

24<sup>th</sup> Sept 2021  
(Friday)

LECTURE-22  
(WEB-DEVELOPMENT)  
(CLASS-3)

JAVASCRIPT

To do list:

- first experience with functions

- ① First experience with arrays
- ① " " " dependencies (external module)
- ① " " " file manipulation
- ① " " " callbacks
- ① " " " timer
- ① " " " downloading data from web
- ① " " " writing data in excel file
- ① " " " at writing pdfs

### ★ first experience with functions

★ (file name - first function.js)★

```

- let clargs = process.argv; // (similar to scanner in java)
- let n = parseInt(clargs[2]); // * Number *
-
- for (let i = 2; i <= n; i++) {
-   let isPrime = isPrime(i); // function is called
-   if (isPrime == true) {
-     console.log(i);
-   }
- }
-
- function isPrime(x) { // function is created
-   let isPrime = true;
-   for (let div = 2; div * div <= x; div++) {
-     if (x % div == 0) {
-       isPrime = false;
-       break;
-     }
-   }
-   return isPrime;
- }

```



## ★ first experience with arrays

★ (file name - firstArray.js) ★

node, firstArray.js 5 10 20 30 40 50  
0 1 2 3 4 5 6 7

- let clargs = process.argv;
- let n = parseInt(clargs[2]); ← // number conversion;
- let arr = []; ← // Array creation  
(just pass square brackets)  
No need of passing size.
- for(let i=0; i<n; i++){
- let val = parseInt(clargs[i+3]); ← // val variable is created  
and it will get the  
input from 3rd place.
- arr.push(val); ← // pushing the value,  
val in the array
- OR arr[i] = val;
- } console.log(arr[i]);
- console.log(arr.length);
- }

SAME  
THING

Now,

Run this by typing this condition on the terminal  
i.e. TERMINAL

```
node firstArray.js 5 10 20 30 40 50
10
20
30
40
50
[ 10, 20, 30, 40, 50 ]
5
```

★  
Single  
line  
array  
view

NOTE:

Node is a process that runs the  
JavaScript file.

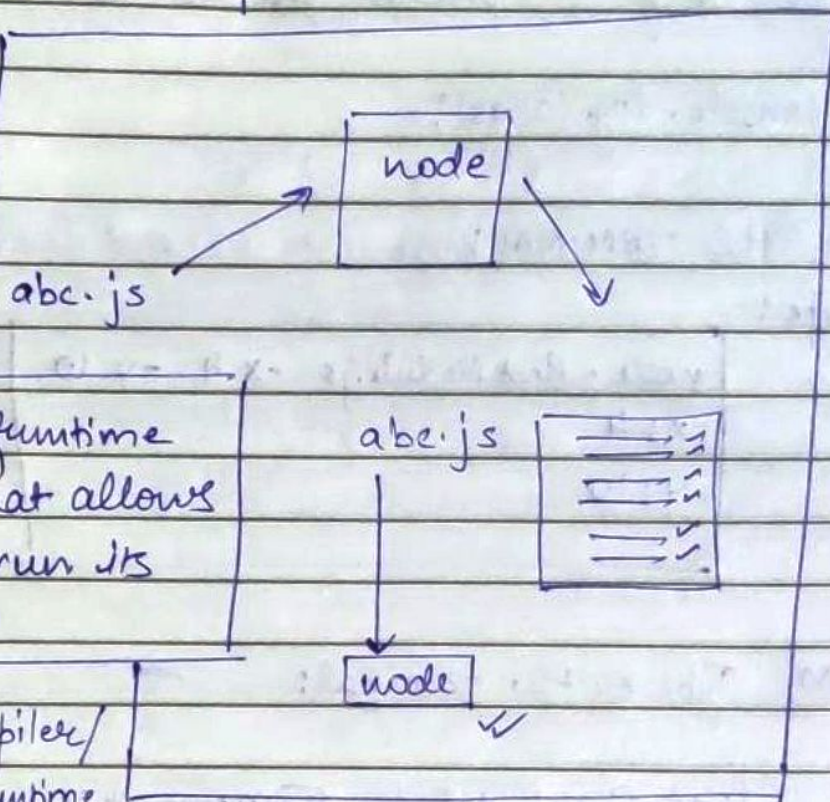
(Similar to  
arraylist) ★



## ★ First experience with dependencies (external module)

[Libraries वो चीज़ होती हैं जिसे कभी इंटाल नहीं होता, बस Use करना होता है।]

★ Node  
JavaScript  
file को चलाता  
है। ★



- Node is a compiler/interpreter/runtime environment (because it has libraries init).

## # NPM: Node package Manager

All the libraries in the node is present in the NPM.  
NPM is a downloader of library/libraries.

### INSTALL A LIBRARY

To install a library:

— on the terminal; type;

example: "npm install minimist" [minimist → to read command line  
Library]

Syntax: "npm install libraries-name"



\* To include an installed library in a javascript file/program.

```
let VARIABLE parser = KEYWORD require('minimist');  
let args = parser(process.argv);  
console.log(args);
```

Installed library  
// parser has  
minimist.

→ ON THE TERMINAL:

Type:

```
node . FirstModule.js -x 4 -y 10  
x: 4  
y: 10
```

// Beautiful and  
organised way  
of printing  
results.

Now, Type on the terminal:

```
node FirstModule.js --name="Apaak" --age=22  
name: Apaak  
age: 22
```

// This represents args.name  
and args.age respectively.

Now,

FirstModule.js.

```
let parser = require('minimist');  
let args = parser(process.argv);  
let name = args.name;  
let age = args.age;  
if (age > 30) {
```



```
console.log("Hello" + name + ". You must go Home.");
} else {
  console.log("Heya" + name + ". Where is the party tonight?");
}
```

### TERMINAL:

```
node firstModule.js --name="Sumcet Malik" --age=34
Hello Sumcet Malik. You must go Home.
node firstModule.js --name="Aparak Garg" --age=22
Hello Aparak Garg. Where is the party tonight?
```

### ★ First experience with file manipulation

// read a file, capitalize every word, write the file  
firstFiles.js [file name]

- let minimist = require("minimist");
- let args = minimist(process.argv);

→ // node firstFiles.js --source=f1.txt --dest=f2.txt  
(On the command line) // args.source (f1.txt) and args.dest (f2.txt)

- let fs = require("fs");
- [ // filesystem = fs  
pre installed in node ]

/\* Create f1.txt in the folder;  
Enter some text to it. \*/

- let stext = fs.readFileSync(args.source, "utf-8");
- console.log(stext);

↓  
to convert  
into text form.

```

- let words = stext.split(" "); // conversion of Textfile to array
- console.log(words);
- for (let i = 0; i < words.length; i++) { // CAPITALIZE EVERY WORD
  words[i] = words[i].toUpperCase();
- }

- let dtext = words.join(" "); // conversion of text file back from array
- fs.writeFileSync(args.dest, dtext, "utf-8");

```