CodeHS

Intro JavaScript (Golden) K-12 CS Framework Alignment Overview

The CodeHS Introduction to Computer Science in JavaScript curriculum teaches the foundations of computer science and basic programming, with an emphasis on helping students develop logical thinking and problem solving skills. This document is an overview of how the Introduction to Computer Science in JavaScript course aligns with the K-12 CS Framework's concepts and practices.

K-12 CS Framework Practices

Fostering an Inclusive Computing Culture

Through collaborative projects, students learn to include a diverse set of perspectives in the development process and functionality of their computational artifacts.

Collaborating Around Computing

The CodeHS course provides several opportunities for students to create computational artifacts both individually and in teams. Several small group and full class activities are included in the curriculum. CodeHS provides guidance for implementing pair programming in the classroom.

Recognizing and Defining Computational Problems

As the problems in the course get more and more complex, the course teaches students to break down large, complex problems into manageable subproblems that can be solved independently. The course continually emphasizes decomposition and top down design.

Developing and Using Abstractions

The Introductory JavaScript course teaches students to develop abstractions to manage the complexity of their programs. Students develop and use their own abstractions to generalize their solutions and simplify the development process.

Creating Computational Artifacts

This course has a huge emphasis on creating computational artifacts. In each lesson of the course, students develop their own computational artifacts both for creative expression as well as to solve computational problems, for over a hundred hours of hands on creation in total.

Testing and Refining Computational Artifacts

Testing and refining computational artifacts is an important part of the development process that is emphasized in the Introduction to JavaScript course. The course teaches students to test their solutions, identify and fix errors, use debugging tools, and consider all edge cases when developing and testing their programs.

Communicating About Computing

This course gives students several opportunities to communicate their ideas and solutions to others. Students are encouraged to describe and justify their solutions to their teachers and their peers. Students are also required to document their code to communicate how it works, and any assumptions their solution makes.

K-12 CS Framework Concepts

Computing Systems

Troubleshooting computing systems is a core concept of the Introductory JavaScript course. Computing systems might not work as expected because of problems in the software. Students are expected to identify problems in their programs and fix them. Errors are opportunities to learn from mistakes.

Data and Analysis

Storing, transforming, and visualizing data are all taught as part of the Introduction to JavaScript course. Students learn to use various types of data structures to store data, as well as how to select the proper data structure to model a problem. Students have the option to write programs that manipulate data, find elements in data structures, remove elements from data structures, and even visualize music data in the Supplemental Units of the course.

Algorithms and Programming

Algorithms and programming are the central focus of this course. Students learn the core principles of developing their own algorithms and implementing them in the JavaScript programming language. Algorithms, variables, control, modularity, and program development are all taught in this course.

Impact of Computing

Computing has had significant impacts in several fields. In this course, students learn first hand the impacts computing has on animation, gaming, and digital art. Students also learn the ethical considerations of sharing their code with others, and searching online for solutions to CodeHS exercises.