Functions in JS

Block of code that performs a specific task, can be invoked whenever needed

Apna College

Functions in JS

redundancy remove

Function Definition

```
function functionName( ) {

    //do some work

}
```

return
after code not execute

```
function functionName( param1, param2 ...) {

    //do some work

}
```

Function Call

```
functionName( );
```

Apna College

Arrow Functions

Compact way of writing a function

```
const functionName = ( param1, param2 ...) => {

    //do some work

}
```

Apna College

```
const mul=(a,b)=>{
console.log(a*b);
}
```

```
const sum = ( a, b ) => {

    return a + b;

}
```

Let‘s Practice

Qs. Create a function using the “function” keyword that takes a String as an argument & returns the number of vowels in the string.

```
function contvowels(str)
{let c=0;
for(const i of str)
{
  if(i==='a'||i==='e'||i==='o'||i==='u'||i==='i')
  c++;
  console.log(i,c);
  }
  }
</script>
</head>
<body>
<script>
contvowels('abcd');
```

Qs. Create an arrow function to perform the same task.

Apna College

forEach Loop in Arrays

arr.forEach(callBackFunction)

CallbackFunction : Here, it is a function to execute for each element in the array

***A callback is a function passed as an argument to another function.**

```
let arr=[1,2,3,4,5];

arr.forEach( ( val ) => {

    console.log(val);

} )
```

print 1
2
3
4
5

```
let name=["bloo","ms","ubr","google","ibm"];
name.forEach((val)=>{
console.log(val.toUpperCase());
})
```

Apna College

Let‘s Practice

Qs. For a given array of numbers, print the square of each value using the forEach loop.

```
let arr=[1,2,3,4,5];
arr.forEach((val)=>{
console.log(val*val);
})
```

Apna College

Some More Array Methods

Map

Creates a new array with the results of some operation. The value its callback returns are used to form new array

arr.map(callbackFnx(value, index, array))

```
let newArr = arr.map( ( val ) => {

    return val * 2;

} )
```

The map() method of Array instances creates a new array populated with the results of calling a provided function on every element in the calling array.

Apna College

Some More Array Methods

The filter() method of Array instances creates a shallow copy of a portion of a given array, filtered down to just the elements from the given array that pass the test implemented by the provided function.

Filter

Creates a new array of elements that give true for a condition/filter.

Eg: all even elements

```
let newArr = arr.filter( ( ( val ) => {

    return val % 2 === 0;

} )
```

Apna College

Some More Array Methods

```
let arr=[1,2,3,4,5];
let op =arr.reduce((preval,currval)=>{
return preval>currval?preval:currval;
});
console.log(op);
```

Reduce

Performs some operations & reduces the array to a single value. It returns that single value.

Apna College

JavaScript Demo: Array.reduce()

```
1  const array1 = [1, 2, 3, 4];
2
3  // 0 + 1 + 2 + 3 + 4
4  const initialValue = 0;
5  const sumWithInitial = array1.reduce(
6    (accumulator, currentValue) => accumulator + currentValue,
7    initialValue,
8  );
9
10 console.log(sumWithInitial);
11 // Expected output: 10
```

```
let arr=[1,2,3,4,5];
let op =arr.reduce
((preval,currval)=>{
return preval+currval;

});
console.log(op);
```

Let's Practice

Qs. We are given array of marks of students. Filter our of the marks of students that scored 90+.

```
let arr=[10,200,30,140,115];
let newarr=arr.filter((val)=>{
return val>90;
});
console.log(newarr);
```

Qs. Take a number n as input from user. Create an array of numbers from 1 to n.

Use the reduce method to calculate sum of all numbers in the array.

Use the reduce method to calculate product of all numbers in the array.

```
let arr[];
let n=prompt("enter number");
for(let i=1;i<=n;i++)
arr[i-1]=i;
console.log(arr);
```

```
let arr=[1,2,3,4,5];
let newarr=arr.reduce((res,curr)=>
{
return res+curr;
});
console.log(newarr);
```

```
let arr=[1,2,3,4,5];
let newarr=arr.reduce
((res,curr)=>
{
return res*curr;
});
console.log(newarr);
```

Apna College