

JavaScript: Task 3

Print odd Numbers in an array?

Anonymous function:

```
var odd=function (arr) {  
    odd="";  
    for(let i=0;i<arr.length;i++)  
    {  
        if(arr[i]%2!==0)  
        {  
            odd=odd+arr[i]+" ";  
        }  
    }  
    console.log(odd)  
}  
arr=[1,2,3];  
odd(arr);
```

IIFE:

```
(function (arr) {  
    odd="";  
    for(let i=0;i<arr.length;i++)  
    {  
        if(arr[i]%2!==0)  
        {  
            odd=odd+arr[i]+" ";  
        }  
    }  
    console.log(odd)  
})([1,2,3])
```

Convert all the strings to title caps in string array

Anonymous function:

```
var capitalize=function (arr) {  
  for(let i=0;i<arr.length;i++)  
  {  
    let a=arr[i];  
    a=a.split("");  
    b=a[0].toUpperCase();  
    a=b+','+a.slice(1);  
    a=a.split(',');  
    let c="";  
    for(let i=0;i<a.length;i++)  
    {  
      c=c+a[i];  
    }  
    console.log(c)  
  }  
}  
arr=['tej','raj'];  
capitalize(arr);
```

IIFE:

```
(function (arr) {  
  for(let i=0;i<arr.length;i++)  
  {  
    let a=arr[i];  
    a=a.split("");  
    b=a[0].toUpperCase();  
    a=b+','+a.slice(1);  
    a=a.split(',');  
    let c="";  
    for(let i=0;i<a.length;i++)  
    {  
      c=c+a[i];  
    }  
    console.log(c)  
  }  
})(['tej','raj'])
```

Sum of all the numbers in the array:

Anonymous function:

```
var sum=function (arr) {  
  var sum=0;  
  for(let i=0;i<arr.length;i++)  
  {  
    sum=sum+arr[i];  
  }  
  console.log(sum)  
}  
arr=[1,2,3];  
sum(arr);
```

IIFE:

```
(function (arr) {  
  var sum=0;  
  for(let i=0;i<arr.length;i++)  
  {  
    sum=sum+arr[i];  
  }  
  console.log(sum)  
})([1,2,3])
```

Return all the prime numbers in an array

Anonymous function:

```
var prime = function (input) {  
    var num = [];  
    for (var i in input) {  
        var count = 0;  
        for (var j = 2; j<=input[i]; j++) {  
            if (input[i] % j === 0) {  
                count = count + 1;  
            }  
        }  
        if (count === 1) {  
            num.push(input[i]);  
        }  
    }  
    return console.log(num)  
}  
prime([1,2,3,4,5,6,7,8,9,10]);
```

IIFE:

```
(function (input) {  
    var num = [];  
    for (var i in input) {  
        var count = 0;  
        for (var j = 2; j<=input[i]; j++) {  
            if (input[i] % j === 0) {  
                count = count + 1;  
            }  
        }  
        if (count === 1) {  
            num.push(input[i]);  
        }  
    }  
    return console.log(num)  
})([1,2,3,4,5,6,7,8,9,10])
```

Return all the palindromes in an array

Anonymous function:

```
var palindrome=function (arr) {  
  for(let i=0;i<arr.length;i++)  
  {  
    let a=arr[i];  
    let c="";  
    let d="";  
    for(let i=0;i<a.length;i++)  
    {  
      c=c+a[i];  
    }  
    for(let i=c.length-1;i>=0;i--)  
    {  
      d=d+c[i];  
    }  
    if(c===d)  
    {  
      console.log(c)  
    }  
  }  
}  
arr=['tej','rar'];  
palindrome(arr);
```

IIFE:

```
(function (arr) {  
  for(let i=0;i<arr.length;i++)  
  {  
    let a=arr[i];  
    let c="";  
    let d="";  
    for(let i=0;i<a.length;i++)  
    {  
      c=c+a[i];  
    }  
    for(let i=c.length-1;i>=0;i--)  
    {  
      d=d+c[i];  
    }  
    if(c===d)  
    {  
      console.log(c)  
    }  
  }  
})(['tej','rar'])
```

Return median of two sorted arrays of same size

Anonymous function:

```
var median=function (arr1,arr2) {  
    var arr;  
    arr=arr1+" "+arr2;  
    arr=arr.split(" ")  
    arr=arr.sort();  
    len=arr.length/2;  
    med=(+arr[len-1]+ +arr[len])/2;  
    console.log(med);  
}  
arr1=[1,2,3,6];  
arr2=[4,5,8,7];  
median(arr1,arr2);
```

IIFE:

```
(function (arr1,arr2) {  
    var arr;  
    arr=arr1+" "+arr2;  
    arr=arr.split(" ")  
    arr=arr.sort();  
    len=arr.length/2;  
    med=(+arr[len-1]+ +arr[len])/2;  
    console.log(med);  
})([1,2,3,6],[4,5,8,7])
```

Remove duplicates from an array :

Anonymous function:

```
var arr = [1,2,8,3,6,1,2,6];
let uniqueArr = [];
var duplicates = function(){
  for(let i of arr) {
    if(uniqueArr.indexOf(i) === -1) {
      uniqueArr.push(i);
    }
  }
  console.log(uniqueArr);
}
duplicates(arr);
```

IIFE:

```
(function(){
  let uniqueArr = [];
  for(let i of arr) {
    if(uniqueArr.indexOf(i) === -1) {
      uniqueArr.push(i);
    }
  }
  console.log(uniqueArr);
})([1,2,8,3,6,1,2,6])
```


Rotate an array by k times and return the rotated array

Anonymous function:

```
var rotate = function(num, k) {  
  for (let i = 0; i < k; i++) {  
    num.unshift(num.pop());  
  }  
  return num;  
};  
console.log(rotate([1,2,3,4,5], 2));
```

IIFE:

```
(function(num, k) {  
  for (let i = 0; i < k; i++) {  
    num.unshift(num.pop());  
  }  
  return num;  
})([1,2,3,4,5], 2)
```