ZEN class JS B27 WE day5 task:

1. Do the below programs in anonymous function & IIFE

* 1. Print odd numbers in an array
  2. Convert all the strings to title caps in a string array
  3. Sum of all numbers in an array
  4. Return all the prime numbers in an array
  5. Return all the palindromes in an array
  6. Return median of two sorted arrays of same size
  7. Remove duplicates from an array
  8. Rotate an array by k times

Sol:

IIFE:

//IIFE;

//a. print odd numbers in an array

(function (inputArray) {

let oddArray = [];

console.log(`input Array is ${inputArray}.`);

for (let i = 0; i < inputArray.length; i++) {

if (inputArray[i] % 2 != 0) {

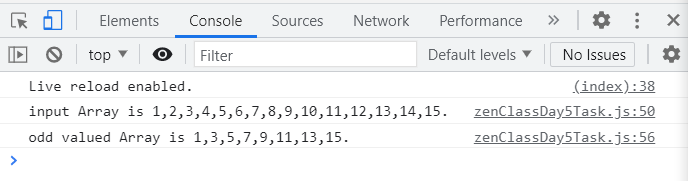
oddArray.push(inputArray[i]); //creating odd valued array

}

}

console.log(`odd valued Array is ${oddArray}.`);

})([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]);

Output:  


//b. convert all strings to title caps ina astring array

(function (inputArray) {

console.log(`input Array is ${inputArray}.`);

let capsArray = [];

for (let i = 0; i < inputArray.length; i++) {

let tempValue =

inputArray[i].slice(0, 1).toUpperCase() +

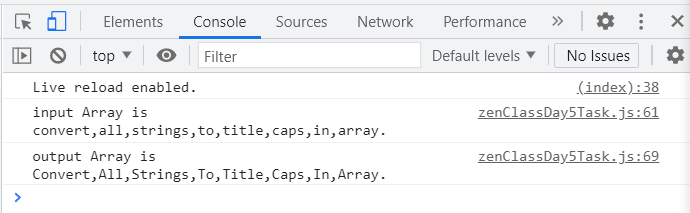
inputArray[i].slice(1, inputArray[i].length).toLowerCase();

capsArray.push(tempValue); //creating odd valued array

}

console.log(`output Array is ${capsArray}.`);

})(["convert", "all", "strings", "to", "title", "caps", "in", "array"]);

Output:  


//c.sum of all numbers in an array

(function (inputArray) {

console.log(`input array is ${inputArray}`);

let sumValue = 0;

for (let i = 0; i < inputArray.length; i++) {

if (typeof inputArray[i] == "number") {

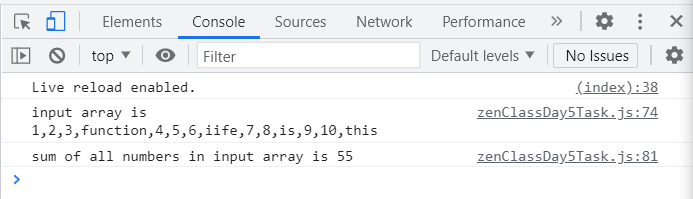
sumValue += inputArray[i]; //to add all numbers in an array

} //ignoring strings

}

console.log(`sum of all numbers in input array is ${sumValue}`);

})([1, 2, 3, "function", 4, 5, 6, "iife", 7, 8, "is", 9, 10, "this"]);

Output:  


//d. return all prime numbers in an array

(function (inputArray) {

let primeArray = [];

console.log(`input array is ${inputArray}`);

for (let i = 0; i < inputArray.length; i++) {

let flag; //declaring flag variable to recognise as prime

if (inputArray[i] == 2) {

flag = 1; //declaring 2 as prime number as below for loop starts with 2

}

if (inputArray[i] > 1) {

//prime number must be greater than 1

for (let j = 2; j < inputArray[i]; j++) {

if (inputArray[i] % j == 0) {

//prime number must not be divisible by other than 1 and itself

flag = 0;

break;

} else {

flag = 1;

}

}

}

if (flag == 1) {

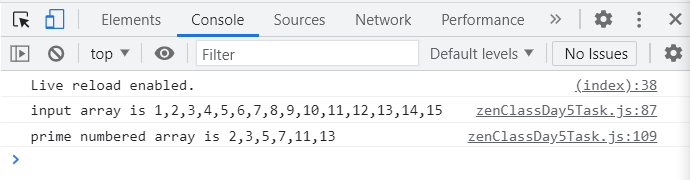
primeArray.push(inputArray[i]); //using flag to recognise prime number and create prime number array

}

}

console.log(`prime numbered array is ${primeArray}`);

})([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]);

Output:  


//e. return all palindromes in an array

(function (inputArray) {

console.log(`input array is ${inputArray}`);

let palindromeArray = [];

for (let i = 0; i < inputArray.length; i++) {

if (typeof inputArray[i] === "string") {

let tempString = inputArray[i];

if (tempString.split("").reverse().join("") == inputArray[i]) {

palindromeArray.push(inputArray[i]);

} //checks if string is palindrome and pushes into new array

} //takes strings as inputs

else if (typeof inputArray[i] === "number") {

let tempNumber = String(inputArray[i]);

if (tempNumber.split("").reverse().join("") == inputArray[i]) {

palindromeArray.push(inputArray[i]);

} //checks if number is palindrome and pushes into new array

} //takes numbers as inputs

}

console.log(`palindrome array is ${palindromeArray}`);

})([

121,

2443,

"checking",

5667,

5679,

"fofof",

1001,

"a",

121314,

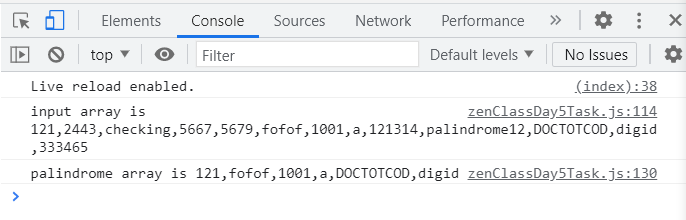
"palindrome12",

"DOCTOTCOD",

"digid",

333465,

]);

Output:  


//f. return median of two sorted arrays of same size after merging them

(function (inpArr1, inpArr2) {

console.log(`inputs are arr1: ${inpArr1}, arr2: ${inpArr2}`);

//merging the arrays together

let finalInpArray = inpArr1.concat(inpArr2);

finalInpArray = finalInpArray.sort(function (a, b) {

return a - b;

});

if (finalInpArray.length % 2 == 0) {

console.log(

(finalInpArray[finalInpArray.length / 2 - 1] +

finalInpArray[finalInpArray.length / 2]) /

2

); //average of two middle elements as the length is even

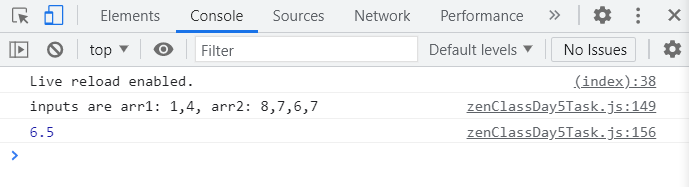
} else {

console.log(finalInpArray[Math.floor(finalInpArray.length / 2)]);

//middle element is the median as the length is odd.

}

})([1, 4], [8, 7, 6, 7]);

Output:  


//g. remove duplicates in an array

(function (inputArray) {

console.log(`input array is ${inputArray}`);

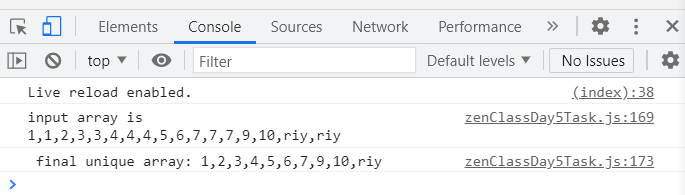
//converting array into set to remove duplicates

let uniqueSet = new Set(inputArray);

uniqueArray = [...uniqueSet];

console.log(` final unique array: ${uniqueArray}`);

})([1, 1, 2, 3, 3, 4, 4, 4, 5, 6, 7, 7, 7, 9, 10, "riy", "riy"]);

Output:  


// h. rotate array by k times

(function (inputArray, noOfTimes) {

console.log(`input array: ${inputArray} must be rotated ${noOfTimes} times`);

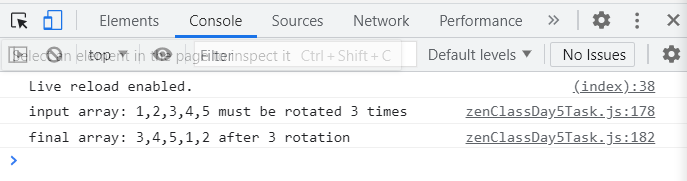
for (let i = 0; i < noOfTimes; i++) {

inputArray.unshift(inputArray.pop()); //unshift adds element to start of the array, pop removes element from end of the array, rotation takes place for input given no of times

}

console.log(`final array: ${inputArray} after ${noOfTimes} rotation`);

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

Output:  


ANONYMOUS FUNCTIONS:

//ANONYMOUS FUNCTIONS;

// a. print odd numbers in an array

let anonyOdd = (function (inputArray) {

let oddArray = [];

console.log(`input Array is ${inputArray}.`);

for (let i = 0; i < inputArray.length; i++) {

if (inputArray[i] % 2 != 0) {

oddArray.push(inputArray[i]); //creating odd valued array

}

}

console.log(`odd valued Array is ${oddArray}.`);

})([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]);

//b. convert all strings to title caps ina astring array

let anonyTitleCaps = (function (inputArray) {

console.log(`input Array is ${inputArray}.`);

let capsArray = [];

for (let i = 0; i < inputArray.length; i++) {

let tempValue =

inputArray[i].slice(0, 1).toUpperCase() +

inputArray[i].slice(1, inputArray[i].length).toLowerCase();

capsArray.push(tempValue); //creating odd valued array

}

console.log(`output Array is ${capsArray}.`);

})(["convert", "all", "strings", "to", "title", "caps", "in", "array"]);

//c.sum of all numbers in an array

let anonyArraySum = (function (inputArray) {

console.log(`input array is ${inputArray}`);

let sumValue = 0;

for (let i = 0; i < inputArray.length; i++) {

if (typeof inputArray[i] == "number") {

sumValue += inputArray[i]; //to add all numbers in an array

} //ignoring strings

}

console.log(`sum of all numbers in input array is ${sumValue}`);

})([1, 2, 3, "function", 4, 5, 6, "iife", 7, 8, "is", 9, 10, "this"]);

//d. return all prime numbers in an array

let anonyPrime = (function (inputArray) {

let primeArray = [];

console.log(`input array is ${inputArray}`);

for (let i = 0; i < inputArray.length; i++) {

let flag; //declaring flag variable to recognise as prime

if (inputArray[i] == 2) {

flag = 1; //declaring 2 as prime number as below for loop starts with 2

}

if (inputArray[i] > 1) {

//prime number must be greater than 1

for (let j = 2; j < inputArray[i]; j++) {

if (inputArray[i] % j == 0) {

//prime number must not be divisible by other than 1 and itself

flag = 0;

break;

} else {

flag = 1;

}

}

}

if (flag == 1) {

primeArray.push(inputArray[i]); //using flag to recognise prime number and create prime number array

}

}

console.log(`prime numbered array is ${primeArray}`);

})([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]);

//e. return all palindromes in an array

let anonyPalindrome = (function (inputArray) {

console.log(`input array is ${inputArray}`);

let palindromeArray = [];

for (let i = 0; i < inputArray.length; i++) {

if (typeof inputArray[i] === "string") {

let tempString = inputArray[i];

if (tempString.split("").reverse().join("") == inputArray[i]) {

palindromeArray.push(inputArray[i]);

} //checks if string is palindrome and pushes into new array

} //takes strings as inputs

else if (typeof inputArray[i] === "number") {

let tempNumber = String(inputArray[i]);

if (tempNumber.split("").reverse().join("") == inputArray[i]) {

palindromeArray.push(inputArray[i]);

} //checks if number is palindrome and pushes into new array

} //takes numbers as inputs

}

console.log(`palindrome array is ${palindromeArray}`);

})([

121,

2443,

"checking",

5667,

5679,

"fofof",

1001,

"a",

121314,

"palindrome12",

"DOCTOTCOD",

"digid",

333465,

]);

//f. return median of two sorted arrays of same size after merging them

let anonyMedian = (function (inpArr1, inpArr2) {

console.log(`inputs are arr1: ${inpArr1}, arr2: ${inpArr2}`);

//merging the arrays together

let finalInpArray = inpArr1.concat(inpArr2);

finalInpArray = finalInpArray.sort(function (a, b) {

return a - b;

});

if (finalInpArray.length % 2 == 0) {

console.log(

(finalInpArray[finalInpArray.length / 2 - 1] +

finalInpArray[finalInpArray.length / 2]) /

2

); //average of two middle elements as the length is even

} else {

console.log(finalInpArray[Math.floor(finalInpArray.length / 2)]);

//middle element is the median as the length is odd.

}

})([1, 4], [8, 7, 6, 7]);

//g. remove duplicates in an array

let anonyDuplicate = (function (inputArray) {

console.log(`input array is ${inputArray}`);

//converting array into set to remove duplicates

let uniqueSet = new Set(inputArray);

uniqueArray = [...uniqueSet];

console.log(` final unique array: ${uniqueArray}`);

})([1, 1, 2, 3, 3, 4, 4, 4, 5, 6, 7, 7, 7, 9, 10, "riy", "riy"]);

// h. rotate array by k times

let anonyRotate = (function (inputArray, noOfTimes) {

console.log(`input array: ${inputArray} must be rotated ${noOfTimes} times`);

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

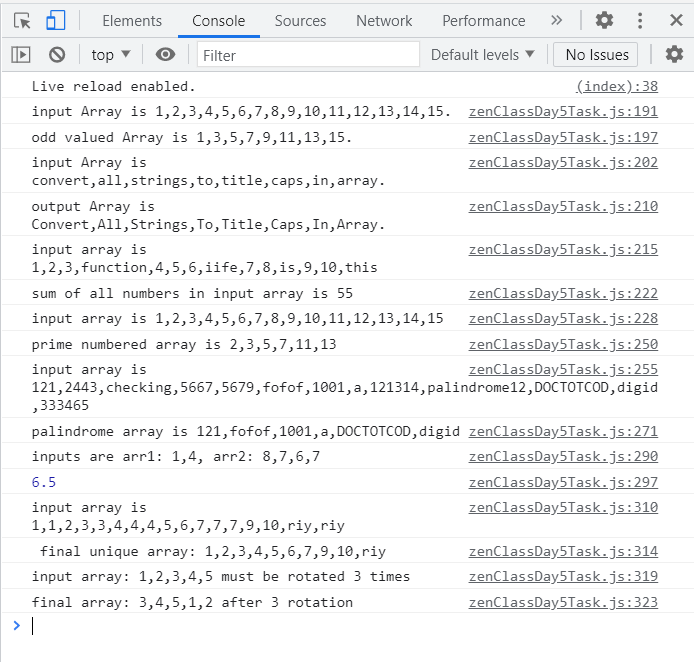
inputArray.unshift(inputArray.pop()); //unshift adds element to start of the array, pop removes element from end of the array, rotation takes place for input given no of times

}

console.log(`final array: ${inputArray} after ${noOfTimes} rotation`);

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

Output for anonymous functions:



ARROW FUNCTIONS:

//ARROW FUNCTIONS;

// a. print odd numbers in an array

let arrowOdd = (inputArray) => {

let oddArray = [];

console.log(`input Array is ${inputArray}.`);

for (let i = 0; i < inputArray.length; i++) {

if (inputArray[i] % 2 != 0) {

oddArray.push(inputArray[i]); //creating odd valued array

}

}

console.log(`odd valued Array is ${oddArray}.`);

};

arrowOdd([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]);

//b. convert all strings to title caps ina astring array

let arrowTitleCaps = (inputArray) => {

console.log(`input Array is ${inputArray}.`);

let capsArray = [];

for (let i = 0; i < inputArray.length; i++) {

let tempValue =

inputArray[i].slice(0, 1).toUpperCase() +

inputArray[i].slice(1, inputArray[i].length).toLowerCase();

capsArray.push(tempValue); //creating odd valued array

}

console.log(`output Array is ${capsArray}.`);

};

arrowTitleCaps([

"convert",

"all",

"strings",

"to",

"title",

"caps",

"in",

"array",

]);

//c.sum of all numbers in an array

let arrowArraySum = (inputArray) => {

console.log(`input array is ${inputArray}`);

let sumValue = 0;

for (let i = 0; i < inputArray.length; i++) {

if (typeof inputArray[i] == "number") {

sumValue += inputArray[i]; //to add all numbers in an array

} //ignoring strings

}

console.log(`sum of all numbers in input array is ${sumValue}`);

};

arrowArraySum([

1,

2,

3,

"function",

4,

5,

6,

"iife",

7,

8,

"is",

9,

10,

"this",

]);

//d. return all prime numbers in an array

let arrowPrime = (inputArray) => {

let primeArray = [];

console.log(`input array is ${inputArray}`);

for (let i = 0; i < inputArray.length; i++) {

let flag; //declaring flag variable to recognise as prime

if (inputArray[i] == 2) {

flag = 1; //declaring 2 as prime number as below for loop starts with 2

}

if (inputArray[i] > 1) {

//prime number must be greater than 1

for (let j = 2; j < inputArray[i]; j++) {

if (inputArray[i] % j == 0) {

//prime number must not be divisible by other than 1 and itself

flag = 0;

break;

} else {

flag = 1;

}

}

}

if (flag == 1) {

primeArray.push(inputArray[i]); //using flag to recognise prime number and create prime number array

}

}

console.log(`prime numbered array is ${primeArray}`);

};

arrowPrime([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]);

//e. return all palindromes in an array

let arrowPalindrome = (inputArray) => {

console.log(`input array is ${inputArray}`);

let palindromeArray = [];

for (let i = 0; i < inputArray.length; i++) {

if (typeof inputArray[i] === "string") {

let tempString = inputArray[i];

if (tempString.split("").reverse().join("") == inputArray[i]) {

palindromeArray.push(inputArray[i]);

} //checks if string is palindrome and pushes into new array

} //takes strings as inputs

else if (typeof inputArray[i] === "number") {

let tempNumber = String(inputArray[i]);

if (tempNumber.split("").reverse().join("") == inputArray[i]) {

palindromeArray.push(inputArray[i]);

} //checks if number is palindrome and pushes into new array

} //takes numbers as inputs

}

console.log(`palindrome array is ${palindromeArray}`);

};

arrowPalindrome([

121,

2443,

"checking",

5667,

5679,

"fofof",

1001,

"a",

121314,

"palindrome12",

"DOCTOTCOD",

"digid",

333465,

]);

//f. return median of two sorted arrays of same size after merging them

let arrowMedian = (inpArr1, inpArr2) => {

console.log(`inputs are arr1: ${inpArr1}, arr2: ${inpArr2}`);

//merging the arrays together

let finalInpArray = inpArr1.concat(inpArr2);

finalInpArray = finalInpArray.sort(function (a, b) {

return a - b;

});

if (finalInpArray.length % 2 == 0) {

console.log(

(finalInpArray[finalInpArray.length / 2 - 1] +

finalInpArray[finalInpArray.length / 2]) /

2

); //average of two middle elements as the length is even

} else {

console.log(finalInpArray[Math.floor(finalInpArray.length / 2)]);

//middle element is the median as the length is odd.

}

};

arrowMedian([1, 4], [8, 7, 6, 7]);

//g. remove duplicates in an array

let arrowDuplicate = (inputArray) => {

console.log(`input array is ${inputArray}`);

//converting array into set to remove duplicates

let uniqueSet = new Set(inputArray);

uniqueArray = [...uniqueSet];

console.log(` final unique array: ${uniqueArray}`);

};

arrowDuplicate([1, 1, 2, 3, 3, 4, 4, 4, 5, 6, 7, 7, 7, 9, 10, "riy", "riy"]);

// h. rotate array by k times

let arrowRotate = (inputArray, noOfTimes) => {

console.log(`input array: ${inputArray} must be rotated ${noOfTimes} times`);

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

inputArray.unshift(inputArray.pop()); //unshift adds element to start of the array, pop removes element from end of the array, rotation takes place for input given no of times

}

console.log(`final array: ${inputArray} after ${noOfTimes} rotation`);

};

arrowRotate([1, 2, 3, 4, 5], 3);

Output:  
