1.Function to return the odd numbers in an array.

**i) Using Anonymous Function**

var odd = function (a){

var c=[ ];

for (var i=0;i<=a.length;i++){

if(a[i]%2==1)

c.push(a[i]);

}

return c;

}

console.log(odd([12,14,17,19]))

**ii) Using IIFE**

(function(a){

var c=[ ];

for (var i=0;i<=a.length;i++){

if(a[i]%2==1)

c.push(a[i]);

}

console.log(c);

})([12,14,17,19]);

2. Strings to title caps

**i)Using Anonymous Function**

var titleCase = function (a) {

a = a.toLowerCase().split(' ');

for (var i = 0; i < a.length; i++) {

a[i] = a[i].charAt(0).toUpperCase() + a[i].slice(1); }

return a.join(' ');

}

console.log(titleCase("I’m learning to become Full stack developer"));

**ii)Using IIFE**

( function titleCase(a) {

a = a.toLowerCase().split(' ');

for (var i = 0; i < a.length; i++) {

a[i] = a[i].charAt(0).toUpperCase() + a[i].slice(1);

}

console.log(a.join(' '));

})("I'm learning to become Full stack developer");

3. Sum of all numbers in an array

**i)Using ANONYMOUS FUNCTION**

var sum = function (a){

var c=0;

for (var i=0;i<a.length;i++){

c+=a[i]

}

return c;

}

console.log(sum([12,14,17,19]))

**ii)Using IIFE**

(function (a){

var c=0;

for (var i=0;i<a.length;i++){

c+=a[i]

}

console.log (c);

})([12,14,17,19]);

4. Prime numbers from an array

**i)Using ANONYMOUS FUNCTION**

var num = function(a){

a = a.filter((number) => {

for (var i = 2; i <= Math.sqrt(number); i++) {

if (number % i === 0) return false;

}

return true;

});

return a

}

console.log(num([2, 3, 4, 5, 11, 7, 8, 9, 10,97,117,113,997]));

**ii)Using IIFE**

( function(a){

a = a.filter((number) => {

for (var i = 2; i <= Math.sqrt(number); i++) {

if (number % i === 0) return false;

}

return true;

});

console.log (a)

})

([2, 3, 4, 5, 11, 7, 8, 9, 10, 97, 117, 113, 997]);

5. Palindrome in a given array

**i)Using ANONYMOUS FUNCTION**

var palindrome= function(a){

var c=[];

for(var i=0;i<(a.length);i++){

var b= a[i].toString().split("")

for(var j=(b.length);j>0;j--){

if((b[j-1])==(b[b.length-j])){

if(j==1){

c.push(a[i]);

}

}else{

break;

}

}

}

return c;

}

console.log(palindrome(["guvi","mom","string","dad","dead"]));

**ii)Using IIFE**

(function(a){

var c=[];

for(var i=0;i<(a.length);i++){

var b= a[i].toString().split("")

for(var j=(b.length);j>0;j--){

if((b[j-1])==(b[b.length-j])){

if(j==1){

c.push(a[i]);

}

}else{

break;

}

}

}

console.log(c)

})(["guvi","mom","string","dad","dead"]);

6. Meridian of two sorted arrays

**i) Using ANONYMOUS FUNCTION**

var median = function (a, b, n)

{

var i = 0;

var j = 0;

var count;

var m1 = -1, m2 = -1;

for (count = 0; count <= n; count++)

{

if (i == n)

{

m1 = m2;

m2 = b[0];

break;

}

else if (j == n)

{

m1 = m2;

m2 = a[0];

break;

}

if (a[i] <= b[j])

{

m1 = m2;

m2 = a[i];

i++;

}

else

{

m1 = m2;

m2 = b[j];

j++;

}

}

return ( (m1 + m2)/2);

};

console.log(median([1,5,7,8,9],[1,3,4,6,9],5))

**ii) Using IIFE**

( function (a, b, n)

{

var i = 0;

var j = 0;

var count;

var m1 = -1, m2 = -1;

for (count = 0; count <= n; count++)

{

if (i == n)

{

m1 = m2;

m2 = b[0];

break;

}

else if (j == n)

{

m1 = m2;

m2 = a[0];

break;

}

if (a[i] <= b[j])

{

m1 = m2;

m2 = a[i];

i++;

}

else

{

m1 = m2;

m2 = b[j];

j++;

}

}

console.log( (m1 + m2)/2);

})

([1,5,7,8,9],[1,3,4,6,9],5));

7. Remove duplicate item from an array

**i)Using ANONYMOUS FUNCTION**

var unique = function (a){

let uniqueArr = [];

for(let i of a) {

if(uniqueArr.indexOf(i) === -1) {

uniqueArr.push(i);

}

}

return(uniqueArr);

}

console.log(unique([10,11,10,12,17,9,12]))

**ii)Using IIFE**

(function (a){

var uniqueArr = [];

for(let i of a) {

if(uniqueArr.indexOf(i) === -1) {

uniqueArr.push(i);

}

}

console.log(uniqueArr)

})([10,11,10,12,17,9,12]);

8) Rotate an array by k times

**i)Using Anonymous Function**

var rotate=function (a,k){

for(var i=0;i<k;i++){

a.unshift(a.pop());

}

return a;

}

console.log(rotate([5,7,9,2,4],3))

**ii)Using IIFE**

(function (a,k){

for(var i=0;i<k;i++){

a.unshift(a.pop());

}

console.log(a)

})([5,7,9,2,4],3);