Js109 interview study session 9-29-2021 9pm

1 hour for interview

2 problems

Pedac

Verbalize

Coderpad – strict mode – tricky with brackets

Medium level small problems

About 20 mins or less

In an interview format

Speak aloud

Pedac format

Try to record yourself

Screen record

Treat like interview for a job

Use correct terminology

// Problem:

// Inputs:

// Outputs:

// Examples:

// Edge Cases:

// Data Structure:

// Algo:

P2.js

/\* eslint-disable max-len \*/

/\* eslint-disable no-trailing-spaces \*/

// Problem:

// Write a function that will return the count of distinct case-insensitive

// alphabetic characters and numeric digits that occur more than once in the

// input string.

// The input string can be assumed to contain only alphabets

// (both uppercase and lowercase) and numeric digits.

// Inputs: string of only alphanumeric

// Outputs: number - integer

// count of distinct case-insensitive

// alphabetic characters and numeric digits that occur more than once in the

// input string.

// Examples:

console.log(duplicateCount("")) == 0;

console.log(duplicateCount("abcde")) == 0;

console.log(duplicateCount("abcdeaa")) == 1;

console.log(duplicateCount("abcdeaB")) == 2;

console.log(duplicateCount("Indivisibilities")) == 2;

// Edge Cases:

// Data Structure:

// Use an object to hold count of each occurence of a alphanumeric

// Algo:

// split the input string into an array of single characters

// create an empty "seen" object

// iterate over the array

// if current element already in seen object then increment count

// otherwise add a new property to the object for the character with a count of 1

// set counter to 0

// iterate over the seen object

// increment counter if a property is > 1

// return counter

function duplicateCount(str) {

let arr = str.split('');

let seen = {};

arr.forEach(elm => {

if (elm.toLowerCase() in seen) {

seen[elm.toLowerCase()] += 1;

} else {

seen[elm.toLowerCase()] = 1;

}

});

let counter = 0;

for (let prop in seen) {

if (seen[prop] > 1) counter += 1;

}

return counter;

}

P1.js

function threeByThree(inArr) {

let sumArr = [];

inArr.forEach(str => {

let digitsArr = str.split('');

sumArr.push(digitsArr.reduce((a,b) => Number(a) + Number(b), 0));

});

return sumArr.filter(sum => sum % 3 === 0).length >= 3;

}

console.log(threeByThree(['01112', '0111', '00030', '2043', '12043']));

// true

console.log(threeByThree(['01112', '2043', '12043']));

// false

console.log(threeByThree(['01112', '2043']));

// false

console.log(threeByThree(['93', '9', '1', '25', '1212']));

// true