

## Course Description:

Introduction to Coding (2 of 2) builds on the basic JavaScript concepts from Introduction to Coding (1 of 2). Topics include the practices of arrays, objects, functions, conditionals and loops, manipulating HTML elements, different types of code documentation, as well as giving and receiving feedback from both users and other developers. The process of solving complex problems is modeled from beginning to end as problems are broken down into smaller pieces and addressed through planning, coding, and putting the pieces together to solve the larger problem.

## Course Objectives:

- Plan and write algorithms using pseudocode.
- Create programs that include arrays and objects.
- Create programs that include conditionals and loops.
- Modify a program by using feedback from others.
- Use appropriate credit when using existing programs and media.
- Test programs systematically to identify and fix errors.
- Develop big programs by following a timeline.
- Provide documentation for programs.
- Break problems down into smaller, manageable subproblems to create programs.
- Create programs that include functions.
- Create programs that manipulate HTML elements.

## Required Materials:

None.

## Course Overview:

### Unit 1: Arrays and Objects

- Lesson 1: Work with Arrays and Data Types
  - Activity 1: Instruction: Work with Arrays and Data Types
  - Activity 2: Workbook: Work with Arrays and Data Types
  - Activity 3: Checkpoint: Work with Arrays and Data Types
- Lesson 2: Develop Programs with Arrays
  - Activity 1: Instruction: Develop Programs with Arrays
  - Activity 2: Workbook: Develop Programs with Arrays
  - Activity 3: Checkpoint: Develop Programs with Arrays
- Lesson 3: Use Objects and Arrays Together
  - Activity 1: Instruction: Use Objects and Arrays Together
  - Activity 2: Workbook: Use Objects and Arrays Together
  - Activity 3: Checkpoint: Use Objects and Arrays Together
- Lesson 4: Test and Fix Programs
  - Activity 1: Instruction: Test and Fix Programs
  - Activity 2: Workbook: Test and Fix Programs
  - Activity 3: Checkpoint: Test and Fix Programs
- Lesson 5: Write Comments in Programs

- Activity 1: Instruction: Write Comments in Programs
- Activity 2: Workbook: Write Comments in Programs
- Activity 3: Unit Review: Arrays and Objects
- Activity 4: Unit 1 Exam: Arrays and Objects

#### Unit 2: Conditionals and Loops

- Lesson 6: Use Conditional Statements
  - Activity 1: Instruction: Use Single Conditional Statements
  - Activity 2: Instruction: Use Chain of Conditional Statements
  - Activity 3: Workbook: Use Conditional Statements
  - Activity 4: Checkpoint: Use Conditional Statements
- Lesson 7: Use Loops
  - Activity 1: Instruction: Use "For" Loops
  - Activity 2: Instruction: Use Loops and Check for Errors
  - Activity 3: Workbook: Use Loops
  - Activity 4: Checkpoint: Use Loops
- Lesson 8: Use Conditionals and Loops
  - Activity 1: Instruction: Use Conditionals and Loops
  - Activity 2: Workbook: Use Conditionals and Loops
  - Activity 3: Checkpoint: Use Conditionals and Loops
- Lesson 9: Use "For...In" and "For...Of" Loops
  - Activity 1: Instruction: Use "For...In" and "For...Of" Loops
  - Activity 2: Workbook: Use "For...In" and "For...Of" Loops
  - Activity 3: Checkpoint: Use "For...In" and "For...Of" Loops
- Lesson 10: Use Break Statements in Loops
  - Activity 1: Instruction: Use Break Statements in Loops
  - Activity 2: Workbook: Use Break Statements in Loops
  - Activity 3: Unit Review: Conditionals and Loops
  - Activity 4: Unit 2 Exam: Conditionals and Loops

#### Unit 3: Functions

- Lesson 11: Do Arithmetic Using Functions
  - Activity 1: Instruction: Do Arithmetic Using Functions
  - Activity 2: Workbook: Do Arithmetic Using Functions
  - Activity 3: Checkpoint: Do Arithmetic Using Functions
- Lesson 12: Use Return Statements in Functions
  - Activity 1: Instruction: Use Return Statements in Functions
  - Activity 2: Workbook: Use Return Statements in Functions
  - Activity 3: Checkpoint: Use Return Statements in Functions
- Lesson 13: Write Algorithms Using Pseudocode
  - Activity 1: Instruction: Write Algorithms Using Pseudocode
  - Activity 2: Workbook: Write Algorithms Using Pseudocode
  - Activity 3: Checkpoint: Write Algorithms Using Pseudocode
- Lesson 14: Code Smaller Parts of a Game
  - Activity 1: Instruction: Code Some Parts of a Game
  - Activity 2: Instruction: Complete Parts of a Game
  - Activity 3: Workbook: Code Smaller Parts of a Game
  - Activity 4: Checkpoint: Code Smaller Parts of a Game

- Lesson 15: Combine Small Parts to Create a Game
  - Activity 1: Instruction: Create a Game
  - Activity 2: Instruction: Test and Refine a Game
  - Activity 3: Workbook: Combine Small Parts to Create a Game
  - Activity 4: Project: Develop a Game
  - Activity 5: Project: Develop a Game - Submission
  - Activity 6: Unit Review: Functions
  - Activity 7: Unit 3 Exam: Functions

#### Unit 4: Documentation and Other Functions

- Lesson 16: Write Supporting Documentation
  - Activity 1: Instruction: Write Supporting Documentation
  - Activity 2: Workbook: Write Supporting Documentation
  - Activity 3: Discussion: Benefits of Documenting Programs
  - Activity 4: Checkpoint: Write Supporting Documentation
- Lesson 17: Use Functions to Work with Arrays
  - Activity 1: Instruction: Use Functions to Work with Arrays
  - Activity 2: Workbook: Use Functions to Work with Arrays
  - Activity 3: Checkpoint: Use Functions to Work with Arrays
- Lesson 18: Use Functions Inside Objects
  - Activity 1: Instruction: Use Functions Inside Objects
  - Activity 2: Workbook: Use Functions Inside Objects
  - Activity 3: Checkpoint: Use Functions Inside Objects
- Lesson 19: Use Factory Functions
  - Activity 1: Instruction: Use Factory Functions
  - Activity 2: Workbook: Use Factory Functions
  - Activity 3: Checkpoint: Use Factory Functions
- Lesson 20: Use Constructor Functions
  - Activity 1: Instruction: Use Constructor Functions
  - Activity 2: Workbook: Use Constructor Functions
  - Activity 3: Unit Review: Documentation and Other Functions
  - Activity 4: Unit 4 Exam: Documentation and Other Functions

#### Unit 5: Interactive Web Pages

- Lesson 21: Replace Text Using jQuery
  - Activity 1: Instruction: Replace Text Using jQuery
  - Activity 2: Workbook: Replace Text Using jQuery
  - Activity 3: Checkpoint: Replace Text Using jQuery
- Lesson 22: Add Text Using jQuery
  - Activity 1: Instruction: Add Text Using jQuery
  - Activity 2: Workbook: Add Text Using jQuery
  - Activity 3: Checkpoint: Add Text Using jQuery
- Lesson 23: Set a Timeout in JavaScript
  - Activity 1: Instruction: Set a Timeout in JavaScript
  - Activity 2: Workbook: Set a Timeout in JavaScript
  - Activity 3: Checkpoint: Set a Timeout in JavaScript
- Lesson 24: Animate Elements in Web Pages
  - Activity 1: Instruction: Animate Elements in Web Pages

- Activity 2: Workbook: Animate Elements in Web Pages
- Activity 3: Checkpoint: Animate Elements in Web Pages
- Lesson 25: Respond to User Actions
  - Activity 1: Instruction: Respond to User Actions
  - Activity 2: Workbook: Respond to User Actions
  - Activity 3: Unit Review: Interactive Web Pages
  - Activity 4: Unit 5 Exam: Interactive Web Pages

## Unit 6: Put Everything Together

- Lesson 26: Use Existing Code and Media
  - Activity 1: Instruction: Use Existing Code and Media
  - Activity 2: Workbook: Use Existing Code and Media
  - Activity 3: Checkpoint: Use Existing Code and Media
- Lesson 27: Write the Algorithm of a Game
  - Activity 1: Instruction: Write the Algorithm of a Game
  - Activity 2: Workbook: Write the Algorithm of a Game
  - Activity 3: Checkpoint: Write the Algorithm of a Game
- Lesson 28: Create Smaller Parts of a Game
  - Activity 1: Instruction: Create Smaller Parts of a Game
  - Activity 2: Workbook: Create Smaller Parts of a Game
  - Activity 3: Checkpoint: Create Smaller Parts of a Game
- Lesson 29: Create the Game
  - Activity 1: Instruction: Create the Game
  - Activity 2: Workbook: Create the Game
  - Activity 3: Checkpoint: Create the Game
- Lesson 30: Use Feedback to Revise the Game
  - Activity 1: Instruction: Use Feedback to Revise the Game
  - Activity 2: Workbook: Use Feedback to Revise the Game
  - Activity 3: Project: Create an Interactive Game
  - Activity 4: Project: Create an Interactive Game- Submission
  - Activity 5: Final Review
  - Activity 6: Final Exam