COURSE UNIT

Bachelor of Technical-Vocational Teacher Education
MAJOR IN COMPUTER PROGRAMMING:
WEB DEVELOPMENT 1

COURSE MODULE	COURSE UNIT	WEEK
1	6	8
JavaScript (Objects and Functions)		

CHECKLIST

- ✓ Read course and unit objectives
- ✓ Read study guide prior to class attendance
- ✓ Read required learning resources; refer to unit terminologies for jargons
- ✓ Proactively participate in classroom discussions
- ✓ Participate in weekly discussion
- ✓ Answer and submit course unit tasks

UNIT EXPECTED OUTCOMES (UEOs)

At the end of this unit, the students are expected to:

Cognitive:

1. Explain JavaScript Objects and Functions.
2. Execute the different programs using JavaScript Objects and Functions.
3. Examine programs using JavaScript Objects and Functions.

Affective:

1. Listen attentively during class discussions
2. Demonstrate tact and respect when challenging other people's opinions and ideas
3. Accept comments and reactions of classmates on one's opinions openly and graciously.

Psychomotor:

1. Participate actively during class discussions and group activities
2. Express opinion and thoughts in front of the class

REQUIRED READINGS

JavaScript Tutorial. (n.d.). https://www.w3schools.com/js/default.asp

STUDY GUIDE

JavaScript

JavaScript is an Object Oriented Programming (OOP) language. A programming language can be called object-oriented if it provides four basic capabilities to developers −

- **Encapsulation** − the capability to store related information, whether data or methods, together in an object.
- **Aggregation** − the capability to store one object inside another object.
- **Inheritance** − the capability of a class to rely upon another class (or number of classes) for some of its properties and methods.
- **Polymorphism** − the capability to write one function or method that works in a variety of different ways.

JavaScript Objects

- It is just a collection of named values. These named values are usually referred to as properties of the object.
- It is an entity having state and behavior (properties and method).

In JavaScript, almost "everything" is an object.

- Booleans can be objects (if defined with the new keyword)
- Numbers can be objects (if defined with the new keyword)
- Strings can be objects (if defined with the new keyword)
- Dates are always objects
- Maths are always objects
- Regular expressions are always objects
- Arrays are always objects
- Functions are always objects
- Objects are always objects

All JavaScript values, except primitives, are objects.

A **primitive value** is a value that has no properties or methods.
A primitive data type is data that has a primitive value.

JavaScript defines 7 types of primitive data types:

- string
- number
- boolean
- null
- undefined
- symbol
- bigint

NOTE: Primitive values are immutable (they are hardcoded and cannot be changed).

- ➢ Objects are variables too. But objects can contain many values.

Object values are written as name: value pairs (name and value separated by a colon).
The named values, in JavaScript objects, are called **properties**.

- ➢ JavaScript objects are containers for named values, called properties and methods.

Methods are actions that can be performed on objects.

Creating a JavaScript Object

- Create a single object, using an object literal.
- Create a single object, with the keyword new.
- Define an object constructor, and then create objects of the constructed type.
- Create an object using Object.create().

JavaScript Functions

- ➢ a block of code designed to perform a particular task.
- ➢ executed when "something" invokes it (calls it).
- ➢ is defined with the function keyword, followed by a name, followed by parentheses ().

Function names can contain letters, digits, underscores, and dollar signs (same rules as variables).
The parentheses may include parameter names separated by commas: (parameter1, parameter2, ...) The code to be executed, by the function, is placed inside curly brackets: {}

- ➢ Function is much the same as a Procedure or a Subroutine, in other programming languages.

TERMINOLOGIES

BROWSER - a computer program with a graphical user interface for displaying and navigating between web pages.

CLIENT - a program, person or things that are capable of obtaining services provided by another program.

CODING - sometimes called computer programming, is how we communicate with computers.

DOMAIN NAME - refers to your website address. This is what users type in a browser's search bar to directly access your website.

FRAMEWORK - a layered structure indicating what kind of programs can or should be built and how they would interrelate.

FRONT-END - refers to the user interface / client-side, everything with which the user interacts.

HYPERTEXT - a word, phrase or chunk of text that can be linked to another document or text.

HYPERTEXT TRANSFER PROTOCOL (HTTP) - The communications protocol used to connect to Web servers on the Internet or on a local network (intranet).

INTERNET PROTOCOL - a set of rules governing the format of data sent over the internet or other network.

IP ADDRESS (INTERNET PROTOCOL ADDRESS) - a series of numbers that identifies any device on a network.

SEARCH ENGINE - a program that searches for and identifies items in a database that correspond to keywords or characters specified by the user, used especially for finding particular sites on the World Wide Web.

SERVER - a computer or system that provides resources, data, services, or programs to other computers, known as clients, over a network.

WEB PAGES - a hypertext document on the World Wide Web.

WEBSITE - a set of related web pages located under a single domain name, typically produced by a single person or organization.

WIREFRAME - a simplified visual guide that represents the skeletal framework of a website.

WORLD WIDE WEB - an information system on the internet which allows documents to be connected to other documents by hypertext links, enabling the user to search for information by moving from one document to another.

FURTHER READINGS

JavaScript Tutorial. (n.d.). https://www.w3schools.com/js/default.asp

UNIT TASK

- Proactively participate in classroom discussions
- Answer and submit course unit tasks

REFERENCES

An Introduction to JavaScript. (2021). https://javascript.info/intro

JavaScript basics. (2022). https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/JavaScript_basics

JavaScript Tutorial. (2022). https://www.javascripttutorial.net/

JavaScript Tutorial. (n.d.). https://www.javatpoint.com/javascript-tutorial

JavaScript Tutorial. (n.d.). https://www.tutorialrepublic.com/javascript-tutorial/

JavaScript Tutorial. (n.d.). https://www.tutorialspoint.com/javascript/index.htm

JavaScript Tutorial. (n.d.). https://www.w3schools.com/js/default.asp