

# Lecture -2 [Development]

Activity : Web scrapping.

- 2.1 Read data from a source : coincito world cup 2019
- 2.2 Process data : Get all teams
- 2.3 Write processed data in excel: Match result per team in their own sheet
- 2.4 Create folders : One for each teams
- 2.5 Write file: PDF file for record of each match in relevant folder.

Agenda →

- 1.1 First experience with functions
- 1.2 arrays
- 1.3 | First dependencies (external modules)
- 1.4 | experience with file manipulation
- 1.5 | callbacks
- 1.6 | timers
- 1.7 | downloading data from web
- 1.8 | processing data from web
- 1.9 | writing data in excel files
- 2.0 | writing PDFs.

1.1 firstfunction.js : →

let clargs = process.argv; // similar to scanner.

let n = parseInt(clargs[2]);

100

for (let i = 2; i < n; i++) { }

let isPrime = IsPrime(i);  
if (isPrime == true) {  
 console.log(i); } }

100 tabs

be aware

prime number

to be prime

function IsPrime(x) { }

let isPrime = true; // pehle main liya that it is true  
for (let div = 2; div \* div <= x; div++) {  
 if (x % div == 0) {  
 isPrime = false;  
 break; } }

3

return isPrime;

1

## 12: FIRST EXPERIENCE WITH ARRAY:

let clargs = process.argv;

let n = parseInt(clargs[2]);

int arr → save  
strictly like this.

No need to specify arr size here.

```

for (let i = 0; i < n; i++) {
    let val = parseInt(clargs[i + 3]);
    arr.push(val);
}

```

3

## Day run →

command line arguments:

size

node firstarray.js 5 10 20 30 40 50  
 ↗ 0 ↗ 1 ↗ 2 ↗ 3 ↗ 4 ↗ 5 ↗ 6 ↗ 7  
 clargs[0] clargs[1] clargs[2]

let clargs = process.argv  
 clargs ↗

store the command line argument.

now we put 'n' at '5' or clargs[2] as  
 for the way we took the input, it considers  
 the size and will parse it as int (parseInt)

next we defined an array

let arr = [];

for loop  
~~i = 0 to n~~ total - 0 1 2 3 4 - ; n  
 ↑ time ity loop will

let val = parseInt(clargs[i + 3]),  $\approx n$ .

why if 3 as i + 2 we have  
 size, from the next index we have the actual  
 elements.

`arr.push(charge[i+3]);`  
`or arr[i] = charge[i+3]`

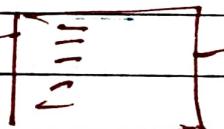
command line par jo  $i+3$  index  
 ke shuru hoga / pada hoga, vo humare  
 array ka first element hoga.

↓

The input on the screen is now stored in `arr`.

```
for (let i=0; i<arr.length; i++) {
  console.log(arr[i]);
}
console.log(arr);
console.log(arr.length);
```

node abc.js



then connect to  
n commands.

script

Node mere abc wale ~~log~~ ko chala de

If it's basically a run time environment.  
 platform

Ans - s → L1 Read  
 - L2 Write  
 → L3 URL to DB

, etc

[node] → abc.js

consist of library which are running on javascript file.

What is node package manager (Npm) ?

- There are libraries already existent that can do the work we want to do can facilitate it's name.
- Those libraries are registered in npm repository.

Usage: npm install Library-name.

kind of downloader of the library in its repository.

Usage: →

• npm install minimist

↳ library name

Flow to read the library: →

- ↳ go records may spend hrs
- Read "usage of it's element" in the command line.

Usage: iss variable ke order minimist library - kudi ei hui hai.

let parser = require('minimist'); // import parser  
parser aagaya main pass.

let args = parser(process.argv);

console.log(args)

## Command Line :-

1. node firstModule.js -n 4 -y 10

This is the way of usage of the minimist to parse.

O/P :-

{  
 - : [ c:: ————— node.exe path  
 c:: ————— js file path ] } O/P  
 format.  
 x: 4;  
 y: 10;

2. node firstModule.js --name="Sumeet Malih" --age=34

O/P :-

{  
 - : [ c:: ————— node.exe path  
 c:: ————— js file path ] }

name = Sumeet Malih

age = 34

3

\* can use args to store object :-

if (args.age > 30) {

console.log("Hello " + args.name + ", You must go home.")

4

else {

or we can use the variable name or which we  
 console.log ("Holla " + args.name + ", where is the  
 party tonight ? ");

}

1 out of 13 libra

stored

args.name.

\* In minimist we create -- & -- as per  
 our req., think not much about it.

No need to use  
 equal to

need to use = (equal  
 to )

\* Globally install karna padega, toh ulo  
 har baat install karne hi jagah require kar  
 dungi. Folder wine install hoti hai library.

#### 1.4 file Manipulation:-

// read a file, capitalize every word, write the file

let minimist = require ("minimist");

let args = minimist (process.argv);

command use

// node firstfile.js --source = f1.txt --dest = f2.txt

let fs = require ("fs");

It is a library that node already  
 contains.

// extext = fs.readFileSync (args.source, "utf-8");

Date \_\_\_\_\_

```
let word = stext.split(" "); // string has split.
```

```
for (let i=0; i< word.length; i++) {  
    words[i] = word[i].toUpperCase();  
}
```

```
let dtext = words.join(" "); // array has join  
fs.writeFileSync(args.dest, dtext, "utf-8");
```

- utf-8 - encoding, data text form mein change while reading.
- write kisi wayah se file apne kab create hogayi.
- How to directly capitalise the string "  
is, etc.