Functions Task

Do the below programs in anonymous function and IIFE

- **Print odd numbers in an array**

Anonymous

```
let arr = [1,2,3,4,5,6,7,8,9];
let oddarr = function (iarray)
{
    let result = iarray.filter(i => i%2!==0);
    return result;
}
console.log(oddarr(arr));
```

IIFE

```
(function ()
{
    let arr = [1,2,3,4,5,6,7,8,9];
    let result = arr.filter(i => i%2!==0);
    console.log(result);
})();
```

- **Convert all the strings to title caps in a string array**

  Anonymous

```
let strarr = ['grape', 'banana', 'mango', 'apple'];
let capstr = function (istr)
{
    let result = istr.map(item => item.substr(0,1).toUpperCase() +
item.substr(1,item.length-1));
    return result;
}
console.log(capstr(strarr));
```

  IIFE

```
let strarr = ['grape', 'banana', 'mango', 'apple'];
(function (istr)
{
    let result = istr.map(item => item.substr(0,1).toUpperCase() +
item.substr(1,item.length-1));
    console.log(result);
})(strarr);
```

- **Sum of all numbers in an array**

  Anonymous

```
let arr = [1,2,3,4,5,6,7,8,9];
let sumarr = function (iarray)
{
    let result = iarray.reduce((sum, item) => sum+item);
    return result;
}
console.log(sumarr(arr));
```

  IIFE

```
let arr = [1,2,3,4,5,6,7,8,9];
(function (iarray)
{
    let result = iarray.reduce((sum, item) => sum+item);
    console.log(result);
})(arr);
```

- **Return all the prime numbers in an array**

  Anonymous

```
let arr = [1,2,3,4,5,6,7,8,9,10,11];
let primearr = function (iarray)
{
    let result = iarray.filter(item =>
    {
        if (item < 2) return false;
        if(item ==2) return true;
        if (item%2 === 0) return false;
        for (let fac = 3; fac*fac<=item; fac += 2)
        {
            if (item%fac === 0) return false;
        }
        return true;
    });
    return result;
}
console.log(primearr(arr));
```

  IIFE

```
let arr = [1,2,3,4,5,6,7,8,9,10,11,13];
(function (iarray)
{
    let result = iarray.filter(item =>
    {
        if (item < 2) return false;
        if(item ==2) return true;
        if (item%2 === 0) return false;
        for (let fac = 3; fac*fac<=item; fac += 2)
        {
            if (item%fac === 0) return false;
        }
        return true;
    });
    console.log(result);
})(arr);
```

- **Return all the palindromes in an array**

  Anonymous

```
let str = ['sample', 'kayak', 'radar'];
let palindrome = function (iarr)
{
    let result = iarr.filter(item =>
    {
        let j = item.length-1;
        for(i=0; i<=(item.length-1)/2;i++)
        {
            if (item[i] == item[j])
            {
                flag = true;
                j--;
            }
            else
            {
                flag = false;
                break;
            }
        }
        return flag;
    })
    return result;
}
console.log(palindrome(str));
```

  IIFE

```
let str = ['sample', 'kayak', 'radar'];
(function (iarr)
{
    let result = iarr.filter(item =>
    {
        let j = item.length-1;
        for(i=0; i<=(item.length-1)/2;i++)
        {
            if (item[i] == item[j])
            {
                flag = true;
                j--;
            }
            else
            {
```

```
                flag = false;
                break;
            }
        }
        return flag;
    })
    console.log(result);
})(str)
```

- **Return median of two sorted arrays of same size**

  Anonymous

```
let arr1 = [1,5,8,15,5];
let arr2 = [2,6,50];
let medianf = function(iarr1,iaar2)
{
    let result = iarr1.concat(arr2).sort((a,b) => a-b);
    let median;
    if (result.length%2 === 0)
    {
        median = (result[(result.length/2)-1] + result[(result.length/2)]) / 2;
    }
    else
    {
        median = result[(result.length-1)/2]
    }
    return median;
}
console.log(medianf(arr1,arr2));
```

  IIFE

```
let arr1 = [1,5,8,15,5,2];
let arr2 = [2,6,50];
(function(iarr1,iaar2)
{
    let result = iarr1.concat(arr2).sort((a,b) => a-b);
    let median;
    if (result.length%2 === 0)
    {
        median = (result[(result.length/2)-1] + result[(result.length/2)]) / 2;
    }
    else
    {
        median = result[(result.length-1)/2]
    }
    console.log(median);
})(arr1,arr2);
```

Remove duplicates from an array

Anonymous

```
let arr1 = [1,2,3,1,3,5,2];
let arrdup = function (arr)
{
    let newarr = [arr[0]];
    for (let i in arr)
    {
        let flag;
        for (let j in newarr)
        {
            if (arr[i] == newarr[j])
            {
                flag = false;
                break;
            }
            else
            {
                flag = true;
            }
        }
        if(flag === true)
        {
            newarr.push(arr[i]);
        }
    }
    return newarr;
}
console.log(arrdup(arr1));
```

IIFE

```
let arr1 = [1,2,3,1,3,5,2];
(function (arr)
{
    let newarr = [arr[0]];
    for (let i in arr)
    {
        let flag;
        for (let j in newarr)
        {
            if (arr[i] == newarr[j])
            {
                flag = false;
```

```
                break;
            }
            else
            {
                flag = true;
            }
        }
        if(flag === true)
        {
            newarr.push(arr[i]);
        }
    }
    console.log(newarr);
})(arr1);
```

- **Rotate an array by k times and return the rotated array**

  Anonymous

```
let arr1 = [1,2,3,4,5,6,7];
const numrotate = 4;
let rotate = function (iarr,k=2)
{
    k = k%iarr.length;
    let temp1 = [];
    let temp2 = [];
    let newarr = [];
    for (let i=0; i<k; i++)
    {
        temp1.push(iarr[i]);
    }
    for (let j=k; j<iarr.length; j++)
    {
        temp2.push(iarr[j]);
    }
    newarr = temp2.concat(temp1);
    return newarr;
}
console.log(rotate(arr1,numrotate));
```

  IIFE

```
let arr1 = [1,2,3,4,5,6,7];
const numrotate = 4;
(function (iarr,k=2)
{
    k = k%iarr.length;
    let temp1 = [];
    let temp2 = [];
    let newarr = [];
    for (let i=0; i<k; i++)
    {
        temp1.push(iarr[i]);
    }
    for (let j=k; j<iarr.length; j++)
    {
        temp2.push(iarr[j]);
    }
    newarr = temp2.concat(temp1);
    console.log(newarr);
})(arr1,numrotate);
```