25 Sunday

Array Methods

```
const arr = ['x', 'y', 'E', 'B', 'o', 'A']

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

arr.length — only applicable in array.

arr.sort() { sorted in alphabetical order }

arr.reverse — to reverse the array.

26 Monday

To add an element main array

arr[6] = 'H'   added as new element

arr[4] = 'h'   the 4th element replaced by 'h'

To add 'h'

The last position can't be predicted manually use

arr.push('H') — new element added as last element.

To remove last element

arr [7]

27 Tuesday

arr.pop () — To remove the last element from the array

const rem = arr.Pop()

shift — unshift method

To add or remove elements from front end and backend.

arr.unshift() — to element in front end. (1st position)

arr.shift() to remove from first element.

28 Wednesday

const arr22 [1, 3, 5, 4, 6, 8, 5]

const arr3 = [3, 5]

→ To mix and create a new array using all 3 array use concatination.

const newArr = arr.concat (arr 2)

const newArr = arr.concat (arr2, arr3)

arr.join() — elements of array become string

29 Thursday

The joining or the position between each
element can be put inside the bracket.
arr.join(x)   arr.join(-)

To check the elements presence.

arr.includes('A')

arr.indexOf() — search from 1st position

30 Friday

arr.lastIndexOf - search from last position.

Slice

To cut a portion of array.

eg:- [A, B, C, D, E, F, G]

slice(0,3) = A B C

slice(2,6) = CDEFG

---

01 Saturday

MAY DAY   arr.slice(0)

[A, B, C, D, E, F, G] This is also
 0 1 2 3 4 5 6   possible in
-7 -6 -5 -4 -3 -2 -1   slice.

arr.slice(1,3)   This method does not change
                   the main array @

arr.slice(0)

Slice

To add an element in particular position

Syntax.

Splice (Index, How many delete, newelements)
         ↓
       starting
       position

arr.splice(1,2)

arr.splice(1,0, 'A','C')

Array.isArray(A) — To check is it an array or
                    not output will be the
                    from of true or false.

02 Sunday

Monday

'To find the last element of an array

arr. it (-1)

arr. Fil (-1) all the elements of array changes to the new element or replace the old element.

Tuesday 04

For Each ( )

forEach ArrayName. foreEach (function)

For .

arr. forEach (display)

function display (ele, index)
{
console. Log (ele, index)
}

3

const arr2. map (() => {

const newArr = arr2. map (() => {

Wednesday 05

const newArr = void [ ] all element from each element. ie,

const newArr = arr2. map ((data) => {
  return {
    size of array
    data xx 3   is same.
  }
  33
})

console. log (newArr)

Filter

const newArr = arr2. Filter ((element, index) => {
  return {
    3
  }
})

(array size gets reduced any gets the result in array)

{
  if (element q 2 == -0) {
    return element string
  }
  3
}

In filter operations can't be possible. only sorting can be possible)

condition checking only possible.

Or

const newArr = arr2. filter ((element, index)
=> element % 2 == -0)

Thursday 06

Reduce

```
const newdata = arr2.reduce((l))          Output
                                          value
const newdata = arr2.reduce((acc,ele)=>{  reduced
                                          to single
    return acc + ele                      value.
3)
                        acc - accumulator

        OR

const newdata = arr2.reduce((acc,ele) => acc + = ele)
```

Some

```
const check = arr2.some((ele)=> {      Some can be
                                       done with
    return ele > 7                     Atlee but
                                       op in arr
3)                                     form
```

Every

```
const check = arr2.every((ele)=> {     all element should
                                       satisfy the
    return ele >=1                     condition.
3)
```

Find index

```
const check = arr2.findIndex((ele)=> {3})
```

Find

```
const check = arr2.find((ele)=> (return ele >5)

const check = arr2.Find((ele)=> {       element
                                        returns
    return ele>6                        1st condition
3)                                      satisfies.
```

String methods

```
const str = 'hi welcome to js module hi below'

                         totally 3 ele
                         including space    str.length

         To change to uppercase

                         str.toLowercase()

str.toUpperCase()

str.includes(``)   ——— to check particular element

str.startsWith(``)
str.endsWith(``)
```

May 2021

11 Tuesday

Str. Search (' ') — — — o/p is in index.

Str. match (/g/g)   g - means globally

Then Str. index OF (' ')   if the element is missing
then o/p is -1

eg:- Str. eplace ('JS', "Javascript")

Str. eplaceAll ('hi', "Javascript")
OR
Str. eplace (/hi/g, "Javascript")

12 Wednesday

Str. trim()   — eleminate the starting and ending
spacing.

Str [5]

Str. charAt (5)

Str. concat ('new string', '***')

Str. split (' ') → o/p as array

Str. split (hi')

Str. split (Str) join (' ') → array chars
to from string

May
S	M	T	W	T	F	S
	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

13 Thursday

Str. slice (4, 7)   -ve value possible

Str. subString (2,7) → no -ve value possible
relevant

Str. Substr (2, 4)
second patron determines no. of elements eg moved
in slice substring etc 2nd position determines the index.

→ Let a = 3000

a. to string ()   — converted into string

Date & Time

current date. —
new Date() → can be put date mode.
using – yyy, m, d, h, m, sec, msec

new Date (11, 2, 3, 4, 5, 7, 8, 8)   new Date (). setDate()

new Date(4). getFullYear
new Date(). get UTC Month
new Date(). get Date
new Date() → get minute mints
new Date () → get Seconds
new Date () → get millirecords
new Date(). get real Time ms

May
S	M	T	W	T	F	S
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					