heading</h1>");

document.write("<p>This is a paragraph.</p><br>");

document.write("<p>This is another paragraph.</p>");

11. function show\_alert() {

alert("Hello! I am an alert box!");

}

<input type="button" onclick="show\_alert()" value="Show alert box" />

12. function show\_confirm() {

var r=confirm("Press a button!");

if (r==true) {

alert("You pressed OK!");

} else {

alert("You pressed Cancel!"); }

}

<input type="button" onclick="show\_confirm()" value="Show a confirm box" />

13. function disp\_prompt() {

var fname=prompt("Please enter your name:","Your name")

document.getElementById("msg").innerHTML="Greetings, " + fname

}

<input type="button" onclick="disp\_prompt()" value="Display a prompt box" /> <br /><br />

<div id="msg"></div>

Chapter – 10 :-

10 A. - function functionname(var1,var2,...,varX) {

some code

}

10 B. - function product(a,b) {

return a\*b;

}

document.write(product(4,3));

Chapter – 11 :-

11. A. - ❑ onLoad and onUnload

❑ onFocus, onBlur, and onChange

❑ onSubmit

❑ onMouseOver and onMouseOut

❑ onClick

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

II. <input type="button" name="language" value="Click for Spanish" onclick="translate()">

Chapter – 12 :-

Try and catch & Throw statement

A. - function message() {

try {

adddlert("Welcome guest!");

} catch(err)

{

txt="There was an error on this page.\n\n";

txt+="Click OK to continue viewing this page,\n";

txt+="or Cancel to return to the home page.\n\n"; if(!confirm(txt))

{ document.location.href="http://www.w3schools.com/";

}

} }

<input type="button" value="View message" onclick="message()" />

B. try {

if(x>10)

{ throw "Err1";

} else if(x<0)   
 { throw "Err2";

} else if(isNaN(x))

{ throw "Err3";

} }

catch(er) {

if(er=="Err1") {

alert("Error! The value is too high"); }

if(er=="Err2") {

alert("Error! The value is too low");

}

if(er=="Err3") {

alert("Error! The value is not a number");

} }

</script>

Chapter – 13 :-

Nothing Special

Chapter 14 :-

Nothing Special

Chapter 15. –

var txt = "Hello World!";

document.write(txt.length);

document.write("<p>Big: " + txt.big() + "</p>"); document.write("<p>Small: " + txt.small() + "</p>");

document.write("<p>Bold: " + txt.bold() + "</p>"); document.write("<p>Italic: " + txt.italics() + "</p>");

document.write("<p>Fixed: " + txt.fixed() + "</p>"); document.write("<p>Strike: " + txt.strike() + "</p>");

document.write("<p>Fontcolor: " + txt.fontcolor("green") + "</p>"); document.write("<p>Fontsize: " + txt.fontsize(6) + "</p>");

document.write("<p>Subscript: " + txt.sub() + "</p>"); document.write("<p>Superscript: " + txt.sup() + "</p>");

document.write("<p>Link: " + txt.link("http://www.w3schools. com") + "</p>");

document.write("<p>Blink: " + txt.blink() + " (does not work in IE, Chrome, or Safari)</p>");

document.write(txt.toLowerCase() + "<br />");

document.write(txt.toUpperCase());

document.write(str.match("world") + "<br />");

document.write(str.replace("Microsoft","w3schools"));

document.write(str.indexOf("Hello") + "<br />");

Chapter – 16 :-

new Date() // current date and time

new Date(milliseconds) //milliseconds since 1970/01/01

new Date(dateString)

new Date(year, month, day, hours, minutes, seconds, milliseconds)

I. var myDate=new Date();

myDate.setFullYear(2010,0,14);

And in the following example, we set a Date object to be five days into the future:

var myDate=new Date();

myDate.setDate(myDate.getDate()+5);

II. var d=new Date();

document.write(“Original: “);

document.write(d + “<br />”);

document.write(“To string (universal time): “);

document.write(d.toUTCString());

// Original: Wed Mar 10 2010 23:07:53 GMT-0500 (Eastern Standard Time)

To string (universal time): Thu, 11 Mar 2010 04:07:53 GMT

III. Clock on the web page :-

function startTime() {

var today=new Date();

var h=today.getHours();

var m=today.getMinutes();

var s=today.getSeconds();

// add a zero in front of numbers<10

m=checkTime(m); s=checkTime(s);

document.getElementById(‘txt’).innerHTML=h+”:”+m+”:”+s;

t=setTimeout(‘startTime()’,500);

}

Chapter – 17 :-

Array :-

1. var myCars=new Array();

myCars[0]="Saab";

myCars[1]="Volvo";

myCars[2]="BMW";

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

3. var myCars=["Saab","Volvo","BMW"];

for (i=0;i<mycars.length;i++) {

document.write(mycars[i] + "<br />");

}

I. var parents = ["Jani", "Tove"];

var children = ["Cecilie", "Lone"];

var family = parents.concat(children);

II. var fruits = ["Banana", "Orange", "Apple", "Mango"];

document.write(fruits.join("+"));

III. document.write(fruits.pop() + "<br />");

document.write(fruits + "<br />");

document.write(fruits.push("Kiwi") + "<br />");

document.write(fruits.push("Lemon","Pineapple") + "<br />");

document.write(fruits.reverse());

document.write(fruits.shift() + "<br />");

document.write(fruits + "<br />");

document.write(fruits.slice(0,1) + "<br />");

document.write(fruits.slice(1) + "<br />");

document.write(fruits.slice(-2) + "<br />");

document.write(fruits.sort());

document.write("Removed: " + fruits.splice(2,0,"Lemon") + "<br />"); // The following example demonstrates how to use splice() to add an element to the second position in an array

document.write(fruits.toString()); // converts an array to a string.

document.write(fruits.unshift("Kiwi") + "<br />"); // The following example shows you how to add new elements to the beginning of an array

document.write(fruits.unshift("Lemon","Pineapple") + "<br />");

III. var n = ["10", "5", "40", "25", "100", "1"];

document.write(n.sort(sortNumber));

Chapter – 18 :-

Boolean :-

I. var b1=new Boolean(0); // 0 ,1 ,””,null, NaN , false

document.write(“0 is boolean “+ b1 +”<br />”); // 0 is Boolean false

Chapter -19 :-

Math object :-

I. var pi\_value=Math.PI;

var sqrt\_value=Math.sqrt(16);

II. Constants:-

Math.E

Math.PI

Math.SQRT2

Math.SQRT1\_2

Math.LN2

Math.LN10

Math.LOG2E

Math.LOG10E

III. document.write(Math.round(4.7)); // 5

The following examples uses the random() method of the Math object to return a random number between 0 and 1 :

document.write(Math.random());

document.write(Math.floor(Math.random()\*11)); // Between 0 and 10

document.write(Math.max(0,150,30,20,38)); // 150

document.write(Math.min(0,150,30,20,38) + “<br />”); // 0

Chapter – 20 :-

RegExp Object :-

I. The syntax is as follows:

var txt=new RegExp(pattern,modifiers);

or more simply:

var txt=/pattern/modifiers;

Regexp Modifiers

A. Modifiers are used to perform case-insensitive and global searches. The I modifier is used to perform case-insensitive matching. The g modifier is used to perform a global match (find all matches rather than stopping after the first match). The following example demonstrates how to do a case-insensitive search for “w3schools” in a string:

var str="Visit W3Schools";

var patt1=/w3schools/i;

<script type="text/javascript">

var str = "Visit W3Schools";

var patt1 = /w3schools/i;

document.write(str.match(patt1));

</script>

B. The following example demonstrates how to do a global search for “is”:

<script type="text/javascript">

var str="Is this all there is?";

var patt1=/is/g; document.write(str.match(patt1));

</script>

C. Global , case-insensitive search for “is” :

<script type="text/javascript">

var str="Is this all there is?";

var patt1=/is/gi;

document.write(str.match(patt1));

</script>

D. The following example searches a string for the character “e”:

var patt1=new RegExp("e");

document.write(patt1.test("The best things in life are free"));

Because there is an “e” in the string, the output of the preceding code is as follows:

true

<script type="text/javascript">

var patt1=new RegExp("e");

document.write(patt1.test("The best things in life are free"));

</script>

E. var patt1=new RegExp("e");

document.write(patt1.exec("The best things in life are free"));

Because there is an “e” in the string, the output of the preceding code is:

e

SECTION – III

JS ADVANCED

Chapter -21 JS Brower detection

A. The navigator Object

appName—holds the name of the browser

appVersion—holds, among other things, the version of the browser

var browser=navigator.appName;

var b\_version=navigator.appVersion;

function detectBrowser() {

var browser=navigator.appName;

var b\_version=navigator.appVersion;

var version=parseFloat(b\_version);

if ((browser=="Netscape"||browser=="Microsoft Internet Explorer") && (version>=4)) {

alert("Your browser is good enough!");

} else {

alert("It's time to upgrade your browser!");

}

}

<body onload="detectBrowser()">

A-I. document.write("<p>Browser: ");

document.write(navigator.appName + "</p>");

document.write("<p>Browserversion: ");

document.write(navigator.appVersion + "</p>");

document.write("<p>Code: ");

document.write(navigator.appCodeName + "</p>");

document.write("<p>Platform: ");

document.write(navigator.platform + "</p>");

document.write("<p>Cookies enabled: ");

document.write(navigator.cookieEnabled + "</p>");

document.write("<p>Browser's user agent header: ");

document.write(navigator.userAgent + "</p>");

A-II. var x = navigator;

document.write("CodeName=" + x.appCodeName);

document.write("MinorVersion=" + x.appMinorVersion);

document.write("<br />");

document.write("Name=" + x.appName);

document.write("Version=" + x.appVersion);

document.write("<br />");

document.write("CookieEnabled=" + x.cookieEnabled);

document.write("<br />");

document.write("CPUClass=" + x.cpuClass);

document.write("<br />");

document.write("OnLine=" + x.onLine);

document.write("<br />");

document.write("Platform=" + x.platform);

document.write("<br />");

document.write("UA=" + x.userAgent);

document.write("<br />");

document.write("BrowserLanguage=" + x.browserLanguage);

document.write("<br />");

document.write("SystemLanguage=" + x.systemLanguage);

document.write("<br />");

document.write("UserLanguage=" + x.userLanguage);

Chapter – 22

JS Cookies :-

What is a Cookie?

A cookie is a variable that is stored on the visitor’s computer. Each time the same computer requests a page with a browser, it sends the cookie, too. With JavaScript, you can both create and retrieve cookie values. Examples of cookies:

**I. Name cookie.** The first time a visitor arrives on your Web page, she must fill in her name. The name then is stored in a cookie. Next time the visitor arrives at your page, she could get a welcome message like “Welcome Jane Doe!” The name is retrieved from the stored cookie.

First , we create a function that stores the name of the visitor in a cookie variable:

function setCookie(c\_name,value,expiredays) {

var exdate=new Date();

exdate.setDate(exdate.getDate()+expiredays);

document.cookie=c\_name+ "=" +escape(value)+ ((expiredays==null) ? "" : ";expires="+exdate.toGMTString());

}

Then , we create another function that checks whether the cookie has been set :

function getCookie(c\_name) {

if (document.cookie.length>0) {

c\_start=document.cookie.indexOf(c\_name + "=");

if (c\_start!=-1) {

c\_start=c\_start + c\_name.length+1;

c\_end=document.cookie.indexOf(";",c\_start);

if (c\_end==-1) c\_end=document.cookie.length;

return unescape(document.cookie.substring(c\_start,c\_ end));

}

}

return "";

}

Last, we create the function that displays a welcome message if the cookie is set, and if the cookie is not set, it displays a prompt box asking for the name of the user:

function checkCookie() {

username=getCookie('username');

if (username!=null && username!="") {

alert('Welcome again '+username+'!');

} else {

username=prompt('Please enter your name:',"");

if (username!=null && username!="") {

setCookie('username',username,365);

}

}

}

**II. Password cookie.** The first time a visitor arrives on your Web page, she must fill in a password. The password then is stored in a cookie. Next time the visitor arrives at your page, the password is retrieved from the cookie.

**III. Date cookie.** The first time a visitor arrives to your Web page, the current date is stored in a cookie. Next time the visitor arrives at your page, she could get a message like “Your last visit was on Tuesday, August 11, 2005!” The date is retrieved from the stored cookie.

**Program:-**

<html>

<head>

<script type="text/javascript">

function getCookie(c\_name) {

if (document.cookie.length>0) {

c\_start=document.cookie.indexOf(c\_name + "=");

if (c\_start!=-1) {

c\_start=c\_start + c\_name.length+1 ;

c\_end=document.cookie.indexOf(";",c\_start);

if (c\_end==-1) c\_end=document.cookie.length

return unescape(document.cookie.substring(c\_start,c\_ end));

}

}

return ""

}

function setCookie(c\_name,value,expiredays) {

var exdate=new Date();

exdate.setDate(exdate.getDate()+expiredays);

document.cookie=c\_name+ "=" +escape(value)+((expiredays= =null) ? "" : ";

expires="+exdate.toGMTString());

}

function checkCookie() {

username=getCookie('username');

if (username!=null && username!="") {

alert('Welcome again '+username+'!');

} else {

username=prompt('Please enter your name:',"");

if (username!=null && username!="") {

setCookie('username',username,365);

}

}

}

</script>

</head>

<body onLoad="checkCookie()">

</body>

</html>

Chapter - 23 :-

JS Form Validation :-

<html>

<head>

<script type=”text/javascript”>

function validate\_required(field,alerttxt)

{ with (field) { if (value==null||value==””) {

alert(alerttxt);return false;

} else {

return true; } } }

function validate\_form(thisform) {

with (thisform) {

if (validate\_required(email,”Email must be filled out!”)==false) {

email.focus();return false;}

} }

</script>

</head>

<body>

<form action=”submit.htm” onsubmit=”return validate\_ form(this)” method=”post”>

Email: <input type=”text” name=”email” size=”30”> <input type=”submit” value=”Submit”>

</form>

</body>

</html>

E-mail Validation :-

<html>

<head>

<script type=”text/javascript”>

function validate\_email(field,alerttxt) {

with (field) {

apos=value.indexOf(“@”);

dotpos=value.lastIndexOf(“.”);

if (apos<1||dotpos-apos<2) {

alert(alerttxt);return false;

} else {

return true; } } }

function validate\_form(thisform) {

with (thisform) {

if (validate\_email(email,”Not a valid e-mail address!”)==false) {

email.focus();return false;} } }

</script>

</head>

<body>

<form action=”submit.htm” onsubmit=”return validate\_ form(this);” method=get”>

Email: <input type=”text” name=”email” size=”30”> <input type=”submit” value=”Submit”>  
 </form>

</body>

</html>

Chapter – 24 JS animation :-

The changing between the images is done with the following JS :-

<script type="text/javascript">

function mouseOver() {

document.getElementById("b1").src ="b\_blue.gif";

}

function mouseOut() {

document.getElementById("b1").src ="b\_pink.gif";

}

</script>

The function mouseOver() causes the image to shift to “b\_blue.gif”

The function mouseOut() causes the image to shift to “b\_pink.gif”

Chapter – 25 :-

JS Image maps

// target=”\_blank” /.- opens the page on next web page

<html>

<body>

<img src ="planets.gif" width ="145" height ="126" alt="Planets" usemap="#planetmap" />

<map name="planetmap">

<area shape ="rect" coords ="0,0,82,126" href ="sun.htm" target ="\_blank" alt="Sun" />

<area shape ="circle" coords ="90,58,3"

href ="mercur.htm" target ="\_blank" alt="Mercury" />

<area shape ="circle" coords ="124,58,8" href ="venus.htm" target ="\_blank" alt="Venus" />

</map>

</body>

</html>

Chapter - 26 :-

JS timing events :-

❑ The setTimeout() Method

❑ The clearTimeout() Method

With JS , it is possible to execute some code after a specified time interval.

This is called timing events.

setTimeout()—Executes a code some time in the future

clearTimeout()—Cancels the setTimeout()

function timedMsg() {

var t=setTimeout("alert('I am displayed after 3 seconds!')",3000);

}

<form>

<input type="button" value="Display alert box!" onClick="timedMsg()" />

</form>

Chapter – 27 :- Create your own Objects with JS :-

<html>

<body>

<script type="text/javascript">

personObj=new Object();

personObj.firstname="John";

personObj.age=50;

document.write(personObj.firstname + " is " + personObj.age + " years old."); </script>

</body>

</html>

// John is 50 years old.

OR

<script type="text/javascript">

function person(firstname,lastname,age,eyecolor)

{

this.firstname=firstname;

this.lastname=lastname;

this.age=age;

this.eyecolor=eyecolor; }

myFather=new person("John","Doe",50,"blue");

document.write(myFather.firstname + " is " + myFather.age + " years old."); </script>

// John is 50 years old.

SECTION – IV

AJAX BASIC

Chapter – 28:-

Ajax XMLHttpRequest