Name: Silambarasan V

JavaScript - Day -3: Functions

**Do the below programs in anonymous function & IIFE**

1. **Print odd numbers in an array**

**Using an anonymous function:**

let numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9];

let odd = function () {

let odd= arr.filter(num => num %2 ==1)

console.log(odd)

}

odd()

**Using an IIFE:**

let numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9];

(function(arr) {

for (let num of arr) {

if (num % 2 !== 0) {

console.log(num);

}

}

})(numbers);

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

**Using an anonymous function:**

let stringArray = ["hello", "world", "javascript"];

let titleCapsArray = stringArray.map(function(str) {

return str.charAt(0).toUpperCase() + str.slice(1);

});

console.log(titleCapsArray);

**Using IIFE:**

let stringArray = ["hello", "world", "javascript"];

let titleCapsArray = (function(arr) {

return arr.map(function(str) {

return str.charAt(0).toUpperCase() + str.slice(1);

});

})(stringArray);

console.log(titleCapsArray);

1. **Sum of all numbers in an array:**

**Using an anonymous function:**

let numbers = [1, 2, 3, 4, 5];

let sum = function(arr) {

let total = 0;

arr.forEach(num => {

total += num;

});

return total;

};

console.log(sum(numbers));

**Using IIFE:**

let numbers = [1, 2, 3, 4, 5];

let sum = (function(arr) {

let total = 0;

arr.forEach(num => {

total += num;

});

return total;

})(numbers);

console.log(sum);

1. **Return all the prime numbers in an array:**

**Using an anonymous function:**

let array = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];

let primes = function(arr) {

return arr.filter(num => {

for (let i = 2; i <= Math.sqrt(num); i++) {

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

}

return num > 1;

});

};

console.log(primes(array));

**Using IIFE:**

let array = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];

let primes = (function(arr) {

return arr.filter(num => {

for (let i = 2; i <= Math.sqrt(num); i++) {

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

}

return num > 1;

});

})(array);

console.log(primes);

1. **Return all the palindromes in an array**

**Using an anonymous function:**

let isPalindrome = function () {

var myArray = ['viicc', 'racecar', 'honda', 'malayalam'];

var arr = myArray.filter(function (c, d) {

var palindrome = c.split('').reverse().join('');

if (c == palindrome) {

console.log(myArray[d]);

}

});

}

isPalindrome() // Output: [‘racecar’, ‘malayalam’]

**Using IIFE:**

( function () {

var myArray = ['viicc', 'racecar', 'honda', 'malayalam'];

var arr = myArray.filter(function (c, d) {

var palindrome = c.split('').reverse().join('');

if (c == palindrome) {

console.log(myArray[d]);

}

});

})() // Output: [‘racecar’, ‘malayalam’]

1. **Return median of two sorted arrays of the same size.**

**Using an anonymous function:**

let findMedian = function(arr1, arr2) {

let mergedArr = arr1.concat(arr2);

let sortedArr = mergedArr.sort((a, b) => a - b);

let n = sortedArr.length;

let mid = Math.floor(n / 2);

if (n % 2 === 0) {

return (sortedArr[mid - 1] + sortedArr[mid]) / 2;

} else {

return sortedArr[mid];

}

};

let arr1 = [1, 3, 5];

let arr2 = [2, 4, 6];

console.log(findMedian(arr1, arr2)); // Output: 3.5

**Using IIFE:**

let median = (function(arr1, arr2) {

let mergedArr = arr1.concat(arr2);

let sortedArr = mergedArr.sort((a, b) => a - b);

let n = sortedArr.length;

let mid = Math.floor(n / 2);

if (n % 2 === 0) {

return (sortedArr[mid - 1] + sortedArr[mid]) / 2;

} else {

return sortedArr[mid];

}

})([1, 3, 5], [2, 4, 6]);

console.log(median); // Output: 3.5

1. **Remove duplicates from an array:**

**Using an anonymous function:**

const removeDuplicates = function(array) {

return Array.from(new Set(array));

};

const myArray = [1, 2, 2, 3, 4, 4, 5];

const uniqueArray = removeDuplicates(myArray);

console.log(uniqueArray); // Output: [1, 2, 3, 4, 5]

**Using IIFE:**

const myArray = [1, 2, 2, 3, 4, 4, 5];

const uniqueArray = (function(array) {

return Array.from(new Set(array));

})(myArray);

console.log(uniqueArray); // Output: [1, 2, 3, 4, 5]

1. **Rotate an array by k times:**

**Using an anonymous function:**

let arr = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];

let k = 3;

let rot = function(arr) {

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

arr.unshift(arr.pop());

}

console.log(arr);

}

rot(arr);

arr = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]; // output: [ 8, 9, 10, 1, 2, 3, 4, 5, 6, 7 ]

**Using IIFE:**

let arr = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];

let k = 3;

(function(arr) {

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

arr.unshift(arr.pop());

}console.log(arr);})(arr); // output: [ 8, 9, 10, 1, 2, 3, 4, 5, 6, 7 ]

**Do the below programs in arrow functions.**

1. **Print odd numbers in an array:**

let numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];

let oddNumbers = numbers.filter(num => num % 2 !== 0);

console.log(oddNumbers); // Output: [ 1, 3, 5, 7, 9 ]

1. **Convert all the strings to title caps in a string array:**

let stringArray = ["hello", "world", "javascript"];

let titleCapsArray = stringArray.map(str => str.charAt(0).toUpperCase() + str.slice(1).toLowerCase());

console.log(titleCapsArray); // output: [ 'Hello', 'World', 'Javascript' ]

1. **Sum of all numbers in an array:**

let array = [1, 2, 3, 4, 5];

let sum = array.reduce((total, num) => total + num, 0);

console.log(sum); // output: 15

1. **Return all the prime numbers in an array:**

let numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];

let isPrime = num => {

for(let i = 2, sqrt = Math.sqrt(num); i <= sqrt; i++) {

if(num % i === 0) {

return false;

}

}

return num > 1;

};

let getPrimeNumbers = arr => arr.filter(isPrime);

console.log(getPrimeNumbers(numbers)); // output: [ 2, 3, 5, 7 ]

1. **Return all the palindromes in an array:**

const arr = ['carecar', 1344, 12321, 'did', 'cannot'];

const isPalindrome = el => {

   const str = String(el);

   let i = 0;

   let j = str.length - 1;

   while(i < j) {

      if(str[i] === str[j]) {

         i++;

         j--;

      }

      else {

         return false;

      }

   }

   return true;

};

const findPalindrome = arr => {

   return arr.filter(el => isPalindrome(el));

};

console.log(findPalindrome(arr)); // output: [12321, 'did' ]