

## Assignment-4

### (MERN Stack)

**Ques 1.** Create an Array of Salaries and find the following:

- a) Sum of Salaries greater than 10000.
- b) Max Salary from the Array.
- c) Count Those Salaries greater than 10000.

**Sol.**

```
var Salary = [12000, 9000, 8500, 15000, 7000, 20000, 21000];  
console.log("Salary: ["+Salary+"]");
```

```
var SumSalary = Salary.reduce((sum, sal) => (sal>10000)?sum+sal:sum+0,0);  
console.log("Sum of the Salary > 10000: "+SumSalary);
```

```
var MaxSalary = Salary.reduce((Max, sal) => (Max<sal)?Max=sal:Max);  
console.log("Maximum Salary: "+MaxSalary);
```

```
var Count = Salary.reduce((count, sal) => (sal>10000)?++count:count,0);  
console.log("No. of Salaries > 10000: "+Count);
```

**Output:**

```
Salary: [12000,9000,8500,15000,7000,20000,21000]  
Sum of the Salary > 10000: 68000  
Maximum Salary: 21000  
No. of Salaries > 10000: 4
```

**Ques 2.** Maintain an Array of Employees (Array of Objects) and perform the following:

- a) Sort the Employees by Name and Salary.
- b) In Employee Salaries add 10% Tax in Each Employee Salary and Store in a new Array, So don't modify the Original Array.

**Sol.**

```
var Employees = [{Name:"Akshay", Salary:12000}, {Name:"Rohit", Salary:10000},  
{Name:"Yuvraj", Salary:12500}, {Name:"Jai", Salary:8000}, {Name:"Vikas",  
Salary:15000}];
```

```
console.log("Employees sorted by Name:")
```

```
console.log(Employees.sort((a,b) => a.Name.localeCompare(b.Name)));
```

```
console.log("Employees sorted by Salary:")
```

```
console.log(Employees.sort((a,b) => a.Salary - b.Salary));
```

```
var Emp = Employees.filter(e => true);
```

```
var NewSalary = Emp.map(e => (e.Salary - e.Salary*0.1));
```

```
console.log("Updated Salary: ["+NewSalary+"]");
```

**Output:**

```
Employees sorted by Name:
```

```
[  
  { Name: 'Akshay', Salary: 12000 },  
  { Name: 'Jai', Salary: 8000 },  
  { Name: 'Rohit', Salary: 10000 },  
  { Name: 'Vikas', Salary: 15000 },  
  { Name: 'Yuvraj', Salary: 12500 }  
]
```

```
Employees sorted by Salary:
```

```
[  
  { Name: 'Jai', Salary: 8000 },  
  { Name: 'Rohit', Salary: 10000 },  
  { Name: 'Akshay', Salary: 12000 },  
  { Name: 'Yuvraj', Salary: 12500 },  
  { Name: 'Vikas', Salary: 15000 }  
]
```

```
Updated Salary: [7200,9000,10800,11250,13500]
```