

SAULT COLLEGE OF APPLIED ARTS AND TECHNOLOGY

SAULT STE. MARIE, ONTARIO



COURSE OUTLINE

COURSE TITLE: Web Scripting Languages

CODE NO. : CSD212 SEMESTER: 2

PROGRAM: All I.T. Studies Students

AUTHOR: Willem deBruyne

DATE: Jan 2015 PREVIOUS OUTLINE DATED: June 2013

APPROVED:

“Colin Kirkwood”

Dec 9/14

Dean

DATE

TOTAL CREDITS: Four

PREREQUISITE(S): CSD120

HOURS/WEEK: Four

Copyright ©2014 The Sault College of Applied Arts & Technology

Reproduction of this document by any means, in whole or in part, without prior written permission of Sault College of Applied Arts & Technology is prohibited.

For additional information, please contact Colin Kirkwood,

Dean, Environment, Technology and Business

(705) 759-2554, Ext. 2688

I. COURSE DESCRIPTION:

Students will be writing comprehensive Client-Side web based applications using JavaScript technology. Students will learn JavaScript code that will be cross-browser compatible. The course content will focus on; using JavaScript with well-formed Web pages; work with JavaScript variables and data types and learn how to use the operations that can perform them; add functions, events, and control structures; use the browser object model; ensuring data that is entered into Web forms is correct before sending to the server; use object oriented programming techniques; manipulate data in strings and arrays; saving state information. It is assumed that student has a good knowledge of XHTML.

II. LEARNING OUTCOMES AND ELEMENTS OF THE PERFORMANCE:

Upon successful completion of this course, the student will demonstrate the ability to:

1. Introduction to JavaScript

Potential Elements of the Performance:

- Understanding Web Browsers
- Creating Web Pages
- Basic HTML Syntax
- Creating an HTML Document
- Working with HTML5
- Understanding Client/ Server Architecture
- JavaScript and Client- Side Scripting
- Understanding Server- Side Scripting
- Should You Use Client- Side or Server- Side Scripting?
- Using the script Element
- Understanding JavaScript Objects
- Using the write() Method
- Case Sensitivity in JavaScript
- Adding Comments to a JavaScript Program
- Using Variables
- Assigning Variable Names
- Declaring and Initializing Variables
- Displaying Variables
- Modifying Variables
- Working with Elements and Events
- Referencing Web Page Elements
- Including a script Element for Each Code Section
- Placing JavaScript in the Document Head or Document Body

- Creating a JavaScript Source File
- Working with Libraries
- Validating Web Pages
- Writing Valid XHTML Code with JavaScript
- Validating HTML Code

2. Functions, Data Types and Operators

Potential Elements of the Performance:

- Defining Functions
- Calling Functions
- Understanding Variable Scope
- Using Built- in JavaScript Functions
- Working with Numeric Values
- Working with Boolean Values
- Working with Strings
- String Operators
- Escape Characters and Sequences
- Arithmetic Operators
- Assignment Operators
- Comparison and Conditional Operators
- Understanding Falsy and Truthy Values
- Logical Operators
- Special Operators

3. Building Arrays and Control structures

Potential Elements of the Performance:

- Declaring and Initializing Arrays
- Accessing Element Information
- Modifying Elements
- Determining the Number of Elements in an Array
- Using the Array Object
- Referencing Default Collections of Elements
- Repeating Code
- while Statements
- do/ while Statements
- for Statements
- Using continue Statements to Restart Execution
- Making Decisions
- if Statements
- if/ else Statements
- Nested if and if/ else Statements

- else if Statements
- switch Statements

4. Debugging and Error Handling

Potential Elements of the Performance:

- Recognizing Syntax Errors
- Recognizing Run- Time Errors
- Recognizing Logic Errors
- Interpreting Error Messages
- Tracing Errors with the window. alert() Method
- Tracing Errors with the console.
- Using Comments to Locate Bugs
- Combining Debugging Techniques
- Tracing Errors with Debugging Tools
- Understanding the IE, Firefox, and Chrome Debugger Windows
- Setting Breakpoints
- Clearing Breakpoints
- Stepping Through Your Scripts
- Tracing Variables and Expressions
- Examining the Call Stack
- Using the try and throw Statements
- Catching Exceptions
- Executing Final Exception Handling Tasks
- Implementing Custom Error Handling
- Checking HTML Elements
- Analyzing Logic
- Testing Statements with the Console Command Line
- Using the debugger Statement
- Using Strict Mode
- Linting
- Reloading a Web Page

5. Working with the Document Object Model (DOM) and DHTML

Potential Elements of the Performance:

- The Browser Object Model
- The Document Object Model
- The DOM and DHTML

- The DOM tree
- DOM Document Object Methods
- DOM Document Object Properties
- Accessing Elements by id Value
- Accessing Elements by Tag Name
- Accessing Elements by Class Name
- Accessing Elements by Name
- Accessing Elements with CSS Selectors
- Accessing an Element's Content
- Accessing an Element's CSS Properties
- Accessing Element Attributes
- Creating Nodes
- Attaching Nodes
- Cloning Nodes
- Inserting Nodes at Specific Positions in the Document Tree
- Removing Nodes
- Opening and Closing Windows and Tabs
- Working with Timeouts and Intervals
- The History Object
- The Location Object
- The Navigator Object
- The Screen Object

6. Enhancing and Validating Forms

Potential Elements of the Performance:

- Using JavaScript with Forms
- Referencing Forms and Form Elements
- Improving Form Usability
- Designing Forms to Collect More Accurate Content
- Programming Forms to Increase Content Accuracy
- Customizing Browser- Based Validation
- Specifying Browser- Based Validation Parameters
- Customizing Browser- Based Validation Feedback
- Programming Custom Validation
- Validating Submitted Data
- Validating Required Fields with Custom Functions
- Validating Dependent Fields with Custom Functions
- Validating Content Type with Custom Functions

III. TOPICS:

1. Introduction to Java Script
2. Functions, Data types and operators
3. Building Arrays and Control Structures
4. Debugging and Error Handling
5. Working with the Document Object Model (DOM) and DHTML
6. Enhancing and Validating Forms

IV. REQUIRED RESOURCES/TEXTS/MATERIALS:

JavaScript, Sixth Edition Sasha Vodnik and Don Gosselin
ISBN13: 978-1-305-07844- 4

V. EVALUATION PROCESS/GRADING SYSTEM:

Quizzes & Tests	70%
Assignments	<u>30%</u>
	100%

The following semester grades will be assigned to students:

Grade	Definition	Grade Point Equivalent
A+	90 – 100%	4.00
A	80 – 89%	3.00
B	70 - 79%	2.00
C	60 - 69%	1.00
D	50 – 59%	0.00
F (Fail)	49% and below	
CR (Credit)	Credit for diploma requirements has been awarded.	
S	Satisfactory achievement in field /clinical placement or non-graded subject area.	
U	Unsatisfactory achievement in field/clinical placement or non-graded subject area.	
X	A temporary grade limited to situations with extenuating circumstances giving a student additional time to complete the requirements for a course.	
NR	Grade not reported to Registrar's office.	
W	Student has withdrawn from the course without academic penalty.	

VI. OTHER EVALUATION CONSIDERATIONS

1. In order to pass this course the student must obtain an overall test/quiz average of **50%** or better, as well as, an overall assignment average of **50%** or better. A student who is not present to write a particular test/quiz, and does not notify the professor beforehand of their intended absence, may be subject to a zero grade on that test/quiz.
2. There will be **no** supplemental or make-up quizzes/tests in this course unless there are extenuating circumstances.
3. Assignments must be submitted by the due date according to the specifications of the professor. Late assignments will normally be given a mark of zero. Late assignments will only be marked at the discretion of the professor in cases where there were extenuating circumstances.
4. Any assignment/projects submissions, deemed to be copied, will result in a **zero** grade being assigned to **all** students involved in that particular incident.
5. It is the responsibility of the student to ask the professor to clarify any assignment requirements.
6. The professor reserves the right to modify the assessment process to meet any changing needs of the class.

VII. SPECIAL NOTES:**Attendance:**

Sault College is committed to student success. There is a direct correlation between academic performance and class attendance; therefore, for the benefit of all its constituents, all students are encouraged to attend all of their scheduled learning and evaluation sessions. This implies arriving on time and remaining for the duration of the scheduled session. *It is the departmental policy that once the classroom door has been closed, the learning process has begun. Late arrivers may not be granted admission to the room.*

Absences due to medical or other unavoidable circumstances should be discussed with the professor, otherwise a penalty may be assessed. The penalty depends on course hours and will be applied as follows:

Course Hours	Deduction
5 hrs/week (75 hrs)	1.0% /hr
4 hrs/week (60 hrs)	1.5% /hr
3 hrs/week (45 hrs)	2.0% /hr
2 hrs/week (30 hrs)	3.0% /hr

Final penalties will be reviewed and assessed at the discretion of the professor.

VIII. COURSE OUTLINE ADDENDUM:

This document (**CourseOutlineAddendum.docx**) can be found along with the course outline on *Desire2Learn (D2L)*.