



# CodeHS

## Intro to Computer Science with JavaScript Syllabus

Collie: One Semester, 55-65 hours

### Course Overview and Goals

The CodeHS introduction to computer science curriculum teaches the foundations of computer science and basic programming, with an emphasis on helping students develop logical thinking and problem solving skills.

**Learning Environment:** The course utilizes a blended classroom approach. The content is fully web-based, with students writing and running code in the browser. Teachers utilize tools and resources provided by CodeHS to leverage time in the classroom and give focused 1-on-1 attention to students. Each unit of the course is broken down into lessons. Lessons consist of video tutorials, short quizzes, example programs to explore, and written programming exercises. Each unit ends with a comprehensive unit test that assesses student's mastery of the material from that unit.

**Programming Environment:** Students write and run JavaScript programs in the browser using the CodeHS editor.

**More information:** Browse the content of this course at <https://codehs.com/course/2359>

### Prerequisites

The Intro to Computer Science in JavaScript course is designed for complete beginners with no previous background in computer science. The course is highly visual, dynamic, and interactive, making it engaging for new coders.

### Course Breakdown

#### Unit 1: Introduction to Programming in JavaScript with Karel the Dog (3 weeks/15 hours)

Browse the full content of this unit at <https://codehs.com/library/course/2359/module/3477>

Objectives / Topics Covered	<ul style="list-style-type: none"><li>● Commands</li><li>● Defining vs. Calling Methods</li><li>● Designing methods</li><li>● Program entry points</li><li>● Control flow</li><li>● Looping</li><li>● Conditionals</li><li>● Classes</li><li>● Commenting code</li><li>● Preconditions and Postconditions</li><li>● Top Down Design</li></ul>
Assignments / Labs	<ul style="list-style-type: none"><li>● 26 Karel programming exercises in total</li><li>● Program-specific tasks for Karel the Dog<ul style="list-style-type: none"><li>○ Example Exercise: Pyramid of Karel</li></ul></li></ul>

	<p>Write a program to have Karel build a pyramid. There should be three balls on the first row, two in the second row, and one in the third row.</p> <ul style="list-style-type: none"> <li>● Teach Karel new commands like <code>turnRight()</code> or <code>makePancakes()</code> <ul style="list-style-type: none"> <li>○ Example Exercise: Pancakes Karel is the waiter. He needs to deliver a stack of pancakes to the guests on the 2nd, 4th, and 6th avenue. Each stack of pancakes should have three pancakes. Create a method called <code>makePancakes()</code> to help Karel solve this problem.</li> </ul> </li> <li>● Solve large Karel problems by breaking them down into smaller, more manageable problems using Top Down Design <ul style="list-style-type: none"> <li>○ Example Exercise: The Two Towers In this program, Karel should build two towers of tennis balls. Each tower should be 3 tennis balls high. At the end, Karel should end up on top of the second tower, facing East.</li> </ul> </li> <li>● Using control structures and conditionals to solve general problems <ul style="list-style-type: none"> <li>○ Example Exercise: Random Hurdles Write a program that has Karel run to the other side of first street, jumping over all of the hurdles. However, the hurdles can be in random locations. The world is fourteen avenues long.</li> <li>○ Example Exercise: Super Cleanup Karel Karel's world is a complete mess. There are tennis balls all over the place, and you need to clean them up. Karel will start in the bottom left corner of the world facing east, and should clean up all of the tennis balls in the world. This program should be general enough to work on any size world with tennis balls in any locations.</li> </ul> </li> </ul>
--	---

## Unit 2: Karel Challenges (1.5 weeks, 7 hours)

Browse the full content of this unit at <https://codehs.com/library/course/2359/module/3478>

Objectives / Topics Covered	<ul style="list-style-type: none"> <li>● Solving large and more complex problems using Karel</li> </ul>
Assignments / Labs	<ul style="list-style-type: none"> <li>● 5 Karel challenges to tie everything learned in the Karel module together</li> <li>● Several Karel challenges to tie everything learned in the Karel module together</li> </ul>

## Unit 3: Javascript & Graphics (1 week/5 hours)

Browse the full content of this unit at <https://codehs.com/library/course/2359/module/3479>

Objectives / Topics Covered	<ul style="list-style-type: none"> <li>● Variables</li> <li>● User Input</li> <li>● Arithmetic Expressions</li> <li>● Graphics</li> </ul>
Assignments / Labs	<ul style="list-style-type: none"> <li>● 7 JavaScript &amp; graphics programming exercises in total</li> <li>● Using variables and getting user input using JavaScript <ul style="list-style-type: none"> <li>○ Example Exercise: Dinner Plans Prompt the user for their name, then ask them what time you should meet for dinner.</li> </ul> </li> </ul>

	Greet them by name and tell them you will meet them at the time they specified!
--	---

#### Unit 4: JavaScript Control Structures (3 weeks/15 hours)

Browse the full content of this unit at <https://codehs.com/library/course/2359/module/3480>

Objectives / Topics Covered	<ul style="list-style-type: none"> <li>● Booleans</li> <li>● For Loops</li> <li>● Conditionals</li> <li>● Nested Control Structures</li> <li>● While Loops</li> </ul>
Assignments / Labs	<ul style="list-style-type: none"> <li>● 22 control structures programming exercises in total</li> <li>● Using comparison and logical operators to control the flow of the program <ul style="list-style-type: none"> <li>○ Example Exercise: Inventory Write a program that keeps track of a simple inventory for a store. While there are still items left in the inventory, ask the user how many items they would like to buy. Then print out how many are left in inventory after the purchase. You should use a while loop for this problem. Make sure you catch the case where the user tries to buy more items than there are in the inventory. In that case, you should print a message to the user saying that their request isn't possible.</li> </ul> </li> <li>● Using for loops <ul style="list-style-type: none"> <li>○ Example Exercise: All Dice Values <ul style="list-style-type: none"> <li>■ Write a program that prints all possible dice rolls with 2 dice. To do so, you should use a double for loop. Hint: You can't use i for both for loops.</li> </ul> </li> </ul> </li> <li>● Drawing basic graphics using JavaScript <ul style="list-style-type: none"> <li>○ Example Exercise: Caterpillar This graphics program should draw a caterpillar. A caterpillar has NUM_CIRCLES circles. Every other circle is a different color, the even circles are red, and the odd circles are green (by even we mean when i is an even number). Use a for loop to draw the caterpillar, centered vertically in the screen. Also, be sure that the caterpillar is still drawn across the whole canvas even if the value of NUM_CIRCLES is changed.</li> </ul> </li> </ul>

#### Unit 5: Functions and Parameters (2 weeks/10 hours)

Browse the full content of this unit at <https://codehs.com/library/course/2359/module/3481>

Objectives / Topics Covered	<ul style="list-style-type: none"> <li>● Functions with and without parameters</li> <li>● Functions with and without return values</li> <li>● Nested Control Structures</li> <li>● Local variables and scope</li> </ul>
Assignments / Labs	<ul style="list-style-type: none"> <li>● 14 functions programming exercises in total</li> <li>● Using various kinds of functions such as functions with and without parameters, and functions with and without return values <ul style="list-style-type: none"> <li>○ Example Exercise: Vertical Lines Write a function that draws vertical lines on the graphics canvas. If a</li> </ul> </li> </ul>

	<p>line is vertical, then the x-values for the endpoints are the same. The parameters to your function should be the x location, and the length, and all of your lines should start at y position 0.</p> <ul style="list-style-type: none"> <li>○ Example Exercise: Is it even? Write a function called isEven that returns a boolean of whether or not a value is even or odd. The isEven function should not print anything out or return a number. It should only take in a number and return a boolean.</li> </ul> <p>Once you've written this function, write a program that asks the user for integers and prints whether the number they entered is even or odd using your isEven function. You should let the user keep entering numbers until they enter the SENTINEL given.</p>
--	---

**Unit 6: Final Project (1-2 weeks, 5-10 hours)**

Browse the full content of this unit at <https://codehs.com/library/course/2359/module/3481>

Objectives / Topics Covered	<ul style="list-style-type: none"> <li>● Review of JS graphics</li> <li>● Review of functions</li> <li>● Review of top-down design</li> </ul>
Assignments / Labs	<ul style="list-style-type: none"> <li>● Final Project: Students will use what they learned in the course to brainstorm, plan, and implement a JS graphics program that draws a picture. Students will go through a brainstorming process, and then select their final idea. They will establish milestones in the completion of their project, pseudocode their project before they finally implement the code to complete their goal.</li> </ul>

**Optional Supplemental Materials (Remainder of school year)**

Objectives / Topics Covered	<ul style="list-style-type: none"> <li>● Extra practice with: <ul style="list-style-type: none"> <li>○ Karel</li> <li>○ Basic JavaScript</li> <li>○ JavaScript functions</li> <li>○ Graphics</li> </ul> </li> <li>● Basic Data Structures in JavaScript</li> <li>● Game design/ Animation</li> <li>● Music Visualization</li> </ul>
Assignments / Labs	<ul style="list-style-type: none"> <li>● Several additional exercises and large projects covering the topics listed above</li> </ul>