TRAINING REPORT (TR-102)

DAY: -19 (3 JULY 2024)

Introduction about JavaScript Array: -

An array can hold many values under a single name, and you can access the values by referring to an index number.

An array is a special variable, which can hold more than one value.

EXAMPLE: -

```javascript
1  let NAME=["GURSHARAN KAUR",20,20000, '7723837266']
2  console.log("Orignal list :", NAME);
3  console.log("Length of list:" ,NAME.length);
4
```

OUTPUT: -

```
node /tmp/3yLUM3p0a7.js
Orignal list : [ 'GURSHARAN KAUR', 20, 20000, '7723837266' ]
Length of list: 4
```

Creating an Array: - Using an array literal is the easiest way to create a JavaScript Array.

Syntax: - const array_name = [item1, item2, ...];

Accessing Array Elements: - You access an array element by referring to the **index number**:

EXAMPLE: -

```javascript
1  let NAME=["GURSHARAN KAUR",20,20000, '7723837266']
2  console.log("Orignal list :", NAME);
3  console.log("Length of list:" ,NAME.length);
4  console.log(NAME[0]);
```

OUTPUT: -

```
node /tmp/ejjAXfjVYt.js
Orignal list : [ 'GURSHARAN KAUR', 20, 20000, '7723837266' ]
Length of list: 4
GURSHARAN KAUR
```

GURSHARAN KAUR   **URN: -2203441**   **CRN: -2215052**

Changing an Array Element: -

EXAMPLE: -

```
1  let NAME=["GURSHARAN KAUR",20,20000, '7723837266']
2  console.log("Orignal list :", NAME);
3  console.log("Length of list:" ,NAME.length);
4  console.log(NAME[0]);
5  NAME [1]=21;
6  console.log("Orignal list :", NAME);
```

OUTPUT: -

```
node /tmp/YnRpppR0fK.js
Orignal list : [ 'GURSHARAN KAUR', 20, 20000, '7723837266' ]
Length of list: 4
GURSHARAN KAUR
Orignal list : [ 'GURSHARAN KAUR', 21, 20000, '7723837266' ]
```

Converting an Array to a String: - The JavaScript method toString() converts an array to a string of (comma separated) array values.

EXAMPLE: -

```
1  let NAME=["GURSHARAN KAUR",20,20000, '7723837266']
2  console.log("Orignal list :", NAME);
3  console.log("Length of list:" ,NAME.length);
4  console.log(NAME[0]);
5  NAME [1]=21;
6  console.log("Orignal list :", NAME);
7  console.log("String:", NAME.toString());
```

OUTPUT: -

```
node /tmp/B3b6rZW8vA.js
Orignal list : [ 'GURSHARAN KAUR', 20, 20000, '7723837266' ]
Length of list: 4
GURSHARAN KAUR
Orignal list : [ 'GURSHARAN KAUR', 21, 20000, '7723837266' ]
String: GURSHARAN KAUR,21,20000,7723837266
```

GURSHARAN KAUR   **URN: -2203441**   **CRN: -2215052**

<u>Arrays are Objects: -</u> Arrays are a special type of objects. The typeof operator in JavaScript returns "object" for arrays.
But, JavaScript arrays are best described as arrays.
Arrays use **numbers** to access its "elements".
Objects use names **to access its "members".**

<u>Array Properties and Methods: -</u> The real strength of JavaScript arrays are the built-in array properties and methods:

- The length Property: The length property of an array returns the length of an array (the number of array elements).

- Accessing the First Array Element

- Accessing the Last Array Element

- Adding Array Elements: The easiest way to add a new element to an array is using the push () method. New element can also be added to an array using the length property.

<u>EXAMPLE: -</u>

```
1   let NAME=["GURSHARAN KAUR",20,20000, '7723837266']
2   console.log("Orignal list :", NAME);
3   console.log("Length of list:" ,NAME.length);
4   console.log(NAME[0]);
5   NAME [1]=21;
6   console.log("Orignal list :", NAME);
7   console.log("String:", NAME.toString());
8   console.log("After pop :", NAME.pop());
9   console.log("Orignal list :", NAME);
10  console.log("Push the element:", NAME.push("Btech,"));
11  console.log("Orignal list :", NAME);
12  console.log("String:", NAME.toString());
13  console.log("String:", NAME.join("/"));
14  let branchesinbtech=["CSE","CIVIL","IT","MECHANICAL","ELECTRICAL"
        ,"ELECTRONICS"]
15  console.log("COMBINE :",NAME+branchesinbtech );
16  console.log("Orignal list :", [NAME + branchesinbtech]);
17  console.log("Orignal list :", [NAME]+[branchesinbtech]);
```

OUTPUT: -

```
node /tmp/IYpW2EmIQn.js
Orignal list : [ 'GURSHARAN KAUR', 20, 20000, '7723837266' ]
Length of list: 4                        4 | Page
GURSHARAN KAUR
Orignal list : [ 'GURSHARAN KAUR', 21, 20000, '7723837266' ]
String: GURSHARAN KAUR,21,20000,7723837266
After pop : 7723837266
Orignal list : [ 'GURSHARAN KAUR', 21, 20000 ]
Push the element: 4
Orignal list : [ 'GURSHARAN KAUR', 21, 20000, 'Btech,' ]
String: GURSHARAN KAUR,21,20000,Btech,
String: GURSHARAN KAUR/21/20000/Btech,
COMBINE : GURSHARAN KAUR,21,20000,Btech,CSE,CIVIL,IT,MECHANICAL,ELECTRICAL
    ,ELECTRONICS
Orignal list : [
  'GURSHARAN KAUR,21,20000,Btech,CSE,CIVIL,IT,MECHANICAL,ELECTRICAL
      ,ELECTRONICS'
]
Orignal list : GURSHARAN KAUR,21,20000,Btech,CSE,CIVIL,IT,MECHANICAL
    ,ELECTRICAL,ELECTRONICS
```

GURSHARAN KAUR   **URN: -2203441**   **CRN: -2215052**