SAN DIEGO COMMUNITY COLLEGE DISTRICT
CONTINUING EDUCATION
COURSE OUTLINE

SECTION I

SUBJECT AREA AND COURSE NUMBER

COMP 649

COURSE TITLE

Programming With Javascript II

TYPE COURSE

Non-fee

CATALOG COURSE DESCRIPTION

This course incorporates JavaScript into the web development process. Students learn about modern web architecture and how JavaScript serves as an armature for every component within that ecosystem. More importantly, students learn how to program for the Web using JavaScript. Students also learn how JavaScript can enhance a webpage, allowing additional interactivity and more precise control of page elements. Techniques used to create a website and make the content more dynamic are also covered. (FT)

LECTURE/LABORATORY HOURS

50 - 60

ADVISORIES

Completion of COMP 648 Programming with JavaScript I

INSTITUTIONAL STUDENT LEARNING OUTCOMES

1. Social Responsibility
   SDCCE students demonstrate interpersonal skills by learning and working cooperatively in a diverse environment.
2. Effective Communication
   SDCCE students demonstrate effective communication skills.
3. Critical Thinking
   SDCCE students critically process information, make decisions, and solve problems independently or cooperatively.
4. Personal and Professional Development
   SDCCE students pursue short term and life-long learning goals, mastering necessary skills and using resource management and self-advocacy skills to cope with changing situations in their lives.

CEISO Approval: 05/01/24
CIC Approval: 08/29/24
BOT Approval: 10/10/24

5. Diversity, Equity, Inclusion, Anti-racism, and Access
   SDCCE students critically and ethically engage with local and global issues using principles of equity, civility, and compassion as they apply their knowledge and skills: exhibiting awareness, appreciation, respect, and advocacy for diverse individuals, groups, and cultures.

COURSE GOALS

1. Understand the purpose of Document Object Model (DOM) scripting
2. Introduce advanced concepts in JavaScript such as Data Structures
3. Learn how to retrieve and manipulate web page content
4. Learn how to consume and display JavaScript Object Notation (JSON) data
5. Understand how to connect to and consume data from a web application programming interface (API).

COURSE OBJECTIVES

Upon successful completion, the student will be able to:

1. Explain the core concepts of DOM scripting
2. Use advanced programming concepts to build more complex programs in the web browser
3. Retrieve and manipulate web page elements, otherwise known as DOM scripting
4. Consume and display the content contained within a JSON file directly within a web page
5. Connect to, consume, and display data from a Web API

SECTION II

COURSE CONTENT AND SCOPE

1. DOM Scripting
   - 1.1. Introduction to the DOM
   - 1.2. Introduction to DOM scripting / DOM core specification
   - 1.3. Selectors
   - 1.4. DOM traversal: parent elements
   - 1.5. DOM traversal: child elements
   - 1.6. DOM traversal: sibling elements
   - 1.7. Creating, modifying, and deleting elements
   - 1.8. Element attributes
   - 1.9. DOM HTML
2. Arrays and Web Storage
   - 2.1. Working with Arrays
     - 2.1.1. Adding and removing items manually
     - 2.1.2. Processing items in an array
     - 2.1.3. Members of the array object
     - 2.1.4. Functions for adding and removing items
     - 2.1.5. Functions for counting, reversing, and sorting
     - 2.1.6. Functions for copying, slicing, and concatenating
     - 2.1.7. Functions for filtering
     - 2.1.8. Utility functions and operators

2.1.9. Two-dimensional arrays
2.2. Web Storage
3. Asynchronous Programming with JavaScript
3.1. Introduction to asynchronous programming in javascript
3.2. The XMLHttpRequest (XHR) object
3.3. Loading XML data
3.4. Loading JSON data
3.5. Using web APIs
3.6. Promises
3.7. The fetch API
3.8. Async / await
3.9. Making cross origin requests
4. Integrating Bootstrap into JavaScript Applications
4.1. A Bootstrap primer
4.2. Installing Bootstrap
4.3. Using the Bootstrap grid system
4.4. Adding buttons
4.5. Using FontAwesome for icons
4.6. Modals
4.7. Canceling a modal
4.8. Capturing an event from a modal
4.9. Toasts
4.10. Adding a search feature
4.11. Adding a sort table feature
5. Strings, Regular Expressions, Math, Numbers, and Dates
5.1. Working with strings
5.2. Working with regular expressions
5.3. Working with math
5.4. Working with numbers
5.5. Working with dates
6. Object-Oriented Programming in JavaScript
6.1. JavaScript objects
6.1.1. Object literals
6.1.2. Object constructors
6.1.3. ES6 syntax
6.2. Basic object-oriented programming concepts
6.2.1. Object constructor and ES6 class syntax
6.3. Four pillars of object-oriented programming
6.3.1. Encapsulation
6.3.2. Abstraction
6.3.3. Inheritance
6.3.4. Polymorphism
6.4. Object literal syntax

APPROPRIATE READINGS

Students may be given reading assignments from the textbook, supplemental reading assignments, industry-related periodicals or magazines, manuals, online help pages, articles posted on the Internet, and information from Web sites, online libraries, and databases. Topics

should be related to web programming techniques with JavaScript and include techniques for manipulating elements on a web page using JavaScript.

WRITING ASSIGNMENTS

Typical writing assignments may include completing assigned reports, providing written answers to assigned questions, performing internet research, and reporting on that research. An example would include a case study of JavaScript data structures that can be used in conjunction with web storage to persist a shopping cart within a web application.

OUTSIDE ASSIGNMENTS

Assignments may include, but are not limited to appropriate internet research, reading, preparing reports and studying as needed to perform successfully in class. An appropriate assignment for instance, would include the creation of a future earnings bank savings calculator web application written in JavaScript.

APPROPRIATE ASSIGNMENTS THAT DEMONSTRATE CRITICAL THINKING

Assignments which demonstrate critical thinking may include but are not limited to building a web application that connects to an open Web API, consumes the JSON data, and then displays the data within a web page. Students may also be expected to participate in online class discussion posts, in-class discussions, and project reviews.

EVALUATION

Evaluation that a student has met the course competencies will include multiple measures of performance related to the course objectives. Evaluation methods may include but are not limited to performance in a variety of activities and assignments, such as completing a research project individually or in a group, hands-on projects, demonstration of use of the internet, quizzes, class participation, written and practical tests, attendance, and punctuality.

Upon successful completion of all program courses, a Certificate of Program Completion will be issued.

METHOD OF INSTRUCTION

Methods of instruction may include, but are not limited to, lectures, in-class and online discussions, hands-on demonstrations, computer-assisted instruction, field trips, and laboratory assignments. This course, or sections of this course, may be offered through distance education.

TEXTS AND SUPPLIES

Murach's JavaScript,
Mary Delamater, Murach Books, current edition
Murach's JavaScript and jQuery,
Mary Delamater and Zak Ruvalcaba, Murach Books, current edition

Web Resources:
Udemy: Programming with JavaScript, https://www.udemy.com/course/programming-with-javascript/

Supplies:

A pen, journal (composition book), notebook paper and a soft 3-ring binder, or a one-subject 110 sheet college ruled notebook, and appropriate storage media such as a USB Drive, external hard drive, or cloud-based storage.

PREPARED BY Zak Ruvalcaba DATE March, 18, 2024

Instructors must meet all requirements stated in Policy 5500 (Student Rights, Responsibilities and Administrative Due Process), and the Attendance Policy set forth in the Continuing Education Catalog.

REFERENCES:

San Diego Community College District Policy 5500
California Community Colleges, Title 5, Section 55002
Continuing Education Catalog