Task3

Que.1 : Print odd numbers in an array?

Ans : with anonymous function

```
var getOddNumbers = function() {
      var arr = [1, 2, 3, 4, 5, 6];

      for (var i = 0; i < arr.length; i++) {
        if (arr[i] % 2 !== 0) {
          console.log(arr[i] );
        }
      }
  };

  getOddNumbers();
->1
  3
  5
```

Ans : with IIFE

```
(function() {
      var arr = [1, 2, 3, 4, 5, 6];

      for (var i = 0; i < arr.length; i++) {
        if (arr[i] % 2 !== 0) {
          console.log(arr[i] );
        }
      }
  })

  ();
->1
  3
  5
```

Que.2 : convert all the strings to title caps in a string array?

Ans : with anonymous function

```
var titleCase = function (string) {
    var sentence = string.toLowerCase().split(" ");
    for(var i = 0; i< sentence.length; i++){
      sentence[i] = sentence[i][0].toUpperCase() + sentence[i].slice(1);
    }
  console.log(sentence);
```

```
  return sentence;
  }
  titleCase("this is guvi task");
-> [ 'This', 'Is', 'Guvi', 'Task' ]
```

Ans: with IIFE

```
( function (string) {
    var sentence = string.toLowerCase().split(" ");
    for(var i = 0; i< sentence.length; i++){
      sentence[i] = sentence[i][0].toUpperCase() + sentence[i].slice(1);
    }
  console.log(sentence);
  return sentence;
  })
  ("this is guvi task");
->[ 'This', 'Is', 'Guvi', 'Task' ]
```

Que.3 : sum of all numbers in an array?

Ans : with anonymous function

```
var getSum = function() {
      var arr = [1, 2, 3, 4, 5, 6];
      var sum = 0;
      for (var i = 0; i < arr.length; i++) {
          sum = sum + arr[i];



      }
      console.log(sum);
   };

   getSum();
-> 21
```

Ans: with IIFE

```
(function() {
      var arr = [1, 2, 3, 4, 5, 6];
      var sum = 0;
      for (var i = 0; i < arr.length; i++) {
          sum = sum + arr[i];



      }
      console.log(sum);
```

```
  })
  ();
->21
```

Que.4 : Returns all the prime numbers in an array?

Ans : with anonymous function

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

Ans: with IIFE

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

Que.5 : Return all the palindromes in an array?

Ans :with anonymous function

```
var palindrome = function(array){
   var str = "";
   var rev = "";
   for(var i in array){
      rev = array[i].split("").reverse().join("");
      if(rev===array[i]){
      str = str + " " + array[i];
      }
   }
   console.log(str.trim(""));
};
palindrome(["hello","12121", "madam"]);
->12121 madam
```

Ans: with IIFE

```
(function(array){
   var str = "";
   var rev = "";
   for(var i in array){
      rev = array[i].split("").reverse().join("");
      if(rev===array[i]){
      str = str + " " + array[i];
      }
   }
   console.log(str.trim(""));
})
(["hello","12121", "madam"]);
->12121 madam
```

Que.6: Remove duplicates from an array?

Ans :with anonymous function

```
var duplicate = function (){
let chars = ['A', 'B', 'A', 'C', 'B','D'];
let uniqueChars = [...new Set(chars)];

console.log(uniqueChars);
}
duplicate();
->[ 'A', 'B', 'C', 'D' ]
```

Ans: with IIFE

```
(function(){
let chars = ['A', 'B', 'A', 'C', 'B','D'];
let uniqueChars = [...new Set(chars)];

console.log(uniqueChars);
})
();
->[ 'A', 'B', 'C', 'D' ]
```

Que.7 : Roatate an array by k times and return the rotated array?

Ans : with anonymous function

```
var num = [1,2,3,4,5,6,7];
const rotateArray1 = function(nums, k) {

  for (let i = 0; i < k; i++) {
     nums.unshift(nums.pop());
  }

  return nums;
};
console.log(rotateArray1(num,3));
-> [ 5, 6, 7, 1, 2, 3, 4 ]
```

Ans: with IIFE

```
(function(){
   var num = [1,2,3,4,5,6,7];
const rotateArray1 = (nums, k)=> {

  for (let i = 0; i < k; i++) {
     nums.unshift(nums.pop());
  }

  return nums;
};
console.log(rotateArray1(num,3));
})();
->[ 5, 6, 7, 1, 2, 3, 4 ]
```

Que.8 : Return median of two sorted arrays of the same size?

Ans : with anonymous function

```
const median = function(arr1, arr2){
  let concatArray = arr1.concat(arr2);
```

```
  concatArray.sort(function (array1,array2) {
    return array1 - array2;
  });
  let len = concatArray.length;

  return len%2 === 0 ? (concatArray[Math.floor(len/2)-1] + concatArray[Math.ceil(len/2)])/2 :
concatArray[Math.floor(len/2)];

};

let array1 = [1,12,15,26,38];
let array2 = [2,13,17,30,45];
console.log(median(array1,array2));
->16
```

Ans: with IIFE

```
(function(){
    const median = (arr1, arr2)=>{
  let concatArray = arr1.concat(arr2);
  concatArray.sort(function (array1,array2) {
    return array1 - array2;
  });
  let len = concatArray.length;

  return len%2 === 0 ? (concatArray[Math.floor(len/2)-1] + concatArray[Math.ceil(len/2)])/2 :
concatArray[Math.floor(len/2)];

};

let array1 = [1,12,15,26,38];
let array2 = [2,13,17,30,45];
console.log(median(array1,array2));
})
();
->16
```