Article on some basic things Of JavaScript

# JavaScript

JavaScript was created on ***September 1995 by Brendan Eich.*** In 4 December 1995 people started using JavaScript. At that time JavaScript ***wasn’t popular*** for making apps and games but it was use in html CSS for some adding features and designing. In present time, JavaScript is so popular as it uses for making games and apps.

Fact:

1. Eich took only 10 days to develop the JavaScript language. It shows that at that time JavaScript language was so small.

Variables

Variables means to vary. Variables hold data value and can be change anytime. We use a keyword var to declare word variable. A variable has a unique name given by you.

We use variable given following:

var name = value

Example:

var number = 1,2,3,4,5,6,7,8,9

Data types

Data type can ***store anything*** in it. String, Number and Boolean are examples of data types.

String

In String we can ***store data*** but we ***cannot change its value***.

var name = permanent value

Example:

var x = Naitik

Undefined Value

A variable that has ***not been defined*** so it shows undefined.

var b;

var name =­ 1variable name

Example:

var test;

var x = test

Null

Null in JavaScript means ‘***Nothing’***. It also represents ***empty*** or ***unknown.***

For example:

­var b;

Loops

JavaScript **loops** are used to again and again repeat the block of code - until a certain condition is met.

There are many types of loops

Example:

***For loop*** = for(i=0;i=>10;i++){

  condition for if it is true so first condition if false second condition

}

***For Each loop*** = var fruits = [“apple, “orange”, “cherry”];

fruits.forEach(myFunction);

function myFunction(item, index) {

  document.getElementById(“demo”).innerHTML += index + “:” + item + “<br>”;

}

Conditions

Condition in JavaScript is same as conditions in English. In JavaScript also we use if and else to put a condition.

We use if else condition given following:

if (a ==””) {

  document. write(“One”)

  }

  else {

  document. write(“two”)

  }

Functions

A ***function*** is a block of code in which we ***define*** something once and called multiple times later.

There are multiple types of function. Example:

1. Anonymous functions:

An ***anonymous*** function is a function ***without*** a name. The **anonymous function** has no name between the **function** keyword and ().

1. Callback Functions:

A ***callback*** function is a function which ***passed*** to ***another function*** as an argument. The sign to call callback function is ***=>***.

Let

Let allows to declare variables with limited scope of a block. In let we use keyword let to declare. Let is a block scope variable. Let also have unique name as var have.

We use let given following:

let name = value

let me = “Naitik”

Scope

***Scope*** defines the accessibility of these variables. Each function or variable creates a new scope.

There are 2 types of scopes:

Global scope:

In JavaScript, Global scope can be using any where in that page of JavaScript.

Block/Function scope:

In JavaScript, block/function scope can be using only in the particular

function in which it has been declare of JavaScript.

Event propagation

***Event propagation*** is a way or type to describe the stack of ***events***.

Event propagates form window down to DOM Element. For example, <html> then <body> then <section> and then <div>.

Event bubbling

Event bubbling is just opposite from event propagation which when first triggers the innermost html element.

For example, <div> then <section> then <body> and then <html>.

Event capturing

Event capturing is just same as event propagation which when, it starts from top to bottom as event propagation.

For example, <html> then <body> then <section> and then <div>.

Event target

Event target return the element which triggered event. In event target we target on a particular Element in JavaScript.

We use event target by writing event.target.

High order function

***High order function*** is a ***function*** that can take another ***function*** as an argument or that returns a **function** as a result.

There are many types of high order function. Example:

1. Filter method:

Filter method is method from which we can fill an array. Filter method is not array really.

We use filter method given following:

fillingState=[“”,””,””,””]

 let boxFilled = fillingState.filter((element) => {

      return element != "";

    });

1. Reduce method:

The ***reduce***() method reduces the array to a single value. This only reduces the value does not increase.

We use reduce method given following:

  function plus(numbers) {

    let total = numbers.reduce((accumulator, element) => {

      return parseInt(accumulator) + parseInt(element);

    });

    return total;

  }

1. Map method:

The ***map () method*** creates a new array with calling of a ***function*** for every array element.

var numbers = [65, 44, 12, 4];

var newArray = numbers.map(myFunction)

function myFunction(number) {

  return number \* 10;

}