

#web design ~~now~~ is after Javascript

Page No. \_\_\_\_\_  
Date \_\_\_\_\_

netscape navigator

### #Javascript

We can use console in browser to write javascript  
in console you have to type ~~or~~ shift + Enter so that  
you can write multiline javascript

→ Javascript line end with ; (semicolon)

Ex: alert("Hello");  
alert("World");

But we have write code properly so we will use the  
snippet in sources. & create a file index.htm

F12 → sources → snippet → write d Javascript

& you can run it at end of file which show a  
key to run it

Console is for one line command

Snippet is for whole big code & run at once

⇒ we get codes from snippet file to console.

use Mdn for reference in Javascript

so now lets break down the code used in JS  
& how it works

All functions in JS will give behaviours to ur website

Page No.	1/1
Date	

alert("Hello");

Function

End

JS with human analogy

human

Say: "Hello".

Initial

message

JS

alert("Hello");

End

say & alert are the  
functionalities

matter @ in " "  $\Rightarrow$  is message or syntax.

$\Rightarrow$  Read readon/diomatic JS

language

$\Rightarrow$  The element of style.

① Data types in JS

① String

alert("Hello")

String

Everything in double quotation is

String

② Numbers (1, 2, 3)

③ Boolean true & false

type of (datatype-item);  $\Rightarrow$  type of (2)  
output datatype  
"number"

### #Variable in Javascript

`prompt ("what is your name");`

So prompt is advanced version of `hello alert` where you can take input from it.

So in order to store piece of information we can use variable

Syntax to store variables:

`var name = "Ghansham";`

so whenever above script runs & you want value inside name it will give it as Ghansham

Var    name = "Ghansham";  
    ↑        ↑  
  Keyword    name of    Value which you  
  to create    the            want to give to that  
  a variable    Variable            variable

$\Rightarrow$  You can declare any type of variable with var & it will automatically detect datatype

Ex:-

\* `Var name = "Ghansham"  $\Rightarrow$  String`

`Var DOB = 17012002  $\Rightarrow$  Number`

`Var Bool = true  $\Rightarrow$  Boolean datatype`

Var => It will create a ~~block~~ block & block is named  
with word you gave & after = (is equal to) that  
data will stored in the that box.

=> you can overwrite values or from variables.  
=> var only used once i.e. when you have create  
a variable after that you can access that  
variable only by using its name.

You can write as follow also in JS

Var myname = "Gaurav"; Var no = 17012002  
Alert ("My name is " + myname + " Birthdate is "  
+ no + " in DDMMyYY format"),

Output: my name is gaurav Birthdate is 17012002  
in DDMMyYY format.

+ is important as it is concanate string &  
variable

① naming conventions

so as you add variables it will get added in the  
sources & cache so you have to remove it  
so you just have to do a thing

long press on reload & do hard cache  
deletion

② Rules for naming variable

=> always give meaningful names so anybody can  
understand your code & it make easy to  
find material such as food rpu

- ⇒ You can't use a keyword as variable name
  - ⇒ Variable can't begin with no. but can be Content numbers
  - ⇒ Variable name should be continuous

You can only use only

## alphabet

Number

\$ dollar

## Camel Case

$\Rightarrow$  any Number present here

\_ underscore

## #String Concatenation

you can add strings with a plus sign only

"a" + " " + "b"  $\Rightarrow$  "a b"

## # Javascript string

word.length & it will show length of the Variable variable.

```
Ex: var name = "Ghansham";  
alert(name.length);
```

& it will get printed

Nan  $\Rightarrow$  a error type

$a+b$  in normal but in alert it should  
be  $(a+b)$

### Slicing

⇒ So many time we need few but get many so in order to cut off need & left exceeded explicit part will help us.

0 1 2 3 4 5 6 7  
ghansham

### # slice function

var name = "ghansham"  
 + less than (not including 2 index)  
 • name.slice(0, 2)  
 Including 0 1  
 • gh ⇒ output

var name = "Hello world"  
 0 1 2 3 4 5

name.slice(0, 5)  $\Rightarrow$  0 1 2 3 4  
 So

Hello.

larger the range larger the no. of elements.

Slice will help us to deal with string element separately via slicing

### # string cases

Var Variable = "ghansham" | Var Variable = "GRS"  
 Variable.toUpperCase() & : | Variable.toLowerCase()

Output  $\Rightarrow$  GHANSHAM | Output  $\Rightarrow$  grs

Javascript uses camel type of scripting in naming of functions

# numbers in Javascript.

# Functions in Javascript

It helps to avoid repetition of code.

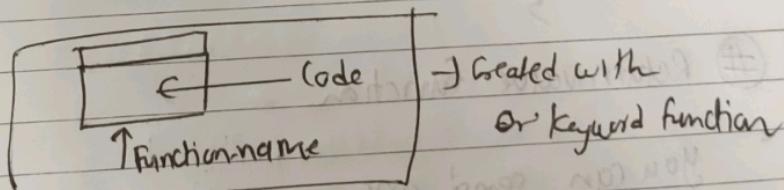
function it is single block of code which can be used only in any time with function name only

Syntax::

```
function function-name() {  
    ↑  
    Create the function  
    }  
    /& Code inside function
```

& call function by  
calling → function-name();  
the function

⇒ function is just like variable keyword function will  
create a box that box is named as function name  
code inside function will put in that box



⇒ alert is hectic so if you want to print output  
in console

Console.log('')  $\Rightarrow$  only for developer

Page No.	
Date	

use

```
Console.log("command");
```

↑ But output is only visible to developer not to users

# Functions & parameters

### ① Parameters function:

You don't have to specify type

```
function getmilk(bottles){}
```

// code

}

Bottle is passed as a parameter to above function

Ex fun to add 2 numbers

```
function add(a,b){}
```

```
    alert("a+b");
```

}

### ② Return value function

You can send one or more return values from function.

Return keyword used for returning value

function returnfunction(a, b) {

$$\text{var } c = (a+b) + (a*b) + (a-b) + (a/b)$$

return c;

5

If you can call by

alert(returnfunction(10, 5));

or var f = returnfunction(10, 5);

You can call multiple functions (nested fun in each other)

at 1 st we know we are at 2nd

then we will write function

function function2() {

function function3() {

function function4() {

ab5

<1> multiplication = a \* b

<2> a <3> a \* a = a \* a

<4> (a) result = a

=

<(a) result>

Intermediate Javascript  
# Random number generation

Var n = Math.random();

It will create a 16 decimal digit number  
from  
0 - 0.9999999999999999

and it will create random no. every time

④ to convert a float no to integer you can  
use

Var m = Math.floor(number);

Trick so If you want no betw 1 to 100  
so Just multiply float value with 100  
(Randomly created)

& you will get number from that range.

after multiply you get no from 0 to 99  
so you just have add 1

Code

Var n = Math.random();

n = n \* 100; → so that we  
get no. ~~from~~  
n = Math.floor(n) + 1; ~~to~~ 100

console.log(n);

## # If else statement

Syntax:

if (condition) { }

Code ; if condition == true

}

else { }

Code ; if condition != true

}

\* i.e false.

## # Operators for condition in Javascript

== Is equal to

!= not equal to

> greater than

< smaller than

>= greater than or equal to

<= less than or equal to

== => two equal sign check for only values

== => three equal sign checks for value & also for the datatype

### logical operators

$\&\&$   $\Rightarrow$  logical and  
 $||$   $\Rightarrow$  logical or  
 $!$   $\Rightarrow$  logical not

### # Javascript array.

we can store multiple type of data in same variable with js array & we just access data in constant time with the help of indexing

Ex:-

`var eggs = [ E1, E2, E3, E4, E5 ]`

Output

`E{eggs[1]}  $\Rightarrow$  E2`

`eggs.length`  $\Rightarrow$  50 It will provide no. of elements there inside the array

① includes (value)

Used on `eggs.includes (E3)`

so it will show that is E3 is present in Eggs array or not.

It will boolean as output True or false.

In Javascript array you can insert elements

of different datatype also

Ex

Var n1 = ["ghansham", 1, 2, 3, true];

word/string

Boolean

& Even though it will not give error

a) Print 1 to 100

for multiples of 3 => Fizz

for multiples of 5 => Buzz

(for multiple of 15 => FizzBuzz)

two useful array function

- ① push() => to push array element at last
- ② pop() => to pop the last element out of the array & decrease length of array at same time

→ All variable outside functions are global variables & we can use it in code without passing them as parameters

while  $\Rightarrow$  state based  
for  $\Rightarrow$  Iterate

Page No.	
Date	

### # loops

Syntax    while (condition) {  
              Code ; }

Just like C

for loop just like C  
start and change  
for( $i=0$ ;  $i < \text{Range}$ ;  $i++$ ) {

Code //

Infinite loop is there if you provide all required  
fields

1 1 2 3 5 8

array as output

$$n_2 = n_1 + n_2$$

$$\boxed{n_2 = 2}$$

$$\circled{n_1 = 1}$$