**JAVASCRIPT**

* Developed by Brandon Eich
* Named to make use of Java market buzz

**Object-oriented and Object-based language**

To qualify as object-oriented, programming languages must provide support for the following:

* Data Abstraction
* Encapsulation
* Data protection ( Under Construction)
* Inheritance

**Simple objects**

* function Dog (name) {

this.name = name;

}

* var fido = new Dog('Fido');

console.log(fido.name); // 'Fido'

* JavaScript does not support data abstraction in the form of Classes, neither is there support for data protection.
* However, JavaScript is defined as an OBJECT ORIENTED LANGUAGE

**Object Life Cycle**

* Objects are created, used, and discarded
* Constructors are implicit in JavaScript - natural
* A constructor in a class is a special block of statements called when an object is created
* Destructors are not provided by JavaScript
* class Point {

constructor(x, y) {

this.x = x;

this.y = y;

}

static distance(a, b) {

const dx = a.x - b.x;

const dy = a.y - b.y;

return Math.hypot(dx, dy);

}

}

const p1 = new Point(5, 5);

const p2 = new Point(10, 10);

p1.distance; //undefined

p2.distance; //undefined

console.log(Point.distance(p1, p2)); // 7.0710678118654755

**Sample for usage of objects in html**

// key : value pair

name:"xyz",

class:"10th",

hobbies:"singing"

}

console.log(obj)

//dot notatation .

console.log(obj.class)

console.log(obj.hobbies)

console.log(typeof(obj))//object

console.log(typeof(arr))//object

**To insert something on the object**

//insert

obj.phone=78998646

console.log(obj)

//[] xxxx .

obj["landline number"]=34567890098765

obj['grade score']="B"

console.log(obj)

**To update something on the object**

//update

obj.name="abc"

console.log(obj)

**To remove something on the object**

//remove

delete obj.hobbies

console.log(obj)