Array Items – reduce method

```
let marks = [40,60,80,40]
    let sum = marks.reduce((total,value)=>{
        return total + value
    })
    console.log(sum)
```

Array Items – example

let score=[5,6,3]

let result = score.map()
1. result will be an array
2. result will have 3 elements
3. result will have same or modified values

let result = score.filter()
1. result will be an array
2. result will have 3 or less than 3 elements
3. result will have same values

let result = score.find()
1. result will be a value
2. result will have single value
3. result will have 5 or 6 or 3

let result = score.reduce()
1. result will be a value
2. result will have single value
3. result will have total of all the values

Array Methods

```
let fruits = ["Apple", "Orange", "Mango", "A"];
     // let fruits = new Array("Apple", "Orange", "Mango");
     console.log(fruits[0])
     console.log(fruits.length)
     console.log(fruits.at(-1)) //  or fuits[fruits.length-1]
     fruits.push("cherry","banana") //appends in the end – can add multiple
     console.log(fruits)
     fruits.pop(); // removes last element
     console.log(fruits);
     fruits.shift() // removes first element
     console.log(fruits)
     fruits.unshift("Plum","Pears") //adds in the beginning – can add multiple
     console.log(fruits)
     console.log(fruits.reverse())
     fruits.sort() //for desc - sort and reverse
     console.log(fruits)
     let num = [4,17,3,5]
     console.log(num.sort(function(a, b){return a-b})) // for ascending
     console.log(num.sort(function(a, b){return b-a})) // for ascending
   console.log(fruits.indexOf("Apple"))
   console.log(fruits.lastIndexOf("Apple")) //if apple appears multiple times
console.log(fruits.includes("Apple"))
     // push/pop run fast, while shift/unshift run slow due to rearrangements
 let num = [4, 17, 3, 5];
     //   delete num[0];
     //   console.log(num);
     num.splice(0, 2);
     console.log(num);
slice
```

Array of Objects

```
<script>
    const products = [
      { name: "Product 1", price: 300 },
      { name: "Product 2", price: 100 },
      { name: "Product 3", price: 500 },
    ];
    const cart = [];
    let item = products[0];
    item.quantity = 2;
    item.total = item.quantity * item.price;
    cart.push(item);
    item = products[2];
    item.quantity = 3;
    item.total = item.quantity * item.price;
    cart.push(item);
    console.log(cart);
    let orderValue = cart.reduce((sum, value) => {
      return sum + value.total;
    },0);
    console.log("Order Value is", orderValue);
  </script>
```