*[Font/size: DejaVu Sans Mono/12] [ ForCode Font/size: Consolas/11]*

***JS is a loosely typed language***

let abc;

abc = "Hello World!";

document.write(abc);

---

let abc;

abc = 5;

document.write(abc);

---

let abc;

abc = 5;

document.write(abc);

abc = "Hello World!";

document.write(abc);

* *We cannnot declare the same variable twice in the code.*

-----

**let**

* It has Block level scope.
* Same Variable cannot be declared twice.
* The variable needs to be initialized.

if(some\_condition){

var abc = "Hello World!";

}

document.write(abc);

-----

**var**

* It has a fucnction or global scope.
* Same Variable can be declared twice.
* No need to initialize the variable.

if(some\_condition) {

let abc = "Hello World";

}

document.write(abc);

-----

**const**

* It is used to declare a constant variable whose value cannot be changed.

***Naming Conventions***

* Contain alphanumeric characters, underscore(\_), and dollar ($).
* Example: \_abc, $abc, abc123, etc.
* Cannot start with a number. Example: 123abc, 123\_abc,etc.
* Cannot contain special characters like !,-,#,etc.
* Case sensitive. Example: abc, ABC.(both are different Variables)

***Operator***

**Arithmetic operators :** +, -, \*, / , Modulo(%), Exponential(\*\*)

*Exponential(\*\*)*

let a = 2;

let b = 5;

let abc = a\*\*b; // 2x2x2x2x2 = 32

document.write(abc);

----

**Asssignment operators:** =,+=,-=,\*=,/=,%=, Exponential equal to(\*\*=)

*Exponential equal to(\*\*=)*

let abc = 2;

abc \*\*= 5; // 2x2x2x2x2 = 32

document.write(abc);

----

**Increment / Decrement operators:** ++,--

-----

**Comparison operators:** ==,!=,<,>,<=,>=

-----

**Logical operators:** &&,||,!

------

**typeof operators**

* It tells us the operand type.

let abc =2;

document.write(typeof abc);

op-> number

----

***Conditional Statements and Loops in javascript***

* **if**
* **if else**
* **if-elseif-else**
* **switch**

----

***Types of Loops***

* **while loop**
* **do while loop**
* **for**
* **for-of-loop**

*It is exclusively used for arrays.*

*It iterates through all the elements of an array.*

***Example***

const arr = [2,4,6,8,10];

for (let element of arr)

{

document.write(element);

}

* **for-in-loop**
* It is exclusively used for objects in javascript.
* It iterates through all the properties in an object.
* ***Example***

class Car {

constructor(make, model, year) {

this.make = make;

this.model = model;

this.year = year;

}

display() {

document.write(this.make + this.model + this.year);

};

};

let car1 = new Car("Hyundai", "i20", 2014);

let car2 = new Car("Honda", "city", 2005);

let car3 = new Car("Tata", "Nano", 2012);

for(let prop in car1) {

document.write(prop + ":" + car1[prop] +"<br>");

}

* **foreach**
* It is used as a loop but not a loop.
* It is a function.
* It is used to iterate through arrays.

***Example***

const arr = [2,4,6,8,10];

arr.forEach(function(element) {

document.write(element+"");

})

-----

***function***

* It is a block of code that performs a specific task.
* It takes data to perform tasks called 'function arguments'
* It can take any number of arguments.
* It retruns only one value at a time.

---

*1. General fucntions*

*2. Anonymous fucnitons*

*3. Arrow fucnction*

---

Example:

function get\_max(a,b) {

if(a>b)

{

return a;

}

else

{

return b;

}

}

let max = get\_max(5,10); //function calling

document.write(max); //as function arguments

-----

*Q. Can we assign default values to the function argruments?*

ans--> YES!

-----

**Call by value/reference**

* It allows us to pass arguments to a function.
* In JS, we can pass arguments to a function using only call by value.

-----

***Summary***

* For arrays and objects. if we change the argument variable itself, the original variable remains unaffected.
* If we change the internals of the argument variable, the original variable also gets changed.
* The concept of call by value is the same for objects as well.

------

**Anonumous fucntions**

* These are the functions that do not have any name.

***Example***

They are mainly used in objects.

let multiply = function (x,y) {

let p = x\*y;

return p;

};

let r = multiply(2,4);

document.write(r);

----

**Function as arguments**

* Function can be passed as an argument.
* The function passed to a function is called 'callback'.

***Example***

*//Anonymous functions*

let multiply = function (x,y)

{

let p = x\*y;

return p;

};

*//Anonymous function*

let add = function(x,y){

let s = x+y;

return s;

};

let a=5, b=20;

//General fuction

function action(x,y,z){

let r = z (x,y);

return r;

};

let r= action(a, b, add);

document.write(r);

// OUTPUT -> 25

------

**Arrow function**

let multiply = (x,y) => {

let p = x\*y;

retrun p;

}

*Arrow function depends on two conditions:*

1) It can have alphanumeric character or hyphen or underscore.

let multiply = (x,y) => {

return x\*y;

}

**--> let multiply = (x,y) => x\*y;**

2) Function has only one argument.

**--> let multiply = x => x\*y;**

==================================================

***Object***

* It is a data type.
* It is a collection of variables.
* ***Example***

let **dog** = { //dog is a object

breed: 'Golden retriever', //Properties

height: '4ft',

age: 2,

display: function() { //Inside the object function we called methods also.

document.write(this.breed + this.height+ this.age);

}

};

-----

**Dot (.) operator**

* It is used to access the property or method of an object.
* It helps to assign a new value to an existing property.
* It helps to define a new property.
* ***Example***

document.write(dog.breed);

document.write(dog.height);

document.write(dog.age);

-----

**Square [] brackets**

* It is used to access the properties of an object.
* It helps us to edit an existing property.
* It helps us to define a new property.
* ***Example***

document.write(dog['breed']);

document.write(dog['age']);

-----

***Ways to create an object***

***1) Using object initializer***

***2) Using new Object()***

***3) Using function constructor***

***4) Classes***

***-------***

***Creating an object***

**new Object()**

let dog = new Object(); // or let dog = { };

dog.breed = "Golden Retriever";

dog.height = "4ft";

dog.age = 2;

dog.display = function() {

    document.write(this.breed + " " + this.height + " " + this.age);

};

dog.display();

//OUTPUT **->** Golden Retriever 4ft 2

-----

***A function constructor.***

* It is a normal function used to create an object.
* ***Example***

**function Car(make, model, year) {**

**this.make = make;**

**this.model = model;**

**this.year = year;**

**this.display = function() {**

**document.write(this.make +this.model + this.year);**

**};**

**}**

let car1 = new Car("Hyundai","i20", 2014);

let car2 = new Car("Honda","city", 2020);

let car3 = new Car("Tata","Nano", 2024);

car1.display();

car2.display();

car3.display();

-----

***Classes***

class Car {

    constructor(make, model, year) {

        this.make = make;

        this.model = model;

        this.year = year;

    }

    display() {

        console.log(this.make + this.model + this.year);

    }

}

// Creating an object

let myCar = new Car("Toyota ", "Corolla ", 2022);

// Calling the display method

myCar.display();

*Why do we need classes to create an object if the syntax is similar to that of a function constructor?*

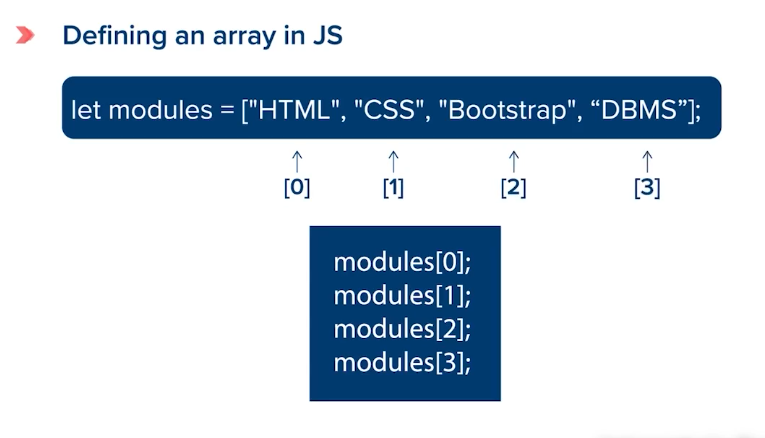
* *It offers a convenient systax.*
* *It allows us to declare the function outside the constructor.*

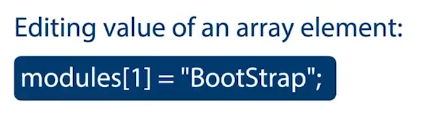
------

***What is an array?***

It is a special type of variable that can store multiple values, provided they are of the same type.

**Let array\_name = [item1, item2, …];**





***Arrays are objects***

Let modules = new Array(“HTML”, “Bootstrap”, “DBMS”, “PHP”);

***Array methods***

* **push()**

It is used to add an element at the end of an array.

* **pop()**

It fetches the last element of the array and then removes it from the array.

* **foreach()**

It is used as a loop but is not a loop.

It is a function/method used to iterate through arrays.

Const arr = [2,4,6,8,10];

arr.forEach(function(element) {  
 document.write(element + “ ”);

});

***OR***

const arr= [1,3,5,7,9];

let printElement = function(element) {

document.write(element + " ");

};

arr.forEach(printElement);

* **map()**

It is specific to array Objects

It runs on an array and creates a new array.

let numbers = [1,2,3,4,5];

let squares = numbers.map(x => x\*x);

squares.forEach(function(element){

document.write(element + " "); //1 4 9 16 25

});

* **filter()**

It is used to create a filter on the elements of an array based on the condition provided by the function.

It creates a new array and assigns the returned values by the filter() to this array.

let number = [1,2,3,4,5];

let odd\_number = number.filter(x => x%2 !=0 );

odd\_number.forEach(function(element){

    document.write(element+ " "); //1, 3, 5

});

Next video => M6T6V3