LESSON 2 – JAVASCRIPT ES6

CSSE3101 – ADVANCED WEB TECHNOLOGIES

What JavaScript ES6?

- ECMAScript 2015 or ES2015 is a significant update to the JavaScript programming language. It is the first major update to the language since ES5 which was standardized in 2009. Therefore, ES2015 is often called ES6.
- React uses ES6, and you should be familiar with some of the new features
- ECMAScript (European Computer Manufacturers Association Script)

Arrow Functions

Arrow functions allow us to write shorter function syntax:

Before

```
function hello()
{
return "Hello World!";
}
```

With arrow functions

```
const hello = () => {
return "Hello World!";
}
```

Arrow Functions

Example:

Before

```
function computeAge(year){
      const d = new Date();
      let y = d.getFullYear();
      let age = y-year;
      return age;
}
```

With arrow functions

```
const computeAge = (year)=>{
      const d = new Date();
      let y = d.getFullYear();
      let age = y-year;
      return age;
}
```

Variable Scopes

- Scope stands for where our variable will be available to use inside our code and where it will not.
- Block means a pair of curly brackets; a block can be anything that contains an opening and closing curly bracket.
- Variable having Block-scope will only be available to use inside the block it declared, will not be accessible outside the block, and will give Reference Error if we try to access.

Block-scope

```
if(true) {
    let myName = "your name";
    console.log(myName); //output=> "your name"
}
console.log(myName); //output => ReferenceError
```

ES6 Variables

Defining your variables: let, and const.

let

```
let x = 20;
```

`let has a block scope.`

const

```
const x = 5.6;
```

`const has a block scope.`

const is a variable that once it has been created, its value can never change.

ES6 Array Methods(map)

The .map() method allows you to run a function on each item in the array, returning a new array as the result.

In React, map() can be used to generate lists.

Example:

```
const myArray = ['apple', 'banana', 'orange'];
myArray.map((item) => console.log(item));
```

ES6 Array Methods(filter)

- The JavaScript **Array filter()** Method is used to create a new array from a given array consisting of only those elements from the given array which satisfy a condition set by the argument method.

ES6 Array Methods(filter)

```
// JavaScript to illustrate findIndex() method
    function canVote(age) {
        return age >= 18;
    }

    function func() {
        var filtered = [24, 33, 16, 40].filter(canVote);
        console.log(filtered);
    }
    func();
```

React ES6 Destructuring

Destructuring is a convenient way of creating new variables by extracting some values from data stored in objects or arrays.

Accessing elements in array

```
const stud =['Ali','A+','Web'];

const name = stud[0];
const mark = stud[1];
const course = stud[2];
console.log(name);
```

Destructuring

```
const stud =['Ali','A+','Web'];
const [name,mark,course] = stud;
console.log(mark);
console.log(name);
```

Destructuring Objects

```
const student = {
  id: '46J12345',
  name: 'Ahmed',
  mark: 'A+',
  course: 'CSSE3101',
}
studentInfo(student);
function studentInfo({id, name, mark, course}) {
  const message = 'Student  ' + id + ' with the name ' + name + '
has the mark ' + mark + ' in the ' + course + '.';

  document.getElementById("demo").innerHTML = message;
}
```

ES6 Spread Operator

The JavaScript spread operator (...) allows us to quickly copy all or part of an existing array or object into another array or object.

Example:

```
const marks1 = [1, 2, 3];
const marks2 = [4, 5, 6];
const marks = [...marks1, ...marks2];
console.log(marks);
```

ES6 Modules

JavaScript modules allow you to break up your code into separate files.

ES Modules rely on the import and export statements.

NAMED EXPORT:

```
const name = "Ali";
const age = 21
export { name, age }
```

Saved in another file.

Import from named exports:

```
import {name,age} from "./person.js";
```

Access the variables in another file.

ES6 Modules

JavaScript modules allow you to break up your code into separate files.

ES Modules rely on the import and export statements.

DEFAULT EXPORT:

```
const message = () => {
  const name = "Jesse";
  const age = 40;
  return name + ' is ' + age +
'years old.';
};

export default message;
```

Saved in another file.

Import from default exports:

```
import message from "./message.js";
```

Access the variables in another file.

Ternary Operator

The ternary operator is a simplified conditional operator like if / else.

Syntax:

```
condition ? <expression if true> : <expression if false>
```

If-else:

```
if(mark>=60)
     message=“Passed”;
else
     message=“Failed”;
```

Ternary Operator:

```
mark>=60? message="passed":
message="failed";
```

References

ES6 Tutorial

https://www.javascripttutorial.net/es6/

Difference Between Var, Let, and Const in Javascript

https://www.scaler.com/topics/javascript/difference-between-var-let-and-const/

Destructuring objects and arrays in JS

https://www.codingame.com/playgrounds/6450/destructuring-objects-and-arrays-in-js

React ES6 Tutorial

https://www.w3schools.com/react/react_es6.asp