College of Engineering, Construction & Living Sciences
Bachelor of Information Technology
ID607001: Introductory Application Development Concepts
Level 6, Credits 15

In-Class Activity: ES6 Basics 1

Instructions

The purpose of this in-class activity is to familiarise yourself with the **ES6** syntax as well as develop your problem-solving skills. The following 15 problems are commonly asked in coding interviews. You may come across one or two of these when you apply for software development/engineering positions in the future. **Note:** do not use functional programming constructs such as **map**, **filter** & **reduce** to solve some of these problems.

Code Review

You must submit all program files via **GitHub Classroom**. Here is the URL to the repository you will use for your code review – https://classroom.github.com/a/_6KSahyX. Checkout from the **master** or **main** branch to the **01-in-class-activity** branch by running the command - **git checkout 01-in-class-activity**. This branch will be your development branch for this activity. Once you have completed this activity, create a pull request & assign the **GitHub** user **grayson-orr** to a reviewer. **Do not** merge your pull request.

Getting Started

Open your repository in **Visual Studio Code**. Create a new file called **01-in-class-activity.js**. In **01-in-class-activity.js**, add the following:

```
console.log('Hello, World!')
```

Open a **terminal** & run the following command:

```
node 01-in-class-activity.js
```

If the output is **Hello, World!**, then you are ready to start coding.

Problem 1:

Declare two **immutable** variables called **name** & **age** with the values Jane & 45. Use the two variables & **string interpolation** to display the expected output.

```
// Write your solution here

// Expected output:
// Hello my name is Jane & I am 45 years old.
```

Problem 2:

Calculate the **sum** of the given **integers** & use **string interpolation** to display the expected output.

```
const x = 1957452
const y = 2975635

// Write your solution here

// Expected output:
// The sum of 1957452 & 2975635 is 4933087
```

Problem 3:

Calculate the **average** of the given **array** of **doubles** called **nums** & use **string interpolation** to display the expected output.

```
const nums = [45.3, 67.5, -45.6, 20.34, -33.0, 45.6]

// Write your solution here

// Expected output:
// Average: 16.69
```

Problem 4:

Write an **arrow function** called **fizzBuzz** which accepts an **integer num**. If **num** is a multiple of three, return **Fizz**, if **num** is a multiple of five, return **Buzz** & if **num** is a multiple of three & five, return **FizzBuzz**. Call the **fizzBuzz** function in the for loop to display the expected output.

```
// Write your fizzBuzz function here

for (let i = 1; i <= 15; i += 2) {
    // Write your solution here
}

// Expected output:
// 1
// Fizz
// Buzz
// 7
// Fizz
// 11
// 13
// FizzBuzz
```

Problem 5:

You have been given an **array** of **integers** called **nums**. Display **only** the odd numbers in **nums**. Sort from lowest to highest.

```
const nums = [21, 19, 68, 55, 42, 12]

// Write your solution here

// Expected output:
// 19
// 21
// 55
```

Problem 6:

Write an **arrow function** called **isAnagram** which accepts two parameters called **someStrOne** & **someStrTwo**. In the function block, write some code that checks whether or not **someStrOne** & **someStrTwo** are an anagram. **Note:** An anagram is a word or phrase that made by arranging the letters of another word or phrase in a different order. If you are still unsure what an anagram is, here is an example:

```
Input: isAnagram('elvis', 'lives')
Output: true

Input: isAnagram('cat', 'sat')
Output : false
```

Call the **isAnagram** function to display the expected output.

```
// Write your solution here

// Expected output:
// true
// false
```

Problem 7:

Write an **arrow function** called **convert** which accepts two parameters called **hours** & **minutes**. In the function block, write some code that converts both **hours** & **minutes** to seconds, then adds them together.

```
// Write your solution here

console.log(convert(1, 3))

// Expected output:
// 3780
```

Problem 8:

Write an **arrow function** called **palindrome** which accepts a single parameter called **someStr**. In the function block, determine whether or not **someStr** is a palindrome. The function should return a **boolean**.

```
// Write your solution here

console.log(palindrome('A man, a plan, a canal - Panama'))
console.log(palindrome('Hello, World!'))

// Expected output:
// true
// false
```

Problem 9:

Write an **arrow function** called **isLessThanFiveLetters** which accepts an **array** of **strings**. In the function block, return all words that are less than **five** letters. Sort from A to Z.

```
// Write your solution here

const transport = ['car', 'bike', 'scooter', 'skateboard', 'truck', 'walk']

// Expected output:
// bike
// car
// walk
```

Problem 10:

Write an **arrow function** called **findBreed** which accepts an unsorted **array** of **strings** called **breeds**. Your code needs to search **breeds** for 'Afghan Hound' & return its location in the **array**, i.e., index. If 'Afghan Hound' is not in **breeds**, return -1.

```
// Write your solution here

const breeds = ['Afghan Hound', /** Add your other breeds here */]
console.log(findBreed(breeds))

// Expected output:
// 1
```

Problem 11:

Write an **arrow function** called **removeVowels** which accepts a **string** called **word** & returns a new **string** with all vowels removed. Also, how would you handle the edge case where **word** does not contain vowels.

```
// Write your solution here

const word = // Add your word here
console.log(removeVowels(word))
```

Problem 12:

Write an **arrow function** function called **missingNum** which accepts an unsorted **array** of **integers** called **nums** & return the missing number.

```
// Write your solution here

const nums = [10, 3, 4, 8, 1, 7, 6, 9, 5]
console.log(missingNum(nums))

// Expected output:
// 2
```

Problem 13:

Write an **arrow function** called **fileExtensions** which accepts an **array** of **objects** called **files** & returns their extension names.

```
// Write your solution here

const files = [
    { 'name': 'index', 'extension': 'html' },
    { 'name': 'main', 'extension': 'js' },
    { 'name': 'sample', 'extension': 'txt' },
    { 'name': 'data', 'extension': 'json' }
]
console.log(fileExtensions(files))

// Expected output:
// html
// js
// txt
// json
```

Problem 14:

What is a substring? It is a portion of a **string**, i.e., 'Hello' is a substring of 'Hello, World!' and 'el' is a substring of 'Hello'. String manipulation is commonly used & working with substrings is something you will often do.

You have been given the following sentence as a **string**:

'The anemone, the wild violet, the hepatica, and the funny little curled-up.'

Write code that returns the number of occurrences of the word **'the'** in the sentence above.

```
const sentence = 'The anemone, the wild violet, the hepatica, and the funny little curled-up.'

// Write your solution here

// Expected output:
// 4
```

Problem 15:

In this problem you are going to use the **abs()** function. Write an **arrow function** called **calcDist** which calculates the distance between two **integers**. It does not matter which order the parameters are given; it should still return the same result.

```
// Write your solution here

console.log(calcDist(-1, 4))
console.log(calcDist(4, -1))

// Expected output:
// 5
// 5
```