

Welcome to Computer Science Discoveries

Code.org's CS Discoveries is an introductory, classroom-based course appropriate for 6-10th grade students. The course aims to empower students to create authentic artifacts and engage with computer science as a medium for creativity, communication, problem solving, and fun. CS Discoveries (CSD) takes a wide lens on computer science by covering topics such as programming, physical computing, web development, design, and data. The course inspires students as they build their own websites, apps, games, and physical computing devices. Our curriculum is available at no cost for anyone, anywhere in the world and can be accessed at code.org/csd.

Curriculum At-a-Glance

CSD is designed with the new-to-CS student and teacher in mind and can be taught as a year-long course, a semester-long course (3-5 hours per week of instruction for 9+ weeks), or in a more modular approach, teaching the units that fit the teacher's needs. See our [Implementation Guide](#) for more details on ways to implement CSD. When teaching these units as a full-year course, the following sequence will satisfy all of the middle school CSTA Standards.

Problem Solving and Computing	Students learn about the problem-solving process, the input-output-store-process model of a computer, and how computers help humans solve problems. Students end the unit by proposing their own app to solve a problem.
Web Development	Students learn to create websites using HTML and CSS inside Code.org's Web Lab environment. Throughout the unit, students consider questions of privacy and ownership on the internet as they develop their own personal websites.
Interactive Animations and Games	Students learn fundamental programming constructs and practices in the JavaScript programming language while developing animations and games in Code.org's Game Lab environment. Students end the unit by designing their own animations and games.
The Design Process	Students apply the problem solving process to the problems of others, learning to empathize with the needs of a user and design solutions to address those needs. During the second half of the unit, students form teams to prototype an app of their own design, first on paper and eventually in Code.org's App Lab environment.
Data and Society	Students explore different systems used to represent information in a computer and the challenges and trade-offs posed by using them. In the second half of the unit, students learn how collections of data are used to solve problems and how computers help to automate the steps of this process.
Creating Apps with Devices	Students use Code.org's App Lab environment, along with either the Adafruit Circuit Playground or the BBC micro:bit, to explore the relationship between hardware and software. Students develop prototypes that mirror existing innovative computing platforms, before ultimately designing and prototyping one of their own
Optional Unit	
AI and Machine	Students learn how machine learning can be used to solve problems by preparing data,

Learning

training a machine learning model, then testing and evaluating the model for accuracy and bias. Students use Code.org's AI Lab environment to train machine learning models, then import their models into App Lab to create apps that solve problems