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

In-Class Activity: ES6 Basics Extension

Instructions

The purpose of this in-class activity is to extend yourself with more complex problems. Also, you will articulate various **JavaScript** concepts.

Submission

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. If you wish to have your code reviewed, message the course lecturer on **Microsoft Teams**.

Getting Started

Open your repository in **Visual Studio Code**. Create a new file called **03-in-class-activity.js**. In **03-in-class-activity.js**, add the following:

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

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

```
node 03-in-class-activity.js
```

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

Problem 1:

You have been given two **arrays** containing the lecturer's favourite programming languages. Use the following hints to display the expected output:

- Add a specified element to the end of a list.
- Add all elements of a specified **array** to the end of a list.

- If present, remove a specified element from a **array**.
- Capitalise the element in the 3rd index.

```
const progLangsOne = ['C#', 'JavaScript', 'Kotlin', 'OCaml']
const progLangsTwo = ['C++', 'Go', 'Swift', 'TypeScript']

// Write your solution here

// Expected output:
// [C#, JavaScript, Kotlin, OCAML, Prolog, C++, Swift]
```

Problem 2:

Create an **arrow function** which simulates the **Rock, Paper, Scissors** game. The **arrow function** takes the input of two players (rock, paper or scissors), first parameter from the first player, second from the second player. The function returns the result as such:

- First player wins
- Second player wins
- Draw

```
// Write your solution here

console.log(rockPaperScissor('paper', 'rock'))
console.log(rockPaperScissor('rock', 'paper'))
console.log(rockPaperScissor('paper', 'paper'))

// Expected output:
// First player wins
// Second player wins
// Draw
```

Problem 3:

You are given the length of a song in minutes. The format is **mm:ss**, i.e., minutes:seconds or ”01:00”. Create an **arrow function** that takes the song’s length & returns it in seconds. **Note:** If the number of seconds is 60, return **false**.

```
// Write your solution here

console.log(minToSecs('01:00'))
console.log(minToSecs('13:56'))
console.log(minToSecs('10:60'))

// Expected output:
// 60
// 836
// false
```

Problem 4:

Explain the difference between **const** & **Object.freeze()** using the following example:

```
// const example
const fruit = { name: 'Apple' }
let vege = { name: 'Carrot' }
fruit = vege

// Object.freeze() example
let person = { name: 'John' }
Object.freeze(person)
person.name = 'Jane'
console.log(person)
```

Problem 5:

Explain the difference between **implicit** & **explicit** type coercion.

Problem 6:

Explain what an **Immediately Invoked Function Expression (IIFE)** is. Provide an example in your answer.

Problem 7:

Explain what a **generator** is. Provide an example in your answer.

Problem 8:

Explain difference between an **undeclared**, **undefined** & **null** variable.

Problem 9:

Provide **three** examples on how you can empty an **array**.

Problem 10:

Explain the expected output for the following example:

```
const progLangsOne = ['C#', 'JavaScript', 'Kotlin', 'OCaml']
delete progLangsOne[2]
console.log(progLangsOne)
```

Problem 11:

Explain the operators used & expected output in the following example:

```
+true
!'John'
```

Problem 12:

Explain the difference between **prefix** & **postfix** increment/decrement operators.

Problem 13:

Explain what **currying** is using the following example:

```
const makeSandwich = (ingredientOne) => {
    return (ingredientTwo) => {
        return (ingredientThree) => {
            return `${ingredientOne}, ${ingredientTwo}, ${ingredientThree}`
        }
    }
}

const sandwich = makeSandwich('Bacon')('Lettuce')('Tomato')
console.log(sandwich) // Bacon, Lettuce, Tomato
```

Problem 14:

Explain the following **expressions** & **operators**:

- Exponentiation assignment (**=)
- Logical nullish assignment (??=)
- Nullish coalescing operator (??)
- Optional chaining (?.)
- Remainder assignment (%=)

Problem 15:

Explain what a **Set** is. Provide an example in your answer.