Javascript

What is javascript

Scripting language for the web/web browser

Why to Javascript

- Dynamate
- User Interaction on Client-Side
- Back-end logics on Server-side
- Interpreted and Compiled
- Large community

Scope

- Var - Function Scope, Global Initialiser. Eg: page_id,
- Let - Block Scope, Can be reassigned. Eg: value assignments like a,b,x,y,z,i,j
- Const - Block Scope, Cannot be reassigned. Eg: Name

Output Viewer

- alert
- console

Datatype

- String Eg:anything between quotes
- Number
- Boolean
- Undefined - variable declared but value is not assigned.
- Null - absence of value
- Object

Objects/Array

Object :

```
  {
  label: 'Testimony',
  value: 'testimony',
  children: [
    { label: 'TV', value: 'TV' },
    { label: 'Phone', value: 'phone' },
    { label: 'Meeting', value: 'meeting' },
    { label: 'Other Testimony', value: 'other_testimony' }
  ]
},
```

Data Structures

- Map
- Set

Map

- Collection of key-value pairs
- Maintains insertion order
- Methods - set(), get(), has(), delete(), and clear()
- Used for multiple data type values

Set

- Stores unique values
- Does not maintain insertion order
- Methods - add(), has(), delete(), and clear()
- Used for single data type values

Function

function functionName() {

//execution code

}

1.Term - function

2.Function name

3.parameter

4.code syntax

Return

When JavaScript reaches a return statement, the function will stop executing.

Functions often compute a return value. The return value is "returned" back to the "caller"

```
let x = multiple(4, 3);

function multiple(a, b) {
  return a * b;
}
```

Basic JS methods generally used

length()

toLowerCase()

toUpperCase()

split()

slice()

indexOf()

sort()

reverse()

includes()

filter()

Dates

new Date()

new Date(date string)

new Date(year,month)

new Date(year,month,day)

new Date(year,month,day,hours)

new Date(year,month,day,hours,minutes)

new Date(year,month,day,hours,minutes,seconds)

new Date(year,month,day,hours,minutes,seconds,ms)

new Date(milliseconds)

Get Dates

getFullYear()	Get year as a four digit number (yyyy)
getMonth()	Get month as a number (0-11)
getDate()	Get day as a number (1-31)
getDay()	Get weekday as a number (0-6)
getHours()	Get hour (0-23)
getMinutes()	Get minute (0-59)
getSeconds()	Get second (0-59)
getMilliseconds()	Get millisecond (0-999)
getTime()	Get time (milliseconds since January 1, 1970)

Loops

- `for` - loops through a block of code a number of times
- `for/in` - loops through the properties of an object
- `for/of` - loops through the values of an iterable object
- `while` - loops through a block of code while a specified condition is true
- `do/while` - also loops through a block of code while a specified condition is true

Loops - for

```
for (let i = 0; i < 5; i++) {
  text += "The number is " + i + "<br>";
}
```

Loops - for in (Object Iterator)

```
let person = {
    firstName: "Jesus",
    lastName: "Redeems",
   location: “Nalumavadi”
};
```

```
let text= "";
let i;
for (i in txt) {
    text += txt[i];
    console.log(text);
}
```

Loops - for of (Array/Map Interator)

```
let person = [
    "Jesus",
" Redeems",
 “Nalumavadi”
];
```

```
let text= "";
let i;
for (i of txt) {
    text += i;
    console.log(text);
}
```

Typeof

```
typeof "John"          // Returns string
typeof ("John"+"Doe") // Returns string
typeof 3.14            // Returns number
typeof 33              // Returns number
typeof (33 + 66)       // Returns number
```

```
typeof true            // Returns boolean
typeof false           // Returns boolean
typeof 1234n           // Returns bigint
typeof Symbol()        // Returns symbol
typeof x               // Returns undefined
```

Errors

Try

Catch

Finally

Threw

Use strict

```
"use strict"; to secure the declared variables to modify
```

Arrow Function

to write shorter function syntax (single parameter functions)

```
let myFunction = (a, b) => a * b;

function myFunction(a,b)

{

return a*b;

}
```

Modules

Block of code to be inserted/referred to the code

Javascript object vs JSON object

Javascript

```
const person = {
  name: "John",
  age: 30,
  isStudent: false,
  greet: function () {
    console.log("Hello!");
  }
};
```

JSON

```
{
  "name": "John",
  "age": 30,
  "isStudent": false
}
```

JSON to javascript convertor

JSON.parse()- converts to javascript
JSON.stringyfy - converts to JSON

Loose vs Tight Coupling

type

==

===