

(5) Harmony (प्रत्यक्षता) :- फ़िरी ओं पैगेज के harmony display करने के लिए निम्न जानकारी है -

अभी कुछ points जो display करने के लिए जानकारी हैं -

- (i) Equal font & colors को use.
- (ii) ~~graphics use~~ जो लिंग और विद्युत वाली subject के relationship रखने के / ग्राफिक्स की quality equal हो।
- (iii) ग्राफिक्स की quality equal हो।
- (iv) ~~प्रत्यक्ष - 2 page~~ जो graphics & text के समन्वय हो।

⇒ History of Java Script :

Scripting language को use website के advance features के odd करने के लिए लिया जाता है। यह जी एक स्क्रिप्ट भी programming language है। यह नो web page को जो कहता है। यहकि उसमें किसी condition की check करना, loop की चलाना etc. work कर सकते हैं। यहकि प्रिय तरीं में HTML के scripting language की help लिनी चाही है। calculate करना है [यदि salary 10,000 situation जो HTML के साथ हो सकता है] तब यह scripting language को use करना है।

इसके बारे में - Scripting language के two types

(i) Client Side Scripting language :

एक language जो write किये जवे programme user की machine पर browser से run होते हैं।

(ii) Server Side Scripting language :

इसका use database से related work के लिए किया जाता है। अप्स - user को login name, password right हो या नहीं लिया info. database को store करना।

History :

Java script object oriented maximum use client side scripting तो किया जाता है। Java script को programmes को run करने के लिए selected browser को script को support करने वाला होता है।

Java script को 1985 में

Net Scape Company ने develop किया था। & उसके बाद Net Scape Navigator browser को भी launch किया गया।

उसी तिथि वर्ष 1995 Net Scape company ने Sun Microsystems को साथ marketing deal की थी, तो उसका name डिवाइल "Java Script" हो गया।

→ features of Java Script :

① HTML को साथ use करने से web page effective बनता है।

② Java Script में programming करना easy होता है।

③ Java Script एक interpreted language है।

Java Script का case sensitive language तो tot different variable होते हैं। Java Script को use करने से web page का अलग performance करना होता है।

④ उसमें interactive game बनाया जा सकते हैं।

⑤ उसमें hero game को function में divide किया जाता है। जिसके programming easy हो जाती है।

⑥ All most यह object oriented language है।

(1) User द्वारा input को validate करना। maximum character को check करना।

(2) User की machine पर cookies set करना।

(3) Web page के किसी element को hide/display करना।

⇒ Differences of Java Script :

- (1) Other programming language की तरह java script की सकला है, जो database handling की फ़िल्हा भी देता है।
- (2) इसकी programmes की client की machine पर file को read या write नहीं करता है।
- (3) यह किसी window की java script की help नहीं नहीं कर सकते।

⇒ Difference between Java & Java script :

Java Java Script

(1) Java language compiler & interpreted language है।

(2) एक strongly typed language है। इसकी language की किसी variable के according to variable की different time data expose होता जाता है तर मूलग का value को store करता है।

(3) इसमें data type की no. अधिक होती है।

(4) इस language की use web app. बनाने के लागू-2 देखा जाता है।

⇒ Programme format :

```

HTML के आप java script की coding के लिए
script language का use किया जाता है। जो उस
script tag के साथ की attribute language
script tag का use किया जाता है।
script tag का use करने की head section (body
section वाली) के बारे में है।
```

Subject _____

Date: / /
MON TUE WED THU FRI SAT SUN

Subject _____

Date: / /
MON TUE WED THU FRI SAT SUN

</script>
</body>
</html>

Java script का programme द्वारा window के text editor Netopad में type करके & फिर save करके browser में open करके file का .html नाम का Java बनाया जाएगा।

⇒ First programme :

```
<html>
<head>
<title> First programme </title>
</head>
<body>
<script type = "text/javascript">
alert ("Hello");
</script>
</body>
</html>
```

Output

Microsoft Internet Explorer



Hello

OK

document.write('document object द्वारा')
⇒ Comments :
⇒ for HTML document को display करना है
& write object की method का, जिसका message को display करता है।

- (i) Single line comment
- (ii) Multi line comment

```
<html>
<head>
<title> First program </title>
</head>
<body>
<script type = "text/javascript">
// document.write ("Hello");
/* document.write ("Hello");
document.write ("Hello"); */
</script>
</body>
</html>
```

⇒ Type conversion

Java script में variable को declare करे तो time
लिए type वाली की need होती है। eg.-
use करने सकते हैं & कुछ समय तो है

value जी store करना सकते हैं।

```
<html>
<head>
<script type = "text/javascript">
var x = "Total is ";
var z = "10" + 40;
document.write(x + "<br>");
document.write(y + "<br>");
document.write(z);
</script>
</head>
</body>
</html>
```

Output
Total is 40
40 is the total
1040

⇒ How to get input from user :

Java script में user का input को कैसे

document.write() का use करता है।
var y = 40 + " is the total";
var z = "10" + 40;
document.write(x + "
");
document.write(y + "
");
document.write(z);

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

```
<script type = "text/javascript">
```

```
var nom;
```

```
nom = prompt ("Enter your name").
```

```
document.write ("Hello " + nom);
```

```
</script>
</body>
</html>
```

Explorer user prompt	<input checked="" type="checkbox"/>
script prompt	<input checked="" type="checkbox"/>
Enter your name	<input checked="" type="checkbox"/>
Cancel	<input checked="" type="checkbox"/>

⇒ prompt() का use करें user
को input देते हैं, फिर उसे input string
करने के प्रे प्रोग्राम को calculate
प्रोग्राम का use होता है।

Subject _____

Date: MON TUE WED THU FRI SAT SUN

Subject _____

Date: MON TUE WED THU FRI SAT SUN

```
<html>
<body>
<script type = "text/javascript">
var a = parseInt(prompt("Enter Ist no.="));
var b = parseInt(prompt("Enter 2nd no.="));
var tot = a+b;
document.write("Total = " + tot);
</script>
</body>
</html>
```

⇒ parseFloat(): इस method के द्वारा float को float के change करता है। अब अपना $\frac{1}{10}$ को float के change करता है।

```
<html>
<body>
<script type = "text/javascript">
var p = parseFloat(prompt("Enter principle Amt ="));
var r = parseFloat(prompt("Enter rate of Interest ="));
var t = parseFloat(prompt("Enter time ="));
var si = p * r * t / 100;
document.write("Principle Amt = " + p + " <br>");
document.write("Rate of Interest = " + r + " <br>");
document.write("Time = " + t + " <br>");
document.write("Simple Interest = " + si);
</script>
</body>
</html>
```

Internet Explorer	X
Chokes user prompt	X
Enter 1st no.	<input checked="" type="checkbox"/> OK
<input checked="" type="checkbox"/> 10	<input checked="" type="checkbox"/> Cancel

⇒ Openpose(): $\frac{1}{10}$ को float के change करता है।

Total = 20

Teacher's Signature

Teacher's Signature

Operators

Different type के calculation & different type के formulae के use में नहीं लाए जाते।
 Operators & operand के need होते हैं।
 यह यह द्वारा ग्रुप के expression को

⇒ Operations of Java Script is as follows:

(1) Arithmetic Operators in Java Script :-
 document.write (" $3 + 3 = " + (3 * 2) + "$ ");
 document.write (" $3 - 2 = " + (3 / 2) + "$ ");
 document.write (" $3 * 2 = " + (3 * 2) + "$ ");
 document.write (" $3 / 2 = " + (3 / 2) + "$ ");
 document.write (" $3 \mod 2 = " + (3 % 2) + "$ ");
 document.write (" $3 \geq 2 = " + (3 >= 2) + "$ ");
 document.write (" $3 \leq 2 = " + (3 <= 2) + "$ ");
 document.write (" $3 > 2 = " + (3 > 2) + "$ ");
 document.write (" $3 < 2 = " + (3 < 2) + "$ ");
 document.write (" $3 == 2 = " + (3 == 2) + "$ ");
 document.write (" $3 != 2 = " + (3 != 2) + "$ ");

</body>

</html>

document.write (" $10 + 3 = " + (10 + 3) + "$ ");
 document.write (" $10 - 3 = " + (10 - 3) + "$ ");
 document.write (" $10 * 3 = " + (10 * 3) + "$ ");
 document.write (" $10 / 3 = " + (10 / 3) + "$ ");
 document.write (" $10 \% 3 = " + (10 \% 3) + "$ ");

</script>

var x = 10;

(3) Logical Operators :-

Generally logical operators के

use relational operator के साथ ही होता है।

(1) Logical AND Operator
 (2) Logical ORS
 (3) Logical NOT

$10 + 3 = 13$
 $10 - 3 = 7$
 $10 * 3 = 30$
 $10 / 3 = 3.3333$
 $10 \% 3 = 1$

Output

⇒ operation की ही सिवाय
 कम्प्रेशन के check करता है। & इनका
 result true या false होता है।

<html>

<body>

<script type = "text/javascript">

document.write (" $3 == 2 = " + (3 == 2) + "$ ");
 document.write (" $3 > 2 = " + (3 > 2) + "$ ");
 document.write (" $3 < 2 = " + (3 < 2) + "$ ");
 document.write (" $3 \geq 2 = " + (3 >= 2) + "$ ");
 document.write (" $3 \leq 2 = " + (3 <= 2) + "$ ");
 document.write (" $3 > 2 = " + (3 > 2) + "$ ");
 document.write (" $3 < 2 = " + (3 < 2) + "$ ");
 document.write (" $3 == 2 = " + (3 == 2) + "$ ");
 document.write (" $3 != 2 = " + (3 != 2) + "$ ");

</body>

</html>

Output

$3 == 2 = False$

$3 > 2 = True$

$3 < 2 = False$

$3 \geq 2 = False$

$3 \leq 2 = False$

```

<body>
<script type="text/javascript">
document.write(("1>5 && 9<6") + ("1>5 && 9<6") + "<br>")
var b = true;
document.write("1b." + (!b + "<br>"));
</script>
</body>
</html>                                         Output
(1>5 && 9<6) : False
(1>5 || 9>6) : true
!b : False

```

(4) Concatenation Operators:

ये गणक से अलिंगनित करने के लिए जाति हैं।

String के साथ numerical value को connect करके ना गए गए off string को जड़ते हैं।

<html>

<body>

<script type="text/javascript">

Var mon1 = "BCA";

Var mon2 = "3rd year";

document.write(mon1 + mon2 + '
');

mon1 + = mon2;

document.write(mon1 + " & " + mon2 + '
');

<script>

</body>

</html>

Output

BCA 3rd year
BCA 3rd year Teacher's Signature

(5) Bitwise Operators

c) Bitwise AND (&):

Bitwise & Logical operation

(ii) Bishnoe, OR (iii) :

660. 3. 17

String \rightarrow numerical value \rightarrow connect string \rightarrow Σ^M .

lhtanl

```
script type="text/javascript">var mom1 = "BCA";
```

Var mno 2 = "Third year."

document. Write (`mon1 + mon2 + "Lbr2"`).
mon1 + = mon2 + "Lbr2"
document. Write (`mon1 + "
"`).
document. write (`mon2 + 420`).
ksright > Output

BCA 3rd year
BCA 2nd year
BCA 1st year

BCA 3rd year
BCA 2nd year
BCA 1st year

Teacher's Signature

Teacher's Signature _____

Subject _____

Date : _____

Date: / /

45 0 1 0 1 0 0 1 0 0 1 0 0 1 0 0 1

A hand-drawn number line on lined paper. The line starts at -1 and ends at 7. It has tick marks every 1 unit. The labels are: -1, 0, 1, 2, 3, 4, 5, 6, 7.

(iii) Otherwise left shift (<<)

Bilwaise left the room-left hand

(6) Assignment operators:

| | | |
|-----------------|------|-----------|
| side | वाली | संख्या |
| right hand side | वाली | संख्या |
| move | मोव | प्रतीक |
| turn | टर्न | प्रक्रिया |

(c) Assignment operator:
The operator \leftarrow
right hand value left hand side

right side ~~over~~ past it. o & add ~~drill~~

प्राते Variable प्रिस्टोगे ए जानि

29. 44. <<1

```
<html>  
<body>  
<script type="text/javascript">
```

o o v i d h e

$$x + (-5) = 10$$

77 77 77 77 77 77 77 77 77 77

document write ("x = " + x + " by");
 $x = 10$

<u>64</u>	<u>8</u>	<u>2</u>
<u>132</u>	<u>16</u>	<u>0</u>

`x = "5"`
document.write ("x = " + x + "
"),

88 - 177 = 11

$$x = 10; \\ x = 5;$$

Indore city any age

$x = 10;$
 $x = 5;$
document.write ("x = " + x + "
");

left hand side ~~right~~ ~~left~~ right
right hand side ~~left~~ ~~right~~ left

$$x = 5$$

Subject	
	Control flow statement
x = 10;	Output x = 15
x = 5;	(1) Branching <script>
document.write("X = " + x + " ");	x = 5
</body>	x = 2
</html>	x = 0
(2) <u>typeof</u> : It operator use for variable	
expression of object to type of	
→ So Pre best way	
<html>	
<body>	
<script type = "text/javascript">	
var m;	
m = parseInt(prompt("Enter a no = "));	
if (m == 0)	
alert("You entered zero")	
</script>	Script prompt <input checked="" type="checkbox"/> OK
</body>	Cancel
</html>	
Output	
document.write("You entered zero")	
var i = 10;	
var pi = 3.14;	
var str = "Hello BCA";	
var n = null;	
var b = true;	
document.write("10 is = " + typeof(i) + " ");	
document.write("3.14 is = " + typeof(pi) + " ");	
document.write("null is = " + typeof(n) + " ");	
document.write("true is = " + typeof(b) + " ");	
<script>	
</body>	
</html>	
Output	
10 is - number	
3.14 is - number	
null is - object	
true is - boolean	
var m;	
m = parseInt(prompt("Enter a no."));	
Hello BCA is - string	
null is - object	
is - object	
true is - boolean	

```
if( m == 0 )
```

if(m < 0)

```
} document.write( "You entered zero").
```

else

```
} document.write( "You entered non zero").
```

else

```
} document.write( " You entered negative value").
```

```
</script>
```

```
</body>
```

```
</html>
```

Output

Enter a no.= 7
You entered non zero

Output

Enter a no.= 0
You entered zero

(iii) Nested if else:

```
<html>
<head>
<body>
<script type="text/javascript">alert( " ")</script>
var n = parseInt(prompt( "Enter a no." ));
if (n > 0)
    document.write( "You entered positive value");
else
    document.write( "You entered negative value");
else if (n < 0)
```

```
} document.write( "You entered zero").
```

```
} else if (n < 0)
```

Teacher's Signature

but while loop की condition false हो जाएगी।

Output

103

```
<html>
<head></head>
<body>
<script type="text/javascript">
```

```
var c;
```

```
c = 1;
```

```
do
```

```
    c++;
```

```
document.write(c + "br");
```

```
}>
```

```
</script>
```

```
</body>
```

```
</html>
```

```
}
```

```
Output
```

```
103
```

```
Nesting of loops :
```

```
जब वह loop के अंदर
```

```
loop का use किया जाता है तो उसे
```

```
Nesting of loops कहते हैं।
```

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

```
var i, j;
```

```
for(i=1; i<=5; i++)
{
    for(j=1; j<=i; j++)
        document.write("*");
}
```

```
</script>
```

```
</body>
```

```
</html>
```

```
}
```

```
Output
```

```
*****
 ****
 ***
 **
 *
```

```
Nesting of loops :
```

```
जब वह loop के अंदर
```

```
loop का use किया जाता है तो उसे
```

```
Nesting of loops कहते हैं।
```

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

```
var c;
```

```
for(c=1; c<=3; c++)
{
    document.write(c);
}
```

```
</script>
```

```
</body>
```

```
</html>
```

```
}
```

```
Output
```

```
1
2
3
```

```
Nesting of loops :
```

```
जब वह loop के अंदर
```

```
loop का use किया जाता है तो उसे
```

```
Nesting of loops कहते हैं।
```

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

```
var c;
```

```
for(c=1; c<=3; c++)
{
    document.write(c);
}
```

```
</script>
```

```
</body>
```

```
</html>
```

```
}
```

```
Output
```

```
1
2
3
```

```
Nesting of loops :
```

```
जब वह loop के अंदर
```

```
loop का use किया जाता है तो उसे
```

```
Nesting of loops कहते हैं।
```

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

```
var c;
```

```
for(c=1; c<=3; c++)
{
    document.write(c);
}
```

```
</script>
```

```
</body>
```

```
</html>
```

```
}
```

```
Output
```

```
1
2
3
```

```
Nesting of loops :
```

```
जब वह loop के अंदर
```

```
loop का use किया जाता है तो उसे
```

```
Nesting of loops कहते हैं।
```

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

```
var c;
```

```
for(c=1; c<=3; c++)
{
    document.write(c);
}
```

```
</script>
```

```
</body>
```

```
</html>
```

```
}
```

```
Output
```

```
1
2
3
```

```
Nesting of loops :
```

```
जब वह loop के अंदर
```

```
loop का use किया जाता है तो उसे
```

```
Nesting of loops कहते हैं।
```

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

```
var c;
```

```
for(c=1; c<=3; c++)
{
    document.write(c);
}
```

```
</script>
```

```
</body>
```

```
</html>
```

```
}
```

```
Output
```

```
1
2
3
```

```
Nesting of loops :
```

```
जब वह loop के अंदर
```

```
loop का use किया जाता है तो उसे
```

```
Nesting of loops कहते हैं।
```

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

```
var c;
```

```
for(c=1; c<=3; c++)
{
    document.write(c);
}
```

```
</script>
```

```
</body>
```

```
</html>
```

```
}
```

```
Output
```

```
1
2
3
```

```
Nesting of loops :
```

```
जब वह loop के अंदर
```

```
loop का use किया जाता है तो उसे
```

```
Nesting of loops कहते हैं।
```

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

```
var c;
```

```
for(c=1; c<=3; c++)
{
    document.write(c);
}
```

```
</script>
```

```
</body>
```

```
</html>
```

```
}
```

```
Output
```

```
1
2
3
```

```
Nesting of loops :
```

```
जब वह loop के अंदर
```

```
loop का use किया जाता है तो उसे
```

```
Nesting of loops कहते हैं।
```

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

```
var c;
```

```
for(c=1; c<=3; c++)
{
    document.write(c);
}
```

```
</script>
```

```
</body>
```

```
</html>
```

```
}
```

```
Output
```

```
1
2
3
```

```
Nesting of loops :
```

```
जब वह loop के अंदर
```

```
loop का use किया जाता है तो उसे
```

```
Nesting of loops कहते हैं।
```

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

```
var c;
```

```
for(c=1; c<=3; c++)
{
    document.write(c);
}
```

```
</script>
```

```
</body>
```

```
</html>
```

```
}
```

```
Output
```

```
1
2
3
```

```
Nesting of loops :
```

```
जब वह loop के अंदर
```

```
loop का use किया जाता है तो उसे
```

```
Nesting of loops कहते हैं।
```

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

```
var c;
```

```
for(c=1; c<=3; c++)
{
    document.write(c);
}
```

```
</script>
```

```
</body>
```

```
</html>
```

```
}
```

```
Output
```

```
1
2
3
```

```
Nesting of loops :
```

```
जब वह loop के अंदर
```

→ Jumping Statement

@ Break :

```

<html>
  help के लिये loop का
  execution के दौरान हमें बहुत बहुत
  अकली ही
  <body>
    <script type = "text/javascript">
      var i = 1, n, t = 0;
      while (i <= 3)
        {
          document.write(i + ", ");
          if (i == 5)
            break;
          i++;
        }
        else
          t = t + n;
        i++;
      }
      document.write("Sum of " + n + " positive no. = " + t);
    </script>
  </body>
</html>

```

Output

1, 2, 3, 4 loop terminated.

(2) Continue :

इसकी help के लिये special situation के loop के दौरान sum का program के control से loop का

बार बार जो वहाँ है

[Enter a positive no. = 3]

Enter a positive no. = 3

[Enter a positive no. = 3]

positive sum of 3 positive nos.

~~Function~~

प्रोग्रामिंग में statements के बहुत सारे types हैं।
functions का use करने से program को आकार में स्ट्रोक्स के जाना होता है।
इससे debug काफी easy होती है।

- (1) Built-in function
- (2) User define function

- (1) Built-in function
 - (2) User define function

(2) Build in function

The java script को library भी ले लियर टो कर्कुत्ता easily perform किये जा सकते हैं। जैसे - setting values of numeric values विं change करना, special character व �expression को handle करना।

\Rightarrow Difference between function & Method

function और method सही तरीके से ही होते हैं।
but इनमें difference है कि एक ही विधि
method पर न किसी object पर
related होती है, जबकि function independent
होती है।
• Method को call करते हैं।
generally related object पर लगती है।
• operator का use करते हैं।
eg. screen पर किसी message को print करते हैं।
use करते हैं। जिसमें document.write()
& write method है।

(2) hasseInt(---)

function to use ~~for~~
string of integer to change to
file test जल्ते हैं।

```
<html>
<body>
<script type="text/javascript">
var a = prompt("Enter a no.");
var b = prompt("Enter another no.");
document.write("Before conversion : ");
document.write(" a+b = " + a+b);
document.write(" After conversion : ");
```

Subject

\Rightarrow Functions :-

/ /

Subject _____

string से function का use number based string expression को solve करने के लिए किया जाता है।

```
<html>
<body>
<script type="text/javascript">
var i;
i = "2 * 3 + 6";
document.write("i = " + i + "<br>");
i = eval("2 * 3 + 6");
document.write(" i = " + i);
</script>
</body>
</html>
```

Output

i = 2 * 3 + 6

i = 12

`document.write(prompt(a) + prompt(b))`.

`</script>`
`</body>`
`</html>`

Output
Output under script prompt
 Enter a no.

Explains user prompt
 script prompt
 Enter another no.

Before conversion 1020
After conversion 30

Before conversion 10.110.120.2
After conversion 30.31

(4) isNaN in `if` function जो use करें तो check

परीक्षा करता है argument no. का कि वह निम्न उम्मीदों में से किसी नहीं नहीं तो `false` return करता।

(3) parseFloat: इस function का use करें तो decimal no. (float)

ग़िरावंश का decimal no. (float)

`<html>`

`<body>`

`<script>`

`var a = prompt("Enter a real no. = ")`;

`var b = prompt("Enter another real no. = ")`;

`document.write("Before conversion ")`,

`document.write(a + b + "
")`;

`document.write("After conversion ")`,

`document.write(false)`;

`document.write('prompt(a) + prompt(b)')`

`</script>`
`</body>`
`</html>`

Output
Output
 true
 false

function fun()

```
? document.write("ABC");
```

```
<html>
<body>
<h1> ABC </h1>
```

(1) event handling function :

```
<html>
<head>
<script type="text/javascript">
function fun()
{
document.write ("Hello java script!");
}
</script>
</head>
<body>
<form action="demo.html" method="post">
<input type="button" value="click to call function" onclick="fun();"/>
</form>
</body>
</html>
```

(2) Argument Pass के बारे में :

Other languages के फलत जैसे C/C++ में function के argument pass करते ही Java script में function के argument को call by value. method के pass करता ही जैसे समझता है।

```
<html>
```

```
<head>
```

```
<script type="text/javascript">
```

```
function add(x,y)
```

```
{
```

```
var z;
```

```
z = x + y;
```

```
document.write ("Sum = " + z);
```

```
}
```

```
</script>
```

```
</head>
```

```
<body>
```

```
<script type="text/javascript">
```

```
add(10, 20);
```

```
}
```

```
</script>
```

```
</body>
```

```
</html>
```

Output

Sum=30

(3) Value के Return के बारे में :

calculate की गई value की calling function
में return करवाया जा सकता है। इसके लिए
javas cript में return statement का use होता है।

Click to call function

Output

```
<html>
<head>
```

```
<script type="text/javascript">
```

```
function add(x, y)
```

```
{
```

```
var z;
```

```
z = x + y;
```

```
return z;
```

```
//document.write ("Sum = " + z);
```

```
}
```

```
</script>
```

```
</head>
```

```
<body>
```

```
<script type="text/javascript">
```

```
var tot = add(10, 10);
```

```
document.write ("Sum = " + tot);
```

```
</script>
```

```
</body>
```

```
</html>
```

```
[Output]
```

```
Sum = 20
```

8. create another file & function call

Programming easy तो file 'java-script' की function द्वारा other file में create कर सकते हैं। लेकिन इसका file name 'js' होना चाहिए,

जिस file से function

call करें तो file का name 'src'

head section में script tag का src

Teacher's Signature

- ⇒ Object :- Object ते functions, वार्षिक वा एक
असी obj. द्वारा property ला method द्वारा
use करते होते असे obj. द्वारा करते
• operators द्वारा करते जाणते आहेत
Java script द्वारा obj. द्वारा two types आहेत.

- ① User define obj.
- ② Build in obj.

- (i) UDO :- UDO द्वारा obj. द्वारा selected property
मध्ये दोन्ही methods uses तरी decide करते
जाने ते, निवारू दोनों types द्वारा नाही।
- (ii) Object initializers द्वारा use करते।

उपरी programme मध्ये myBook name द्वारा object
द्वारा three properties द्वारा करते होते
Value की print करतात।

(iii) Constructor द्वारा use करते।

Syntax :- ObjectName = { property1: Value,
property2: Value,
property3: Value,
... };

```

<html>
<head> </head>
<body>
<script type="text/javascript">
function book()
{
  this.title = "Shogun";
  this.owner = "Vishal Sharma";
  this.RCA = 20000;
}

myBook = new book();
document.write(myBook.title, "<br>");
document.write(myBook.owner, "<br>");
document.write("myBook.RCA", "<br>");

</script>
</body>
</html>
  
```

Output

Shogun	Vishal Sharma
20000	

```

document.write('myBook.title', "<br>");  

document.write('myBook.owner', "<br>");  

document.write("myBook.RCA", "<br>");

</script>
</body>
</html>
  
```

एक program में myBook, वा obj को
 जैसी 3 properties को define करके तो
 values of print करते जाते हैं। इस
 द्वारा keyword को use करके obj.
 create कर सकते हैं।

```

<html>
<head> </head>
<body>
<script type="text/javascript">
function fun()
{
document.write("Hello BCA");
}
function book()
{
this.title = "C++";
this.author = "Chupia Lekhyani";
this.price = 190;
this.myfun = fun();
myBook = new book();
document.write(myBook.title, "<br>");
document.write(myBook.author, "<br>");
document.write(myBook.price, "<br>");

myBook.myfun();
</script>
</body>
</html>
  
```

(2.) Build In Object :
 यह तैयार object होता है, जो
 java script में प्रैग्सी से मिलते
 हैं तो types द्वारा divide होते हैं -

- (i) Language object
- (ii) Window object

i) Language object :

[Language]

- [Array]
- [Boolean]
- [Date]
- [Math]
- [String]
- [Number]
- [Function]

Array : यह कुछ values को group होना है।
 index का use किया जाता है। Java script
 में यह array create करके तो object का
 use किया जाता है।
Syntax : arr.obj.name = new array (value1,value2,...)
 or
 arr.obj.name = new array (length)

Subject _____

Date : / /
MON TUE WED THU FRI SAT SUN

Subject _____

Date : / /
MON TUE WED THU FRI SAT SUN

```

<html> </head>
<body>
<script type="text/javascript">
var = new Array ("Anil", "Sunil", "Ajay");
Set = new Array ('s');
Set [0] = 8000;
Set [1] = 10000;
Set [2] = 12000;

document.write ("Names are", " ", "<br>");

document.write (var [0], "<br>"),
document.write (var [1], "<br>"),
document.write ("Salaries are", "<br>"),
document.write (" ", "<br>"),
document.write (Set [0], "<br>"),
document.write (Set [1], "<br>"),
document.write (Set [2], "<br>");

</script>
</body>
</html>

```

→ METHODS OF ARRAY OBJECT :

(i) Concat () : इस method का लिए एक array दर्ता होता है। जो दो वाले array को जोड़कर एक नया array बनाता है।

```

one = new Array (1, 2, 3),
st = one.join ();
document.write (st, "<br>");

st = one.join (" - ");
document.write (st);

```

Output

[1,2,3,4,5,6]

(ii) join () : इस method का use किसी array के elements को एक string में कैसे जोड़ता है। यह फल एक string होता है। जो elements के बीच by default (,) वाला जाता है। यह इनके बीच character नहीं होता।

```

<html>
<head> </head> <script type="text/javascript">
<body>

```

Output

1,2,3

<head>

<head>

<body>

Subject _____

Date : _____

Date : / /

(iii) `pop()`: JS method array to last element to array after array shift

```

<html>
<body>
<script type = "text/javascript">
one = new Array(1,2,3);
document.write("Array elements Before reverse:");
st = one.pop();
document.write("Last element after pop:", one, "<br>");
document.write("Array elements before pop:", one, "<br>");

</script>
</body>
</html>

```

Output

Array elements Before reverse: 1,2,3
Last element after pop: 1,2
Array elements before pop: 1,2

(iv) sort():

JS method to use array element to ascending order & manage array to sort

```

<html>
<body>
<script type = "text/javascript">
one = new Array(3,2,1,4,5,6);
document.write("Array elements before sorting:");
one.sort();
document.write("Array elements after sorting:");
one.sort();

```

Array elements before sorting: 3,2,1,4,5,6
Array elements after sorting: 1,2,3,4,5,6

(v) reverse():

JS method to use array element to reverse order & manage array to reverse

```

<html>
<body>
<script type = "text/javascript">
one = new Array("Sunit", "Anil", "Vijay", "Ajay");
document.write("Array elements before sorting:");
one.reverse();
document.write("Array elements after sorting:");
one.reverse();

```

Array elements before sorting: Sunit, Anil, Vijay, Ajay
Array elements after sorting: Ajay, Vijay, Anil, Sunit

`document.write ("Array elements after sorting" +
one, "abc") ;`

`</script>` ~~Output~~ ~~no - no~~
`<body>` ~~Output~~ ~~no - no~~
`</body>` ~~Output~~ ~~no - no~~
`</html>` ~~Output~~ ~~no - no~~

"Array element before sorting: 4, 3, 1, 2, 5
 Array element after sorting: 1, 2, 3, 4, 5

Array element before sorting: Sunil, Anil, Vijay, Ajay

Array element after sorting: Ajay, Anil, Sunil, Vijay.

(vi) unshift (item 1, item 2, ...)

method to insert array beginning of
 to store element at odd index of arr

arr = new Array ("Anil", "Sunil", "Ajay");
`arr = new Array ("Anil", "Sunil", "Ajay");`

`<script>` ~~Output~~ ~~no - no~~
`<body>` ~~Output~~ ~~no - no~~
`</body>` ~~Output~~ ~~no - no~~
`</html>` ~~Output~~ ~~no - no~~

length() : " property to use array ?

(vii) length() : " property to use array ?
 elements of array check करें ?
 for first 1

for first 1

`<script type="text/javascript">`
`arr = new Array ("Anil", "Sunil", "Ajay");`

`document.write("No. of elements are = " + arr.length, "`

`<script>`

`<body>`

`</body>`

`</html>`

→ Two dimensional Array

To situations : one-dimensional array &
 values ~~परिस्थिति~~ करने के ~~परिस्थिति~~ नहीं रहते हैं तो
 problems को solve करने के लिए
 2-d का use करते हैं।

Two-d. declaration : [System]

array-name = new Array (dimension);

document.write ("Array elements after unshift : ",

one, "
");

one.unshift (201, 202).

document.write ("Array elements after unshift : ",

one, "
");

one.unshift (201, 202).

document.write ("Array elements before unshift : ",

one, "
");

one.unshift (201, 202).

document.write ("Array elements before unshift : ",

one, "
");

one.unshift (201, 202).

document.write ("Array elements before unshift : ",

one, "
");

one.unshift (201, 202).

Teacher's Signature

says it's columns तरीके and rows जैसी हैं।
 Column नाहीं तरीके एवं following way है।
 declare करें।

eg. arr[0] = new Array(2);
 arr[1] = new Array(3);

⇒ 2-0 तरीके use करने के लिए values of
 above तरीके "first" करवाना है।
 <html>
 <body>
 <script type="text/javascript">

arr = new Array(2);
 for (i=0; i<2; i++)
 {
 arr[i] = new Array(3);
 for (j=0; j<3; j++)
 {
 arr[i][j] = prompt("Enter a number");
 }
 }
 </script>

⇒ Date object
 Date object जैसा है। date object
 to use date & time के related work
 perform करने के लिए जरूर है।
 Date object के Date() constructor के
 help से create किया जाता है।

⇒ Date object के method
 (1) getDay(): इस method का use week के
 day return करते हैं। लिए Sun=0, Mon=1, Tuesday=2, ...

eg. (Sun=0, Mon=1, Tuesday=2, ...)

(2) getMonth():

इस method का use date
 object की से month return करते हैं।

eg. (Jan=0, Feb=1, ...)

(3) getFullYear():

इस method का use date object
 की से year का return करते हैं।

(4) getHours():

इस method का use date object
 document.write("The current hour is ",);
 document.write("The current minute is ",);
 document.write("The current second is ",);
 </script> </body>

(5) getMinutes() :- यह method का use date obj. के से minute second का return करता है।

(6) getSeconds() :- यह method का use date obj. के seconds return करता है।

(7) getMilliseconds() :- यह method का use date obj. के milliseconds return करता है।

ss = today.getSeconds();
 ms = today.getMilliseconds();
 document.write("Hours = " + hh, "Minutes = " + mm, "Seconds = " + ss, "Milliseconds = " + ms);
 <script> </body> </html>

Output

Today is Sun May 23 03:02:59 PDT 2010

Day is = 0
 Month is = 4
 Year is = 2010

Hours = 03
 Minutes = 02
 Seconds = 59

Milliseconds = 15

Programme :-

```
<html>
<head>
<body>
<script type="text/javascript">
today = new Date();
document.write("Today is, " + "today", "<br>");
```

d = today.getDate();

year = today.getFullYear();

mon = today.getMonth();

document.write("Day is = " + d, "
");

" ("Month is = " + mon, "
");

" ("Year is = " + year, "
");

hh = today.getHours();

mm = today.getMinutes();

Window Object

यह object client browser का प्रतिक्रिया करता है। जोसमें

following obj. includes :-

Window

Layer
 Link
 Password
 Hidden
 Submit
 Area
 Reset
 Radio
 Checkbox
 Form
 Textarea
 Option

⇒ Window Object

Window Obj.

Closed

इस property का use ज्यू check करने के लिए लिखा जाता है। To window.close();

ग्रंथि: एक boolean value returns जोफ़।

Syntax: - window.closed;

defaultStatus

इस property का use browser की status bar के लिए लिखा जाता है। To default message print

Syntax: - window.defaultStatus = "Message";

frames

ज्यू property window के सभी frames का array है। Syntax>window.frames["Frame"]

status

set property का use off status bar का message set करने के लिए लिखा जाता है। ⇒ Window object का methods :

1 Syntax: window.status = "Message";

(1) alert():

इस property का use web page के available frames के window जाते हैं।

message, box display करने के लिए लिखा जाता है। Syntax: - window.alert("Message")

Syntax: window.length

open इस property का use new window के source windows की properties है।

methods # 1 use के लिए लिखा जाता है।

इसका use new window के new window.open(), closed() है।

e.g. open के लिए new window के new window.open(), closed() है।
use कर source window के close करने का लिखा जाता है।

(3) close() :

इस method को use window.close() करने के लिए किया जाता है।
Syntax:- window.close()

(4) confirm() :

इस method को use window.confirm() करने के लिए cancel को लिए गया है।
Input रेखे के लिया आवंट है। अग्री user
ok पर click करता है तो इस true,
return करता है तभी ज्ञान false देता।
Syntax:- window.confirm()

(5) focus() :

इस method को use window.focus() करने के लिए focus set करने है। ऐसा जाना है।
Syntax:- window.focus()

(6) moveBy() :

इस method को use window moveBy() करने के लिए x, y location set करना है। यह
depends on pixels.
Syntax:- window.moveBy(x,y)

(7) moveTo() :

इस method को use window.moveTo() करने के लिए x, y location set करना है।
परन्तु करने के लिए किया जाता है।

Syntax:- window.moveTo(x,y)

(8) prompt() :

इसका use के input रेखे
करने के लिए किया जाता है।
Syntax:- window.prompt()

(9) open() :

इसका use new browser window open करने के लिए किया जाता है।
Syntax:- window.open("URL", name
[, features])

eg:- window.open("http://www.itgamer.com", "width=800px, height=600px, resizable")

(10) print() :

इसका use window.print() करने के लिए किया जाता है।
Syntax:- window.print()

(11) resizeBy() :

इसका use window.set size करना है।
Syntax:- window.resizeBy(100, -50)
window.resizeBy(pixels, verticalpixels)

(12) Document Object:

इसकी help से इस object page के
elements को access कर सकते हैं।
eg. इस obj की help से इस text box की
value किसी variable में store कर सकते हैं।
जैसे text box को किसी नहीं value नहीं
set कर सकते हैं।

इस obj का use टेबल et.
properties के साथ भी कर सकते हैं।

⇒ Methods of document objects & Properties

(B) **titleColor** :- इसका use web page का title set करने के लिए किया जाता है।

Syntax :- Document.titleColor = "Web page";

(1) **linkColor** :- इस property का use use web page द्वारा hyperlink का color set करने के लिए किया जाता है।

Syntax :- document.linkColor = "colorinfo";

(2) **alinkColor** :-

इस property का use web page द्वारा active hyperlink का color set करने के लिए किया जाता है।

Syntax :- document.alinkColor = "colorinfo";

(3) **vlinkColor** :- इस property का use use web page

द्वारा visit link की hyperlink color set करने के लिए किया जाता है।

Syntax :- document.vlinkColor = "colorinfo";

<body>

<script type="text/javascript">

document.linkColor = "red"

document.alinkColor = "green"

document.vlinkColor = "blue"

</script>

Hello

Hello

Hello

</body>

</html>

(4) **bgColor** :- इसका use web page "background" color set करने के लिए जाना है।

Syntax :- document.bgColor = "colorinfo";

(5) **fgColor** :- इसका use web page का द्वारा display के रूप text का color set करने के लिए जाना है।

Syntax :- document.fgColor = "colorinfo";

Output

Hello

⇒ Special Methods of Document Object

(1) write() :-

इस method का use screen पर message जैसे html expression को दर्शाने के लिये जितना है।
Syntax - document.write("expression (S)");
eg. document.write("Hello ");

(2) getElementById() :-

इसका use form के भाग पर access करने के लिये जितना है।

Syntax :- document.getElementById("ID");
document.getElementById("text name");

(3) getElementByName() :-

इस method का use form के भाग पर access करने के लिये जितना है।

Syntax:- document.getElementById("Name");
eg. document.getElementById("text name");

forms & Event

→ Introduction :-

किसी web page पर use करें form वाले तो उसमें different element होंगे जैसे - text box, check box, radio button, command button etc. In different elements में user कर सकते हैं जैसे input को द्वारा javas cript के validate कर सकते हैं जिसको meaning यह है कि input किया गया अदृष्ट है या नहीं।

→ forms :-

form की help के web page के attractive ढौऱ्हा जा सकता है। किसी web page के data बोला server or send करने के लिए form tag का use किया जाता है।

form tag के attributes following हैं -

- ① <input>
- ② <select>
- ③ <text area>

(i) name :-

इस attribute का use form की name के लिए किया जाता है।

(ii) input :-

Input tag की help के web pages में इस attribute में उस form के script को नाम दिया जाता है, जिस form के submit button पर click करें तो use करना है।

eg. `<input type="text" />`
`<input type="password" />`
`<input type="radio" />`
`<input type="checkbox" />`

(iii) method :-

जब attribute को माध्यम से server पर data send करने के two types होते हैं -

(a) get() :-

get() के data को send करना जाता है, नीचे वह data browser की address bar में display होता है।

(b) post() :-

post() के display कुछ नहीं होता, data direct server पर जाता है।

⇒ Form tag :-

Form tag के following tag का use होता है

- ① <input>
- ② <select>
- ③ <text area>

name के लिए attribute का use form की

name के लिए किया जाता है।

eg. `<input type="text" />`
`<input type="password" />`
`<input type="radio" />`
`<input type="checkbox" />`

Teacher's Signature

Subject _____

Date : / /
MON TUE WED THU FRI SAT SUN

Subject _____

Date : / /
MON TUE WED THU FRI SAT SUN

(2) `<select> tag को help से वेब पेज पर ड्रॉप डाउन लिस्ट गेनरेट की जाती है। इसका उद्देश्य एक ऑप्शन रिप्रेसेंटेशन को एक एलेमेंट की रूपांतरण करना है। ऑप्शन रिप्रेसेंटेशन को वेब पेज पर विभिन्न जाता है।`

(3) `<textarea> tag को help से वेब पेज पर ड्रॉप डाउन लिस्ट गेनरेट की जाती है। इसका उद्देश्य एक लाइन ओफ टेक्स्ट बॉक्स बनाना है।`

eg.
`<html>
<head> <title>
<title> form-Demo </title>
</head>
<body>
<form name = "form" method = "get" action = "demo.html">
Enter your name: <input type = "text" size = 20>

Enter password: <input type = "password" size = 20>

`

Output

Enter your name:
Enter password:
Enter Gender: Male Female
Enter Hobbies: Music Games
Select your city: Ajmer
Enter your comments:

Enter Gender: `<input type = "radio" name = "g" > male
<input type = "radio" name = "g" > female`
`

`

Hello	Submit	Reset
-------	--------	-------

Enter Hobbies: `<input type = "checkbox" > Music`
`<input type = "checkbox" > Games

`

Select your city:
`<select>`

`<option> Ajmer </option>`
`<option> Jaipur </option>`

Teacher's Signature

Teacher's Signature

⇒ Event :

किसी element से किसी गति validate करने के लिए यह element input को किसी दूसरी event का use किया जाता है। जैसे जब user form में complete लगा है, तो उसके दूसरे submit button का onsubmit event का use करता है।

Java script के तरह events focussing है -

(1) onabort :

जब event तभी घटता है, जब user web page पर image को load करता है।

(2) onbeforeunload :

यह event तब घटता है, जब user page पर बैठता है।

(3) onchange :

यह event तब घटता है, जब user page पर डार्किंग किसी obj: पर से focus चला जाता है।

(4) onclick :

जब user text box से click change करता है।

(5) oncontextmenu :

जब user किसी element पर click करता है।

(6) ondblclick :

जब user किसी element पर click करता है।

(7) onerror :

जब web page या image का नहीं लिया जा सकता है।

(8) onfocus :

जब किसी element का focus करता है।

(9) onmouseover :

जब किसी element पर से mouse pointer लगता है।

(10) onmouseout :

जब किसी element पर से mouse pointer नहीं जाता है।

(11) onselect :

user ने text area से किसी text को select करता है।

(12) onsubmit :

जब user "submit" button पर click करता है।

(13) onunload :

जब user web page को close करता है।

Element

event

form onblur, onchange, onfocus, onselect,
onsubmit

Image onabort, onload, onerror
Link onclick, onmousedown, onmouseup, onmouseover

Window onbefore, onerror, onfocus, onload,
onunload

Enter IdNo.:
Enter IdNo.:
Result =

Output

eg:
<html>
<head>

<script type="text/javascript">
function add()
{

```
val a = document.getElementById("textNo1").value  
val b = document.getElementById("textNo2").value  
var tot = parseInt(a) + parseInt(b).  
document.getElementById("textResult").value = tot  
}
```

eg.
text box result text +
program, button, onclick

</script> </head>

<body>

<form name="form" method="get" action="demo.html"

Enter IdNo. = <input type="text" id="textNo1">

Enter IdNo. = <input type="text" id="textNo2">

Result = <input type="text" id="textResult" />

Subject _____

Date : _____
MON TUE WED THU FRI SAT SUN

Subject _____

Date : _____
MON TUE WED THU FRI SAT SUN

Internet Explorer
Hello this is a Demo --
Google

→ Exception Handling :

exception handling का use करने के programme

के लिए exception का handle करना है।

e.g. कि यह अच्छे programme का run करने के time, user को no. या लाचा, सीधे error, situation के run time error.

Java script में यह run time error

की exception करना है।

(i) try-catch block का use करें :-

Note - जब try block में किसी statement (code) ने आवृत्ति (try-catch) का divide कर देता है। try block में वह code write किया जाता है, जिसमें error आने की संभावना है, और catch block में try block से बास हुई error को handle करने के लिए code लिखा जाता है।

<html>
<body>
<script type="text/javascript">
try
{
document.write('a');
}
catch (err)
{
document.write(" Error Number: ", err.number,
"
");
(ii) On error event का use करें

```
</script>  
</body>  
</html>
```

Error Number: 2146029279
Description: 'a' is undefined

Teacher's Signature

Subject _____

Date : / /
MON TUE WED THU FRI SAT SUN

Subject _____

Date : / /
MON TUE WED THU FRI SAT SUN

```
<html>
<body>
<script type="text/javascript">
try {
    document.write("a");
} catch (err) {
    alert(err);
}
</script>
</body>
</html>
```

```
Output
Hello
Error occurred
```

Document obj. model वा [www] के standard द्वारा अंदर document द्वारा access करने के standard define करते हैं। Other words यह document obj. model किए document के structure, style तथा content को dynamically access & update करने के form program & script के form platform provide करते हैं।

जब तक web page load होता है, तब तक Java script की help से DOM तथा HTML tags की dynamic access कर सकते हैं जो change कर सकते हैं।

⇒ Document object Model

[Document]

[HTML]

[Head]

[Body]

[TITLE]

[IMG]

[FORM]

[ANCHOR]

[TABLE]

[Button]

[TEXT]

[Checkbox]

Teacher's Signature

Teacher's Signature

1. 'HTML' document का meaning 'web page' है।

2. Object जो mean different elements का text, attributes etc.

3. Model का mean ~~is~~ ~~not~~ & structure है।

⇒ DOM Accessible Basic HTML Elements

1. Anchor
2. Form
3. Textarea
4. Checkbox
5. Link
6. C. Radio
7. Select
8. Option
9. Reset
10. Button
11. Input

To browser tags को nested list के रूप में पढ़ते हैं। लाइन html tag को पर हीला & वाली के tags need nested order का हो।

<HTML>

<HEAD>

<TITLE>

<BODY>

<P>

<H2>, <H3>, <H4>, <H5>, <H6>

<FORM>

<TABLE>

इन सभी tags को access करने के लिए DOM API का प्रयोग करते हैं और present tag को access करने के लिए DOM API का use करते हैं। यह -

form को use करके करते हैं। इस document.form property का use करके फॉर्म form के fields के value को access करते हैं जो field के name property का value होता है। इसका feature है। DOM का java script help के लिए tags को dynamic access करते हैं।