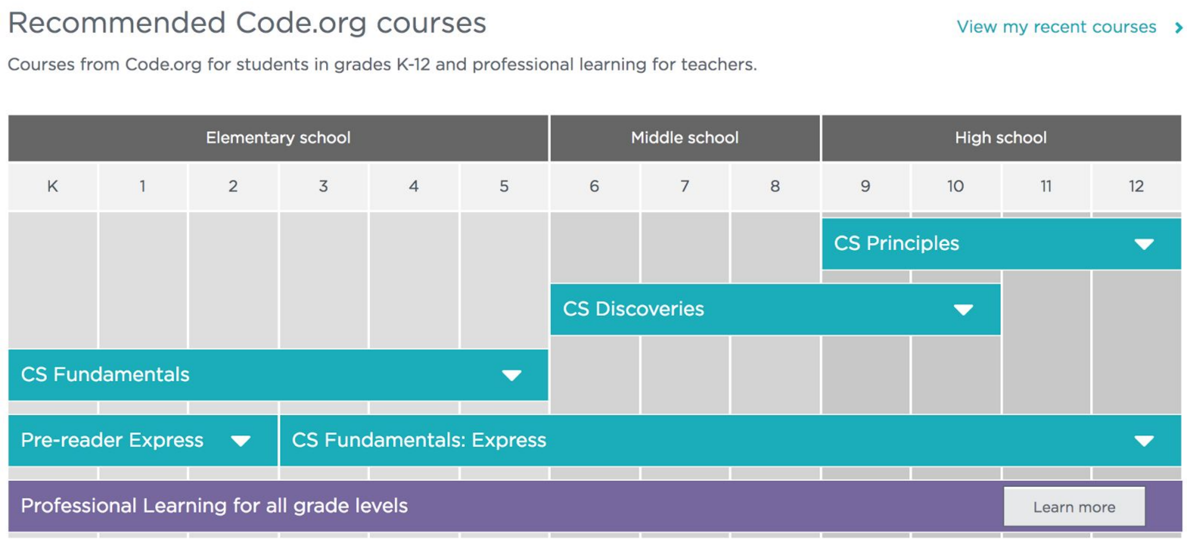
CODING

Coding should be introduced from the Stage 5 through to the College.
Students can work through the various levels at their own pace as all the necessary resources can be found online.



Recommended Code.org courses

View my recent courses ›

Courses from Code.org for students in grades K-12 and professional learning for teachers.

Elementary school						Middle school			High school			
K	1	2	3	4	5	6	7	8	9	10	11	12
									CS Principles ▼			
						CS Discoveries ▼						
CS Fundamentals ▼												
Pre-reader Express ▼			CS Fundamentals: Express ▼									
Professional Learning for all grade levels											Learn more	

STAGE 5 - COURSE A

https://studio.code.org/s/coursea

Course A

Course A offers computer science curriculum for beginning readers around the kindergarten age range. Students will learn to program using commands like loops and events. The lessons featured in this course also teach students to collaborate with others meaningfully, investigate different problem-solving techniques, persist in the face of difficult tasks, and learn about internet safety. By the end of this course, students create their very own custom game or story from Play Lab that they can share.

Lesson 1: Debugging: Unspotted Bugs

Bug | Debugging | Persistence | Unplugged

Teacher Links: Teacher Prep Guide | Storybook (PDF) | Reflection Journal Student Links: Online Story

Lesson 2: Persistence & Frustration: Stevie and the Big Project

Fail | Frustrated | Persistence | Unplugged

Teacher Links: Teacher Prep Guide | Storybook (PDF) | Reflection Journal Student Links: Online Story

Lesson 3: Real-Life Algorithms: Plant a Seed

Unplugged | Algorithms

Teacher Links: Teacher Video | Worksheet | Assessment | Reflection Journal

Lesson 4: Learn to Drag and Drop

Click | Double-Click | Drag | Drop | Pair Programming

Teacher Links: Teacher Video | Website | Manipulatives | Lesson Recommendations | Teacher Prep Guide | Reflection Journal Student Links:Student Video

Lesson 5: Programming Unplugged: Happy Maps

Unplugged | Algorithms | Sequencing

Teacher Links: Worksheet | Manipulatives | Teacher Video | Reflection Journal

Lesson 6: Programming in Maze

Algorithms | Debugging | Program | Programming | Widget - Text Compression

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 7: Common Sense Education: Going Places Safely

Common Sense Education | Unplugged

Teacher Links: Teacher Video | Lesson Video | Assessment | Website | Reflection Journal | Resource List

Lesson 8: Loops Unplugged: Happy Loops

Unplugged | Loop | Repeat

Teacher Links: Worksheet | Worksheet | Manipulatives | Manipulatives | Reflection Journal

Lesson 9: Loops in Collector

Loop | Collector

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 10: Loops in Artist

Loop | Artist

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 11: Events Unplugged: The Big Event

Unplugged | Event

Teacher Links: Teacher Video | Worksheet | Assessment | Reflection Journal

Lesson 12: Events in Play Lab

Play Lab | Event

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

GRADE 1 - COURSE B

https://studio.code.org/s/courseb

Course B

Course B was developed with first graders in mind. Tailored to a novice reading level, this course also assumes limited knowledge of shapes and numbers.

At the moment, Course B closely parallels Course A, but provides more complex unplugged activities and more variety in puzzles. Students will learn the basics of programming, collaboration techniques, investigation and critical thinking skills, persistence in the face of difficulty, and internet safety. At the end of this course students will create their very own custom game from Play Lab that they can share with a link.

Lesson 1: Debugging: Unspotted Bugs

Unplugged | Bug | Debugging | Persistence

Teacher Links: Teacher Prep Guide | Storybook (PDF) | Reflection Journal Student Links: Online Story | Student Video

Lesson 2: Persistence & Frustration: Stevie and the Big Project

Unplugged | Fail | Frustrated | Persistence

Teacher Links: Teacher Prep Guide | Storybook (PDF) | Reflection Journal Student Links: Online Story

Lesson 3: Real-Life Algorithms: Plant a Seed

Unplugged | Algorithms

Teacher Links: Worksheet | Assessment | Teacher Video

Lesson 4: Learn to Drag and Drop

Click | Double-Click | Drag | Drop | Pair Programming

Teacher Links: Website | Manipulatives | Lesson Recommendations | Teacher Prep Guide | Reflection Journal Student Links: Student Video

Lesson 5: Common Sense Education: Your Digital Footprint

Common Sense Education | Unplugged

Teacher Links: Teacher Video | Lesson Video | Worksheet | Assessment | Website | Reflection Journal | Resource List

Lesson 6: Programming Unplugged: My Robotic Friends

Algorithms | Debugging | Unplugged

Teacher Links: Teacher Prep Guide | Video | Symbol Key | Manipulatives | Manipulatives | Reflection Journal

Lesson 7: Programming in Maze

Algorithms | Debugging | Program | Programming

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 8: Programming in Star Wars

Programming | Maze

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 9: Loops Unplugged: My Loopy Robotic Friends

Unplugged | Loop | Repeat

Teacher Links: Video | Teacher Prep Guide | Teacher Prep Guide | Reflection Journal

Lesson 10: Loops in Collector

Loop | Collector

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 11: Loops in Artist

Loop | Artist

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 12: Events Unplugged: The Big Event

Event | Unplugged

Teacher Links: Teacher Video | Worksheet | Assessment | Reflection Journal

Lesson 13: Events in Play Lab

Event | Play Lab

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

GRADE 2 - COURSE C

https://curriculum.code.org/csf/coursec/

Course C

Course C was developed for students in and around the second grade. It uses a limited understanding of shapes and elementary math concepts.

Students will create programs with loops, events, and conditionals. They will translate their initials into binary, investigate different problem-solving techniques, and discuss how to respond to cyberbullying. By the end of the course, students will create interactive games that they can share. Each concept in Course C is taught from the beginning, graduating toward experiences that allow for growth and creativity to provide all students a rich and novel programming experience.

Lesson 1: Building a Foundation

Unplugged

Teacher Links: Teacher Video | Lesson in Action Video | Teacher Prep Guide | Reflection Journal

Lesson 2: Programming in Maze

Programming | Algorithms | Maze | Sequencing

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 3: Debugging in Maze

Debugging | Bug | Maze

Teacher Links: Lesson Recommendations | Reflection Journal Student Links: Student Handout

Lesson 4: Real-Life Algorithms: Paper Airplanes

Unplugged | Algorithms

Teacher Links: Teacher Video | Lesson in Action Video | Worksheet | Assessment | Reflection Journal

Lesson 5: Programming in Collector

Collector | Program | Programming

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 6: Programming in Artist

Artist | Sequencing

Teacher Links: Website | Lesson Recommendations | Reflection Journal Student Links: Student Video | Student Video | Student Handout

Lesson 7: Getting Loopy

Unplugged | Loops

Teacher Links: Teacher Video | Worksheet | Assessment | Reflection Journal

Lesson 8: Loops in Star Wars

Loops | Maze

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 9: Loops in Artist

Loop | Artist

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 10: Loops in Harvester

Loops | Harvester

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 11: Events Unplugged: The Big Event

Unplugged | Events

Teacher Links: Teacher Video | Worksheet | Assessment | Reflection Journal

Lesson 12: Build a Flappy Game

Flappy | Event

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 13: Events in Play Lab

Play Lab | Event

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 14: Common Sense Education: Screen Out the Mean

Common Sense Education | Cyberbullying | Unplugged

Teacher Links: Teacher Prep Guide | Website | Reflection Journal | Resource List Student Links: Student Handout

Lesson 15: Binary Bracelets

Unplugged | Binary

Teacher Links: Teacher Video | Lesson in Action Video | Worksheet | Assessment | Reflection Journal Student Links: Student Video

GRADE 3 - COURSE D

https://curriculum.code.org/csf/coursed/

Course D

Course D was created for students who read at roughly a third grade level. Angles and mathematical concepts are introduced with helpful videos and hints.

The course begins with a review of the concepts found in Courses A, B, and C. This review helps introduce or refresh basic ideas such as repeat loops and events. Students will develop their understanding of algorithms, nested loops, while loops, conditionals, and events. Lessons on digital citizenship are also included. This course is crafted to build a strong foundation of basic concepts before opening up to a wide range of new and exciting topics.

Lesson 1: Programming: Graph Paper Programming

Unplugged | Programming | Program

- **Warm Up (20 min)**
- **Main Activity (20 min)**
- **Wrap Up (15 min)**
- **Assessment (10 min)**
- **Extended Learning**

In this lesson, you will program your friend to draw pictures!

Teacher Links: Teacher Video | Lesson in Action Video | Worksheet | Assessment | Reflection Journal

Lesson 2: Sequences in Maze

Sequencing | Debugging | Loops | Angry Bird | Collector | Artist | Harvester

- **Warm Up (10 min)**
- **Bridging Activities - Programming (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (10 min)**

This lesson will give you practice in the skills you will need for this course.

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 3: Events in Bounce

Event | Bounce

- **Warm Up (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (10 min)**
- **Extended Learning**

Ever wish you could play video games in school? In this lesson, you will get to make your own!

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 4: Nested Loops

Nested Loops | Loops | Bee | Maze

- **Warm Up (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Loops inside loops inside loops. What does this mean? This lesson will teach you what happens when you place a loop inside another loop.

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 5: Nested Loops in Artist

Nested Loops | Loops | Artist

- **Warm Up (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**
- **Extended Learning**

More nested loops! This time, you get to make some AMAZING drawing with nested loops.

Teacher Links: Website | Lesson Recommendations | Reflection Journal Student Links: Student Handout | Student Video

Lesson 6: Nested Loops with Frozen

Loop | Nested Loop | Frozen

- **Warm Up (15)**
- **Main Activity (30)**
- **Wrap Up (15)**

Now that students know how to layer their loops, they can create so many beautiful things. This lesson will take students through a series of exercises to help them create their own portfolio-ready images using Anna and Elsa's excellent ice-skating skills!

Teacher Links: Reflection Journal | Lesson Recommendations | Website

Lesson 7: Debugging Unplugged: Relay Programming

Unplugged | Relay Programming | Algorithms

- **Warm Up (15 min)**
- **Main Activity (15 min)**
- **Wrap Up (15 min)**
- **Assessment (10 min)**
- **Extended Learning**

Remember at the beginning of the course when you made drawings with code? In this lesson, you will be working with a team to do something very similar!

Teacher Links: Teacher Video | Teacher Prep Guide | Assessment | Reflection Journal

Lesson 8: Debugging in Collector

Debugging | Bug | Collector | Laurel

- **Warm Up (15 min)**
- **Bridging Activities - Debugging (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Have you ever run into problems while coding? In this lesson, you will learn about the secrets of debugging. Debugging is the process of finding and fixing problems in your code.

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 9: While Loops in Farmer

While Loops | Loops | Farmer

- **Warm Up (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**
- **Extended Learning**

Loops are so useful in coding. This lesson will teach you about a new kind of loop: while loops!

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 10: If/Else: Conditionals with Cards

Conditionals | Unplugged

- **Warm Up (20 min)**
- **Main Activity (20 min)**
- **Wrap Up (15 min)**
- **Assessment (5 min)**
- **Extended Learning**

It's time to play a game where you earn points only under certain conditions!

Teacher Links: Teacher Video | Lesson in Action Video | Teacher Prep Guide | Assessment | Assessment Video | Reflection Journal

Lesson 11: Conditionals in Bee

Conditional | Bee | Maze

- **Warm Up (10 min)**
- **Bridging Activity - Conditionals (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**
- **Extended Learning**

Now that you understand conditionals, it's time to program Bee to use them when collecting honey and nectar.

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 12: Conditionals & Loops in Maze

Conditional | Loop | Maze | Angry Bird | Zombie

- **Warm Up (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**
- **Extended Learning**

You can do some amazing things when you use conditionals and loops together!

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 13: Conditionals & Loops in Farmer

Conditional | Loop | Farmer

- **Warm Up (5 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

It's not always clear when to use each conditional. This lesson will help you get practice deciding what to do.

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 14: Digital Citizenship: Practicing

Common Sense Education | Unplugged

- **Warm Up (15 min)**
- **Main Activity (35 - 40 min)**
- **Wrap Up (15 min)**
- **Assessment (5 min)**

- **Extended Learning**

Some information is not safe to share online. This lesson will help you learn the difference between safe and private information.

Teacher Links: Teacher Video | Manipulatives | Assessment | Website | Reflection Journal | Resource List

Lesson 15: Build a Play Lab Game

Play Lab | Event

- **Warm Up (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**
- **Extended Learning**

This lesson will guide you through making your very own video game.

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 16: Beyond Programming: Binary

Binary | Unplugged

- **Warm Up (10 min)**
- **Main Activity (20 min)**
- **Wrap Up (10 min)**
- **Assessment (10 min)**
- **Extended Learning**

Learn how computers store pictures using simple ideas like on and off.

Teacher Links: Teacher Video | Worksheet | Assessment | Reflection Journal

Lesson 17: Binary Artist

Binary | Artist

- **Warm Up (15)**
- **Main Activity (30)**
- **Wrap Up (15)**

This series of online lessons will have students learning to make images using on and off.

Teacher Links: Website

Lesson 18: Building a Foundation

Unplugged | Persistence | Frustration

- **Warm Up (20 min)**
- **Main Activity (30 min)**
- **Wrap Up (10 min)**
- **Extended Learning**

Teacher Links: Teacher Video | Lesson in Action Video | Teacher Prep Guide

Lesson 19: Loops in Artist

Loop | Artist

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Teacher Links: Website | Lesson Recommendations | Reflection Journal

GRADE 4 - COURSE E

https://curriculum.code.org/csf/coursee/

Course E

Created with fourth grade students in mind, this course begins with a brief review of concepts previously taught in courses C and D. This introduction is intended to inspire beginners and remind the experts of the wonders of computer science. Students will practice coding with algorithms, loops, conditionals, and events before they are introduced to functions. At the end of the course, students will have the opportunity to create a capstone project that they can proudly share with peers and loved ones.

Lesson 1: Programming: My Robotic Friends

Unplugged | Loop | Repeat

- **Warm Up (5 min)**
- **Main Activity (45 min)**
- **Wrap Up (10 min)**

Teacher Links: Teacher Prep Guide | Video | Symbol Key

Lesson 2: Sequences in Maze

Sequencing | Debugging | Maze

- **Warm Up (10 min)**
- **Bridging Activities - Programming (10 min)**
- **Main Activity (30 min)**

- **Wrap Up (10 min)**

Lessons 1-9 are considered ramp-up lessons. If you feel that the first few activities are too simple for your class, feel free to pick and choose your favorites or skip to lesson #10.

Teacher Links: Website | Manipulatives | Lesson Recommendations

Lesson 3: Building a Foundation

Unplugged | Persistence | Frustration

- **Warm Up (20 min)**
- **Main Activity (20 min)**
- **Wrap Up (10 min)**
- **Extended Learning**

Teacher Links: Teacher Video | Lesson in Action Video | Teacher Prep Guide

Lesson 4: Debugging in Scrat

Debugging | Ice Age

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (5 - 10 min)**
- **Extended Learning**

Teacher Links: Website | Lesson Recommendations | Reflection Journal Student Links: Student Handout

Lesson 5: Programming in Artist

Program | Programming | Artist

- **Warm Up (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (10 - 15 min)**
- **Extended Learning**

Teacher Links: Website | Lesson Recommendations | Reflection Journal Student Links: Student Video | Student Video | Student Handout

Lesson 6: My Loopy Robotic Friends

Unplugged | Loop | Repeat

- **Warm Up (10 - 15 min)**
- **Main Activity (15 - 20 min)**
- **Wrap Up (10 min)**

- **Extension Activities**

Teacher Links: Video | Teacher Prep Guide | Symbol Key

Lesson 7: Loops in Artist

Loop | Artist

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 8: Nested Loops

Nested Loops | Loops | Bee | Maze

- **Warm Up (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 9: Nested Loops with Frozen

Loop | Nested Loop | Frozen

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 10: Algorithms: Dice Race Unplugged

Unplugged | Dice Race | Algorithms

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**
- **Extended Learning**

Learn how to describe a game from the computer's point of view in this lesson on algorithms.

Teacher Links: Teacher Video | Worksheet | Assessment | Reflection Journal

Lesson 11: Introduction to Online Puzzles

Sequencing | Debugging | Loop | Angry Bird | Maze | Artist

- **Warm Up (15 min)**
- **Bridging Activity (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

This introduction is very fast in pace. If you feel that your class could benefit from a more in-depth introduction to computer science, please begin with the ramp-up activities of lessons 1-9.

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 12: Conditionals in Farmer

Conditionals | Farmer

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**
- **Extended Learning**

You will get to tell the computer what to do under certain conditions in this fun and challenging series.

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 13: Digital Citizenship

Common Sense Education | Personal Information | Private Information | Identity Theft

- **Warm Up (5 min)**
- **Main Activity (35 min)**
- **Wrap Up (15 min)**
- **Assessment (10 min)**

The internet is fun and exciting, but it's important to stay safe too. This lesson teaches you the difference between information that is safe to share and information that is private.

Teacher Links: Teacher Prep Guide | Website | Reflection Journal | Resource List

Lesson 14: Build a Star Wars Game

Star Wars | Event

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Feel the force as you build your own Star Wars game in this lesson.

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 15: Functions: Songwriting Unplugged

Unplugged | Function

- **Warm Up (20 min)**
- **Main Activity (20 min)**
- **Wrap Up (5 min)**
- **Assessment (5 min)**
- **Extended Learning**

Even rockstars need programming skills. This lesson will teach you about functions using lyrics from songs.

Teacher Links: Teacher Video | Lesson in Action Video | Worksheet | Assessment | Reflection Journal

Lesson 16: Functions in Artist

Function | Artist

- **Warm Up (15 min)**
- **Bridging Activity - Functions (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**
- **Extended Learning**

Make complex drawings more easily with functions!

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 17: Functions in Bee

Function | Bee

- **Warm Up (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Don't write too much code to gather all of the nectar and honey. Use functions instead!

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 18: Functions in Farmer

Function | Farmer

- **Warm Up (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Functions will save you lots of work as you help the farmer with her harvest!

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 19: Determine the Concept

Bee

- **Warm Up (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

We aren't giving away any secrets! This lesson could use any of the skills you've learned so far.

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 20: Build a Play Lab Game

Event | Play Lab

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Practice making games to share with your friends and family.

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 21: Explore Project Ideas

Project | Define | Prepare | Try | Reflect

- **Day 1 - Explore Project Ideas (45 min)**
- **Day 2 - The Design Process (45 min)**
- **Day 3 - Build Your Project (45 min)**
- **Day 4 & 5 - Present Your Project (45 min each)**
- **Extension Activity**

Here are several games and drawings. Play with each of them to get ideas for projects of your own!

Teacher Links: Worksheet | Rubric | Teacher Prep Guide | Reflection Journal

Lesson 22: The Design Process

Project

- **Day 2 - The Design Process (45 min)**

Projects this big take time and plenty planning. Here, you will learn about the design process that you'll use to build your own creation.

Lesson 23: Build Your Project

Project

- **Day 3 - Build Your Project (45 min)**

Get those hands ready for plenty of coding! It's time to start building your project.

Lesson 24: Present Your Project

Project

- **Day 4 & 5 - Present Your Project (45 min each)**

Get ready to show off! It's time to present your finished project to your peers.

Lesson 25: Beyond Programming: The Internet

Unplugged | Internet

- **Warm Up (20 min)**
- **Main Activity (20 min)**
- **Wrap Up (15 min)**
- **Assessment (5 min)**

Ever wondered how information travels across the internet? It's not magic! This lesson will teach you the basics of how the internet works.

Teacher Links: Teacher Video | Manipulatives | Assessment | Reflection Journal

Lesson 26: Beyond Programming: Crowdsourcing

Unplugged | Crowdsourcing

- **Warm Up (20 min)**
- **Main Activity (20 min)**
- **Wrap Up (15 min)**
- **Extended Learning**

This lesson will teach you about crowdsourcing, the process of building a project with a team.

Teacher Links: Teacher Video | Worksheet | Reflection Journal

GRADE 5 - COURSE F

https://curriculum.code.org/csf/coursef/

Course F

The last course in CS Fundamentals was tailored to the needs students in the fifth grade.

In these lessons, students will create programs with different kinds of loops, events, functions, and conditionals. They will also investigate different problem-solving techniques and discuss societal impacts of computing and the internet. By the end of the curriculum, students create interactive stories and games that they can share with their friends and family.

Lesson 1: Programming: My Robotic Friends

Algorithms | Debugging | Unplugged

- **Warm Up (5 min)**
- **Main Activity (45 min)**
- **Wrap Up (10 min)**

Teacher Links: Teacher Prep Guide | Video | Symbol Key | Manipulatives | Reflection Journal

Lesson 2: Sequences in Maze

Sequencing | Debugging | Loop

- **Warm Up (10 min)**
- **Bridging Activities - Programming (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (10 min)**

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 3: Building a Foundation

Unplugged | Persistence | Frustration

- **Warm Up (20 min)**
- **Main Activity (20 min)**
- **Wrap Up (10 min)**
- **Extended Learning**

Teacher Links: Teacher Video | Lesson in Action Video | Teacher Prep Guide | Reflection Journal

Lesson 4: Debugging with Scrat

Bug | Debugging | Scrat | Ice Age

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (5 - 10 min)**
- **Extended Learning**

Teacher Links: Lesson Recommendations | Reflection Journal Student Links: Student Handout

Lesson 5: Programming in Artist

Artist | Programming

- **Warm Up (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (10 - 15 min)**
- **Extended Learning**

Teacher Links: Website | Lesson Recommendations | Reflection Journal Student Links: Student Video | Student Video | Student Handout

Lesson 6: My Loopy Robotic Friends

Unplugged | Loop | Repeat

- **Warm Up (10 - 15 min)**
- **Main Activity (15 - 20 min)**
- **Wrap Up (10 min)**
- **Extension Activities**

Teacher Links: Video | Teacher Prep Guide | Symbol Key | Reflection Journal

Lesson 7: Loops in Artist

Loop | Artist

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 8: Nested Loops

Nested Loops | Loops | Bee | Maze

- **Warm Up (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 9: Nested Loops with Frozen

Loop | Nested Loop | Artist

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 10: Algorithms: Tangrams

Unplugged | Algorithms

- **Warm Up (10 min)**
- **Main Activity (20 min)**
- **Wrap Up (15 min)**
- **Assessment (10 min)**

Here you will learn about algorithms using puzzles called tangrams!

Teacher Links: Teacher Video | Manipulatives | Assessment | Reflection Journal

Lesson 11: Introduction to Online Puzzles

Algorithms | Loop | Repeat

- **Warm Up (5 min)**
- **Bridging Activity (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

This introduction is very fast in pace. If you feel that your class could benefit from a more in-depth introduction to computer science, please begin with the ramp-up activities of lessons 1-9.

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 12: Digital Citizenship

Common Sense Education | Cyberbullying

- **Warm Up (5 min)**

- **Main Activity (35 min)**
- **Wrap Up (15 min)**
- **Assessment (10 min)**

Bullying is never okay. This lesson will teach you about what is and isn't okay to say online.

Teacher Links: Teacher Prep Guide | Lesson Video | Website | Reflection Journal | Resource List

Lesson 13: Events in Ice Age

Event | Ice Age

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Think of your favorite video game. Ever wondered how it was made? In these puzzles you will develop a video game of your own with friends from Ice Age!

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 14: Conditionals in Minecraft

Conditional | Minecraft

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**
- **Extended Learning**

Avoid the lava! Here you will learn about conditionals in the world of Minecraft.

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 15: Variables: Envelope Variables

Unplugged | Variable

- **Warm Up (10 min)**
- **Main Activity (20 min)**
- **Wrap Up (10 min)**
- **Assessment (10 min)**
- **Extended Learning**

Envelopes and variables have something in common: both can hold valuable things. Here you will learn what variables are and the awesome things they can do.

Teacher Links: Teacher Video | Worksheet | Assessment | Reflection Journal

Lesson 16: Variables in Artist

Variable | Artist

- **Warm Up (15 min)**
- **Bridging Activity - Variables (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Don't forget to bring creativity to class! In these puzzles you will be making fantastic drawings using variables.

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal Student Links: Student Video

Lesson 17: Variables in Play Lab

Variable | Play Lab

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Soon you will learn about making characters interact in a game using variables!

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 18: For Loops: For Loop Fun

Unplugged | For Loops

- **Warm Up (20 min)**
- **Main Activity (20 min)**
- **Wrap Up (15 min)**
- **Assessment (5 min)**
- **Extended Learning**

You're going to have loads of fun learning about for loops!

Teacher Links: Teacher Video | Lesson in Action Video | Worksheet | Assessment | Reflection Journal

Lesson 19: For Loops in Bee

For Loop | Bee

- **Warm Up (15 min)**
- **Bridging Activity - For Loops (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Buzz buzz. In these puzzles you will be guiding a bee to nectar and honey using for loops!

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 20: For Loops in Artist

For Loop | Artist

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Get ready to make your next masterpiece. Here you will be using for loops to make some jaw-dropping pictures.

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 21: Functions: Songwriting with Parameters

Unplugged | Function | Parameter

- **Warm Up (15 min)**
- **Main Activity (20 min)**
- **Wrap Up (15 min)**
- **Assessment (5 min)**
- **Extended Learning**

You just might release the next big hit single! In this lesson, you will be learning what parameters are and how they make some fantastic songs!

Teacher Links: Teacher Video | Lesson in Action Video | Worksheet | Assessment | Reflection Journal

Lesson 22: Functions in Bee

Function | Bee

- **Warm Up (15 min)**
- **Bridging Activity - Functions (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**
- **Extended Learning**

The bee needs your help again! Here you will be using functions to get nectar and make honey!

Teacher Links: Website | Manipulatives | Lesson Recommendations | Reflection Journal

Lesson 23: Functions with Parameters in Artist

Function | Parameter | Artist

- **Warm Up (10 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

Get your programming fingers ready. In these puzzles you will make impressive drawings in Artist using functions with parameters.

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 24: Functions with Parameters in Bee

Function | Parameter | Bee

- **Warm Up (15 min)**
- **Main Activity (30 min)**
- **Wrap Up (15 min)**

You've had a little practice using functions with parameters. This lesson will continue your practice with Bee!

Teacher Links: Website | Lesson Recommendations | Reflection Journal

Lesson 25: Explore Project Ideas

Project | Define | Prepare | Try | Revise | Reflect

- **Day 1 - Explore Project Ideas (45 min)**
- **Day 2 - The Design Process (45 min)**
- **Day 3 - Build Your Project (45 min)**
- **Day 4 (Recommended for 5th Grade) - Revise Your Project (45 min)**
- **Day 5 & 6 - Present Your Project (45 min each)**
- **Extension Activity**

Time to get some inspiration! These puzzles will show you a handful of pre-built games and illustrations to help develop your plan for your BIG project.

Teacher Links: | Teacher Prep Guide | Worksheet | Rubric | Website |

Lesson 26: The Design Process

Project

- **Day 2 - The Design Process (45 min)**

Projects this big take time and plenty planning. Here, you will learn about the design process that you'll use to build your own creation.

Lesson 27: Build Your Project

Project

- **Day 3 - Build Your Project (45 min)**

Finally you can start building your project!

Lesson 28: Revise Your Project

Project

- **Day 4 - Revise Your Project (45 min)**

Rome wasn't built in a day and your project shouldn't be, either. Take time to edit and revise your project to make it the best it can be.

Lesson 29: Present Your Project

Project

- **Day 5 & 6 - Present Your Project (45 min each)**

Time to show your work! Here you will be presenting your awesome project to your peers.

GRADE 6 - 10

https://studio.code.org/courses/csd

Students work through the levels at their own pace.

Computer Science Discoveries

WHAT IS CS DISCOVERIES?

Computer Science Discoveries (CS Discoveries) is an introductory computer science course that empowers students to create authentic artifacts and engage with computer science as a medium for creativity, communication, problem solving, and fun.

Designed with equity in mind

CS Discoveries is designed from the ground up to be an accessible and engaging course for all students, regardless of background or prior experience. It provides students opportunities to engage with culturally and personally relevant topics in a wide variety of contexts and aims to show all students that CS is for them.

Focus on creation

We know that giving students agency in their learning is a powerful tool for creating fun, engaging, and lasting learning experiences. CS Discoveries focuses on the skills that enable students to create and express themselves in a variety of contexts and media. Whether they are developing their own website, designing an app, building a game, or creating a physical computing device, students are empowered to bring their ideas to life.

Completing a K-12 pathway

CS Discoveries is designed to fit naturally between our CS Fundamentals courses and our CS Principles course. While each of these courses is designed to be an age-appropriate entry point to computer science, students with previous experience will find many new topics to explore, and they will revisit familiar topics in novel and more challenging contexts.

Supports for new-to-CS teachers

CS Discoveries is specifically designed to support new-to-CS teachers. The curriculum includes detailed lesson plans, and frequent teaching tips. The accompanying forum is an active community of support that teachers can use to discuss their practice and find additional resources. Teachers can also apply for our professional learning program for further support.

Flexible Implementation

CS Discoveries was specifically designed for 7th to 9th grade classrooms and can be used in either middle school or high school. The two semesters build on each other, allowing the course to be taught as a single semester, two sequential semesters, a full-year course, or even integrated into existing technology classes.

Powerful tools with a high ceiling

CS Discoveries introduces students to tools and programming languages that are accessible for beginners while offering more advanced students opportunities to create sophisticated projects. Using our existing App Lab programming environment and our new programming environment, Game Lab, students will be able to transition from blocks to typed code at their own pace while learning JavaScript. Students will also develop maker skills through the study of physical computing at a very accessible cost.

PROBLEM SOLVING

This unit is a highly interactive and collaborative introduction to the field of computer science, as framed within the broader pursuit of solving problems. Through a series of puzzles, challenges, and real world scenarios, students are introduced to a problem solving process that they will return to repeatedly throughout the course. They then explore how computers represent and process information in order to solve certain kinds of problems. The unit concludes with students designing an application that could help address a problem of their choosing.

WEB DEVELOPMENT

In this unit, students are empowered to create and share the content on their own web pages. They begin by thinking about the role of the web, and how it can be used as a medium for creative expression. As students develop their pages and begin to see themselves as programmers, they are encouraged think critically about the impact of sharing information online and how to be more critical content consumers. They are also introduced to problem solving as it relates to programming, as they learn valuable skills such as debugging, commenting, and structure of language. At the conclusion of the unit, students compile their work to create a personal website they can publish and share.

INTERACTIVE GAMES AND ANIMATIONS

In this unit, students build on their coding experience as they create programmatic images, animations, interactive art, and games. Starting off with simple, primitive shapes and building up to more sophisticated sprite-based games, students become familiar with the programming concepts and the design process computer scientists use daily. They then learn how these simpler constructs can be combined to create more complex programs. In the final project, students develop a personalized, interactive program. Along the way, they practice design, testing, and iteration, as they come to see that failure and debugging are an expected and valuable part of the programming process.

THE DESIGN PROCESS

This unit transitions students from thinking about computer science as a tool to solve their own problems towards considering the broader social impacts of computing. Through a series of design challenges, students are asked to consider and understand the needs of others while developing a solution to a problem. The second half of the unit consists of an iterative team project, during which students have the opportunity to identify a need that they care about, prototype solutions both on paper and in App Lab, and test their solutions with real users to get feedback and drive further iteration.

DATA AND SOCIETY

This unit is about the importance of data in solving problems and highlights how computers can help in this process. The first chapter explores different systems used to represent information in a computer and the challenges and tradeoffs posed by using them. In the second chapter students learn how collections of data are used to solve problems, and how computers help to automate the steps of this process. The chapter concludes by considering how the data problem solving process can be applied to an area of the students' choosing.

PHYSICAL COMPUTING

In Unit 6, students further develop their programming skills while exploring the role of hardware platforms in computing. Harkening back to the Input and Output elements of the Input/Storage/Processing/Output model for a computing, students look towards current and “smart” devices to understand the ways in which different sensors can provide more effective input and output than the traditional keyboard, mouse, and monitor. Using App Lab and Adafruit’s Circuit Playground, students develop programs that utilize the same hardware inputs and outputs that students saw in the smart devices they explored earlier, and they get to see how a simple rough prototype can lead to a finished product. The unit concludes with a design challenge that asks students to use the Circuit Playground as the basis for an innovation of their own design.

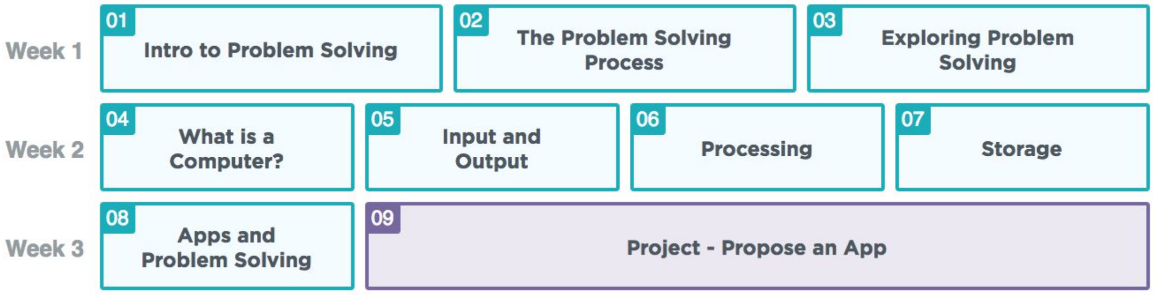
Unit	Lesson Plans	Code Studio
Unit 1 - Problem Solving	Lesson plans	Code Studio
Unit 2 - Web Development	Lesson plans	Code Studio
Unit 3 - Animations and Games	Lesson plans	Code Studio
Unit 4 - The Design Process	Lesson plans	Code Studio
Unit 5 - Data and Society	Lesson plans	Code Studio
Unit 6 - Physical Computing	Lesson plans	Code Studio

CSD Unit 1 - Problem Solving

https://curriculum.code.org/csd/unit1/

Unit 1 is a highly interactive and collaborative introduction to the field of computer science, as framed within the broader pursuit of solving problems. You'll practice using a problem solving process to address a series of puzzles, challenges, and real world scenarios. Next, you'll learn how computers input, output, store, and process information to help humans solve problems. The unit concludes with a project in which you design an application that helps solve a problem of your choosing.

https://studio.code.org/users/sign_in



Week 1	01 Intro to Problem Solving	02 The Problem Solving Process	03 Exploring Problem Solving	
Week 2	04 What is a Computer?	05 Input and Output	06 Processing	07 Storage
Week 3	08 Apps and Problem Solving	09 Project - Propose an App		

Chapter 1: The Problem Solving Process

Big Questions

- What strategies and processes can I use to become a more effective problem solver?

Week 1

Lesson 1: Intro to Problem Solving

Unplugged

- Tech Setup
- CSD Pre-Course Survey
- Warm Up (10 min)
- Activity (30 min)
- Wrap Up (10 mins)

The class works in groups to design aluminum foil boats that will support as many pennies as possible. At the end of the lesson groups reflect on their experiences with the activity and make connections to the types of problem solving they will be doing for the rest of the course.

Teacher Links: Exemplar | Student Links: Activity Guide

Lesson 2: The Problem Solving Process

Unplugged

- Warm Up (5 min)
- Activity (30 min)
- Wrap Up (15 min)
- Extended Learning

This lesson introduces the formal problem solving process that the class will use over the course of the year, Define - Prepare - Try - Reflect. The class relates these steps to the aluminum boats problem from the previous lesson, then a problem they are good at solving, then a problem they want to improve at solving. At the end of the lesson the class collects a

list of generally useful strategies for each step of the process to put on posters that will be used throughout the unit and year.
Teacher Links: Graphic | Exemplar Student Links: Activity Guide

Lesson 3: Exploring Problem Solving
Unplugged

- Warm Up (5 min)
- Activity (75 min)
- Wrap Up (20 min)

In this lesson the class applies the problem solving process to three different problems: a word search, a seating arrangement for a birthday party, and planning a trip. The problems grow increasingly complex and poorly defined to highlight how the problem solving process is particularly helpful when tackling these types of problems.
Teacher Links: Key | Website Student Links: Activity Guide
Chapter Commentary
This chapter guides students to develop and adopt a more formal structured problem solving process by reflecting on problems they have problems they have encountered, both in the classroom and everyday life. By working through a diverse set of problems, such as logic puzzles, engineering challenges, and planning a trip, students learn to identify different classes of problems, decompose large problems, and develop their personal problem solving skills.

Chapter 2: Computers and Problem Solving
Big Questions

- How do computers help people to solve problems?
- How do people and computers approach problems differently?
- What does a computer need from people in order to solve problems effectively?

Week 2
Lesson 4: What is a Computer?
Unplugged

- Warm Up (5 mins)
- Activity (40 mins)
- Wrap Up (5 mins)

In this lesson the class develops a preliminary definition of a computer. After brainstorming the possible definitions for a computer, the class works in groups to sort pictures into “is a computer” or “is not a computer” on poster paper and explain their motivations for choosing some of the most difficult categorizations. The teacher then introduces a definition of the computer and allows groups to revise their posters according to the new definition.
Teacher Links: Video | Graphic | Exemplar Student Links: Activity Guide

Lesson 5: Input and Output

- Warm Up (5 mins)
- Activity
- Wrap Up (15 mins)

In this the class students consider a number of computing devices to determine what types of inputs and outputs they use. Groups are assigned to a computing device and based on a teacher-provided definition of input and output, list the inputs and outputs of their device. To

conclude the lesson the class examines common activities they do on a computing device and select the inputs and outputs used for that activity from the chart.
Teacher Links: Exemplar Student Links: Activity Guide

Lesson 6: Processing
Unplugged

- Warm Up (10 min)
- Activity (30 mins)
- Wrap Up (10 min)

This lesson dives deeper into the concept of processing that was introduced as part of the definition of a computer. Pairs work together to puta deck of cards in order, a form of processing information. In the end, the class discusses what processing means within the context of solving information problems.
Teacher Links: | Exemplar Student Links: Activity Guide

Lesson 7: Storage

- Warm Up
- Activity
- Wrap Up (15 mins)

This lesson focuses on the storage component of the definition of a computer, within the content of processing information. The class spends the majority of the lesson developing and sharing algorithms to process information, with an emphasis on how much storage is needed for any particular algorithm. The lesson concludes with a discussion of the importance of storage while processing information.
Teacher Links: Exemplar Student Links: Activity Guide
Week 3

Lesson 8: Apps and Problem Solving
Unplugged

- Warm Up (5 min)
- Activity (40 min)
- Wrap Up (10 min)
- Extension Activities

This lesson covers the input and output aspects of computers in a context that is relevant and familiar to students: apps. The class evaluates various web applications to analyze the specific problems that they were designed to solve, the inputs that they need to work, and the outputs they provide to users. The class concludes with observations of these apps as well as a teacher led discussion about the impact of apps on society.
Teacher Links: Key | Presentation Student Links: Resource | Activity Guide

Lesson 9: Project - Propose an App
Unplugged | Project

- Warm Up (10 min)
- Activity (150 min)
- Wrap Up
- Extended Learning

To conclude the study of the problem solving process and the input/output/store/process model of a computer, the class proposes apps designed to solve real world problems. This project is completed across multiple days and culminates in a poster presentation highlighting the features of each app. The project is designed to be completed in pairs though it can be completed individually.

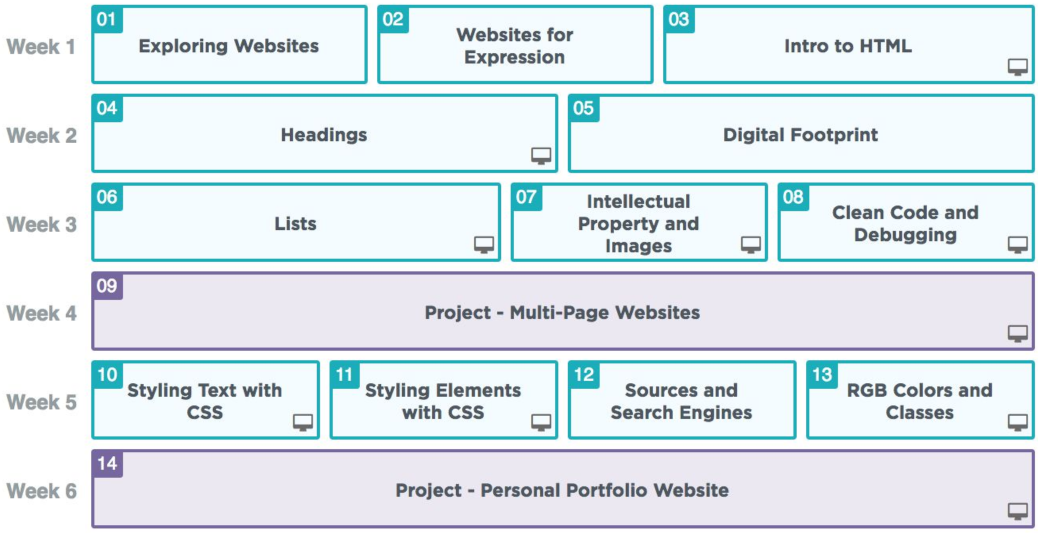
Teacher Links: Project Guide Exemplar | Peer Review Exemplar Student Links: Project Guide | Peer Review | Rubric

Chapter Commentary

In the second half of the unit, students move on to thinking about computers as machines that solve information problems. Students begin by building a common definition for a computer that focuses on functionality instead of specific hardware. They then explore the ways that computers approach problems. For their final project, students propose an app that could be used to solve a problem of their choosing.

CSD Unit 2 - Web Development

https://curriculum.code.org/csd/unit2/



Week 1	01 Exploring Websites	02 Websites for Expression	03 Intro to HTML 🖥	
Week 2	04 Headings 🖥	05 Digital Footprint		
Week 3	06 Lists 🖥	07 Intellectual Property and Images 🖥	08 Clean Code and Debugging 🖥	
Week 4	09 Project - Multi-Page Websites 🖥			
Week 5	10 Styling Text with CSS 🖥	11 Styling Elements with CSS 🖥	12 Sources and Search Engines	13 RGB Colors and Classes 🖥
Week 6	14 Project - Personal Portfolio Website 🖥			

In Unit 2, you'll learn how to create and share the content on your own web pages. After deciding what content you want to share with the world, you'll learn how to structure and style your pages using HTML and CSS. You'll also practice valuable programming skills such as debugging and commenting. By the end of the unit, you'll have a personal website that you can publish to the Internet.
https://studio.code.org/users/sign_in

Chapter 1: Web Content and HTML

Big Questions

- Why do people create websites?
- How can text communicate content and structure on a web page?
- How can I incorporate content I find online into my own webpage?
- What strategies can I use when coding to find and fix issues?

Week 1

Lesson 1: Exploring Websites

Unplugged

- Warm Up (10 minutes)
- Activity (45 minutes)
- Wrap Up (5 min)

This lesson covers the purposes that a website might serve, both for the users and the creators. The class explores a handful of the most-used websites in the United States and discusses how each of those sites is useful for users and how it might also serve its creators. Teacher Links: Exemplar Student Links: Activity Guide

Lesson 2: Websites for Expression

Unplugged

- Warm Up
- Activity
- Wrap Up (5 mins)

This lesson introduces websites as a means of personal expression. The class first discusses different ways that people express and share their interests and ideas, then looks at a few exemplar websites made by students from a previous course. Finally everyone brainstorms and shares a list of topics and interests to include, creating a resource for developing a personal website in the rest of the unit.
Teacher Links: Exemplar Student Links: Activity Guide

Lesson 3: Intro to HTML
Web Lab

- Warm Up (10 mins)
- Activity (30 minutes)
- Wrap Up (5 minutes)

This lesson introduces to HTML as a solution to the problem of how to communicate both the content and structure of a website to a computer. The lesson begins with a brief unplugged activity demonstrating the challenges of effectively communicating the structure of a web page. The class looks at an HTML page in Web Lab and discusses how HTML tags help solve this problem, then uses HTML to write the first web pages of the unit.
Teacher Links:

Week 2
Lesson 4: Headings
Web Lab

- Warm Up (5 minutes)
- Activity (45 minutes)
- Wrap Up (5 minutes)

This lesson continues the introduction to HTML tags, this time with headers. The class practices using header tags to create page and section titles and learns how the different header elements are displayed by default. Next, the class plans how to organize their content on the personal web pages that will be built across the unit and begins the first page of the project.

Lesson 5: Digital Footprint
Unplugged

- Warm Up (10 min)
- Activity (30-40 min)
- Wrap Up (5 minutes)

This lesson takes a step back from creating the personal website to talk about personal information people choose to share digitally. The class begins by discussing what types of information are good to share with other people, then looks at several sample social media pages to see what types of personal information could be shared intentionally or unintentionally. Finally, the class comes up with a set of guidelines to follow when putting information online.
Teacher Links: Exemplar | Exemplar Student Links: Activity Guide | Activity Guide

Week 3
Lesson 6: Lists
Web Lab

- Warm Up (10 minutes)
- Activity (40 minutes)

- Wrap Up (10 minutes)

This lesson introduces ordered and unordered lists and the associated <ul>, <ol>, and <li> HTML tags. The class practices using the tags, then goes back to the personal web page project to add a new HTML page that includes the new tags.

Lesson 7: Intellectual Property and Images

Unplugged | Web Lab

- Warm Up (10 min)
- Activity 1 (20 min)
- Activity 2 (20 min)
- Wrap Up (5 min)

This lesson covers how to use media such as images, video, or music created by others a website while respecting the rights of the creator of that media. After first studying Creative Commons licensing, the class learns how to add images to web pages, and how to give proper attribution when doing so.

Teacher Links: Exemplar Student Links: Video | Activity Guide

Lesson 8: Clean Code and Debugging

Web Lab

- Warm Up (10 minutes)
- Activity (35 minutes)
- Wrap Up (10 minutes)

In this lesson covers common issues that arise when designing web pages in HTML. The class will correct errors in a sequence of increasingly complex web pages found on Code Studio and learn the importance of comments, whitespace, and indentation as tools for making web pages easier to read.

Week 4

Lesson 9: Project - Multi-Page Websites

Web Lab | Project

- Warm Up (15 minutes)
- Activity
- Wrap Up (10 minutes)

This lesson covers hyperlinks, which allow web developers to connect pages together into one website. The class will link together all the previous pages into one project, and create navigation bars for each page before publishing the entire site to the Web.

Teacher Links: Exemplar Student Links: Project Guide | Peer Review | Rubric

Chapter Commentary

Students use computing as a form of self expression as they design and develop basic web pages. Focusing on the tags, keywords, and syntax used to communicate instructions to the computer, students use HTML to structure the content of a web page. They also explore the privacy and intellectual property implications of publishing their work online.

Chapter 2: Styling and CSS

Big Questions

- How do I modify the appearance and style of my web pages?
- How do I safely and appropriately make use of the content published on the Internet?

Week 5

Lesson 10: Styling Text with CSS

Web Lab

- Warm Up (5 minutes)
- Activity (40 minutes)
- Wrap Up (10 minutes)

This lesson introduces CSS as a way to style elements on the page. The class learns the basic syntax for CSS rule-sets and then explores properties that impact HTML text elements. Finally, everyone applies text styles to their personal websites.

Lesson 11: Styling Elements with CSS

Web Lab

- Warm Up (5 minutes)
- Activity (40 minutes)
- Wrap Up (5 minutes)

This lesson continues the introduction to CSS style properties, this time focusing more on non-text elements. The class begins by investigating and modifying the new CSS styles on a Desserts of the World page. Afterwards, everyone applies this new knowledge to their personal websites.

Lesson 12: Sources and Search Engines

- Warm Up (10 minutes)
- Activity (30 minutes)
- Wrap Up (5 minutes)

After first completing a web search scavenger hunt, the class learns about the inner workings of search engines and has an opportunity to flex their analytical skills in a search for strange and unlikely animals.

Teacher Links: Key Student Links: Video | Activity Guide | Activity Guide

Lesson 13: RGB Colors and Classes

Web Lab

- Warm Up (5 minutes)
- Activity (40 minutes)
- Wrap Up (5 minutes)

This lesson covers classes and custom colors. The class first learns how to specify custom colors using RGB (red, green, blue) values, then applies these colors to a new Four Seasons web page, which uses CSS classes. Using classes, the class adds more styles to the Four Seasons web page, then use them to style their personal websites.

Teacher Links: Key Student Links: Activity Guide

Week 6

Lesson 14: Project - Personal Portfolio Website

Web Lab | Project

• Day 1 • Day 2 • Day 3	In the last few days of the unit, the class finalizes their personal websites, working with peers to get feedback, review the rubric, and put the finishing touches on the site. To cap off the unit, everyone shares their projects and how they were developed.

Teacher Links: Project Guide Exemplar | Peer Review Exemplar Student Links: Peer Review | Project Guide | Rubric

Chapter Commentary

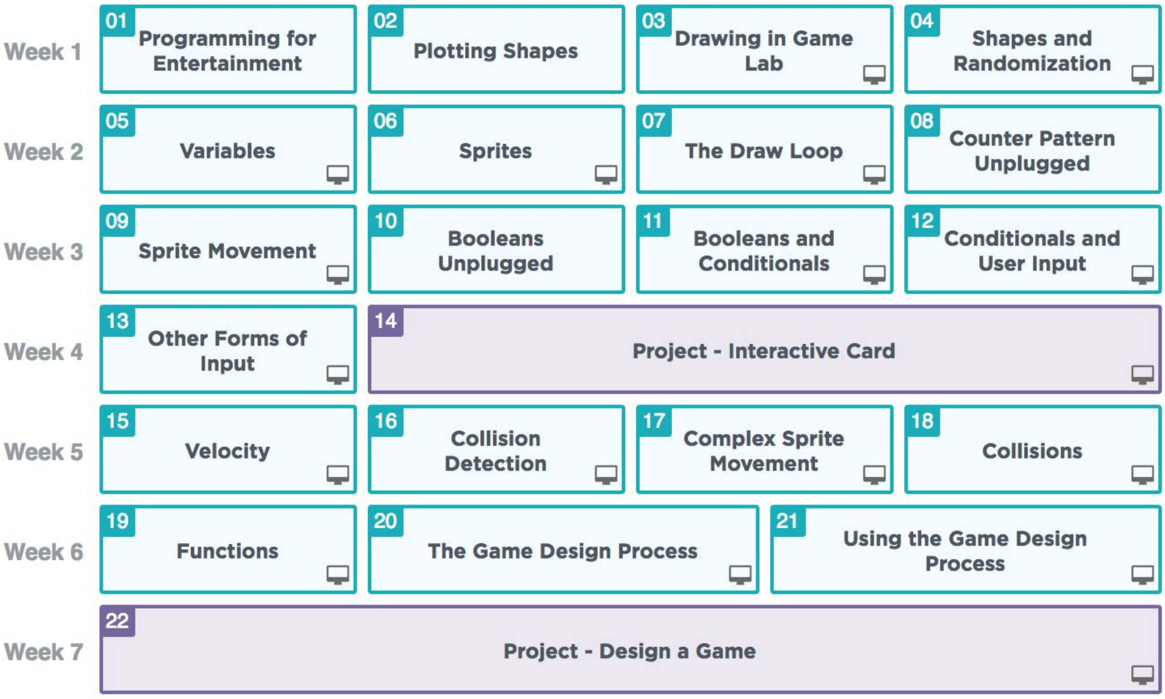
After covering the basics of HTML, students dive into improving their websites with CSS. They learn to use colors, fonts and margins to create a unique style of their own design. At the end of the chapter students publish a personal portfolio website that demonstrates use of HTML for content and CSS for a personalized.

CSD Unit 3 - Animations and Games

https://curriculum.code.org/csd/unit3/

In Unit 3, you'll build on your coding experience as you program animations, interactive art, and games in Game Lab. The unit starts off with simple shapes and builds up to more sophisticated sprite-based games, using the same programming concepts and the design process computer scientists use daily. In the final project, you'll develop a personalized, interactive program.

https://studio.code.org/users/sign_in



Week 1	01 Programming for Entertainment	02 Plotting Shapes	03 Drawing in Game Lab	04 Shapes and Randomization
Week 2	05 Variables	06 Sprites	07 The Draw Loop	08 Counter Pattern Unplugged
Week 3	09 Sprite Movement	10 Booleans Unplugged	11 Booleans and Conditionals	12 Conditionals and User Input
Week 4	13 Other Forms of Input	14 Project - Interactive Card		
Week 5	15 Velocity	16 Collision Detection	17 Complex Sprite Movement	18 Collisions
Week 6	19 Functions	20 The Game Design Process	21 Using the Game Design Process	
Week 7	22 Project - Design a Game			

Chapter 1: Images and Animations

Big Questions

- What is a computer program?
- What are the core features of most programming languages?
- How does programming enable creativity and individual expression?
- What practices and strategies will help me as I write programs?

Week 1

[Lesson 1: Programming for Entertainment]()

Unplugged

- Warm Up (10 min)
- Activity (45 min)
- Wrap Up (5 min)

The class is asked to consider the "problems" of boredom and self expression, and to reflect on how they approach those problems in their own lives. From there, they will explore how

Computer Science in general, and programming specifically, plays a role in either a specific form of entertainment or as a vehicle for self expression.
Teacher Links: Exemplar Student Links: Activity Guide

Lesson 2: Plotting Shapes
Unplugged

- Warm Up (10 min)
- Activity (35 min)
- Wrap Up (5 min)

This lesson explores the challenges of communicating how to draw with shapes and use a tool that introduces how this problem is approached in Game Lab.The class uses a Game Lab tool to interactively place shapes on Game Lab's 400 by 400 grid. Partners then take turns instructing each other how to draw a hidden image using this tool, accounting for many of the challenges of programming in Game Lab.
Teacher Links: | Exemplar Student Links: Activity Guide

Lesson 3: Drawing in Game Lab
Game Lab

- Warm Up (5 minutes)
- Activity (30 minutes)
- Wrap Up (10 minutes)

The class is introduced to Game Lab, the programming environment for this unit, and begins to use it to position shapes on the screen. The lesson covers the basics of sequencing and debugging, as well as a few simple commands. At the end of the lesson, the class creates an online version of the image they designed in the previous lesson.

Lesson 4: Shapes and Randomization
Game Lab

- Warm Up (5 min)
- Activity (40 min)
- Wrap Up (5 min)

This lesson extends the drawing skills to include width and height and introduces the concept of random number generation. The class learns to draw with versions of ellipse() and rect() that include width and height parameters and to use the background() block to fill the screen with color. At the end of the progression the class is introduced to the randomNumber() block and uses the new blocks to draw a randomized rainbow snake.
Week 2

Lesson 5: Variables
Game Lab

- Warm Up (10 Mins)
- Activity (30 Mins)
- Wrap Up (5 Mins)

This lesson introduces variables as a way to label a number in a program or save a randomly generated value. The class begins the lesson with a very basic description of the purpose of a variable and practices using the new blocks. Afterwards, the class uses variables to save a random number, allowing the programs to use the same random number multiple times.
Student Links: Video

Lesson 6: Sprites

Game Lab

- Warm Up (5 minutes)
- Activity
- Wrap Up (5-10 min)
- Assessment

In order to create more interesting and detailed images, the class is introduced to the sprite object. Every sprite can be assigned an image to show, and sprites also keep track of multiple values about themselves, which will prove useful down the road when making animations. At the end of the lesson, everyone creates a scene using sprites.

Teacher Links: Exemplar Student Links: Activity Guide

Lesson 7: The Draw Loop

Game Lab

- Warm Up (5 mins)
- Activity (60 min)
- Wrap Up (10 mins)

This lesson introduces the draw loop, one of the core programming paradigms in Game Lab. The class combines the draw loop with random numbers to manipulate some simple animations with dots and then with sprites. Afterwards, everyone uses what they learned to update the sprite scene from the previous lesson.

Teacher Links: Manipulative | Manipulative | Video

Lesson 8: Counter Pattern Unplugged

Unplugged

- Warm Up (15 mins)
- Activity (20 mins)
- Activity 2 (30 mins)
- Wrap Up (10 mins)

This unplugged lesson explores the underlying behavior of variables. Using notecards and string to simulate variables within a program, the class implements a few short programs. Once comfortable with this syntax, the class uses the same process with sprite properties, tracking a sprite's progress across the screen.

Teacher Links: Key Student Links: Activity Guide | Manipulative

Week 3

Lesson 9: Sprite Movement

Game Lab

- Warm Up (5 minutes)
- Activity (40 minutes)
- Wrap Up (5 minutes)

By combining the Draw Loop and the Counter Pattern, the class writes programs that move sprites across the screen, as well as animate other sprite properties.

Lesson 10: Booleans Unplugged

Unplugged

- Warm Up (10 min)
- Activity (30 min)

- Wrap Up (5 min)

This lesson introduces boolean values and logic, as well as conditional statements. The class starts by playing a simple game of Stand Up, Sit Down in which the boolean (true/false) statements describe personal properties (hair or eye color, clothing type, age, etc). The class then groups objects based on increasingly complex boolean statements, then looks at how conditionals can impact the flow of a program.
Student Links: Activity Guide

Lesson 11: Booleans and Conditionals
Game Lab

- Warm Up (5 min)
- Activity (40 min)
- Wrap Up (5 min)

The class starts by using booleans to compare the current value of a sprite property with a target value, using that comparison to determine when a sprite has reached a point on the screen, grown to a given size, or otherwise reached a value using the counter pattern. After using booleans directly to investigate the values or sprite properties, the class adds conditional if statements to write code that responds to those boolean comparisons.

Lesson 12: Conditionals and User Input
Game Lab

- Warm Up (5 min)
- Activity (40 min)
- Wrap Up (5 min)

Following the introduction to booleans and if statements in the previous lesson, students are introduced to a new block called keyDown() which returns a boolean and can be used in conditionals statements to move sprites around the screen. By the end of this lesson students will have written programs that take keyboard input from the user to control sprites on the screen.
Student Links: Video

Week 4
Lesson 13: Other Forms of Input
Game Lab

- Warm Up (5 minutes)
- Activity (40 minutes)
- Wrap Up (5 minutes)

The class continues to explore ways to use conditional statements to take user input. In addition to the simple keyDown() command learned yesterday, the class learns about several other keyboard input commands as well as ways to take mouse input.

Lesson 14: Project - Interactive Card
Game Lab | Project

- Warm Up (10 min)
- Activity (2 days)
- Wrap Up (10 minutes)
- Assessment

In this cumulative project for Chapter 1, the class plans for and develops an interactive greeting card using all of the programming techniques they've learned to this point.

Teacher Links: Peer Review Exemplar | Project Guide Exemplar Student Links: Project Guide | Rubric | Peer Review

Chapter Commentary

Students build up toward programming interactive animations in the Game Lab environment. They begin with simple shapes and sprite objects, then use loops to create flipbook style animations. Next, they learn to use booleans and conditionals to respond to user input. At the end of the chapter, students design and create an interactive animation that they can share with the world.

Chapter 2: Building Games

Big Questions

- How do software developers manage complexity and scale?
- How can programs be organized so that common problems only need to be solved once?
- How can I build on previous solutions to create even more complex behavior?

Week 5

Lesson 15: Velocity

Game Lab

- Warm Up (15 min)
- Activity (75 minutes)
- Wrap Up (5 min)

After a brief review of how the counter pattern is used to move sprites, the class is introduced to the properties that set velocity and rotation speed directly. As they use these new properties in different ways, they build up the skills they need to create a basic side scroller game.

Lesson 16: Collision Detection

Game Lab

- Warm Up (10 min)
- Activity
- Wrap Up (5 min)

The class learns about collision detection on the computer. Pairs explore how a computer could use sprite location and size properties and math to detect whether two sprites are touching. The class then uses the isTouching() block to create different effects when sprites collide, including playing sounds. Last, they use their new skills to improve the sidescroller game that they started in the last lesson.

Teacher Links: Exemplar Student Links: Activity Guide

Lesson 17: Complex Sprite Movement

Game Lab

- Warm Up (10 mins)
- Activity (60 mins)
- Wrap Up (10 mins)

The class learns to combine the velocity properties of sprites with the counter pattern to create more complex sprite movement, such as simulating gravity, making a sprite jump, and allowing a sprite to float left or right. In the final levels the class combine these movements to animate and control a single sprite and build a simple game in which a character flies around and collects coins.

Lesson 18: Collisions

Game Lab

- Warm Up
- Activity
- Wrap Up (10 mins)

The class programs their sprites to interact in new ways. After a brief review of how they used the isTouching block, the class brainstorms other ways that two sprites could interact. They then use isTouching to make one sprite push another across the screen before practicing with the four collision blocks (collide, displace, bounce, and bounceOff).

Week 6

Lesson 19: Functions

Game Lab

- Warm Up (10 mins)
- Activity (60 mins)
- Wrap Up (10 mins)

This lesson covers functions as a way to organize their code, make it more readable, and remove repeated blocks of code. The class learns that higher level or more abstract steps make it easier to understand and reason about steps, then begins to create functions in Game Lab. At the end of the lesson the class uses these skills to organize and add functionality to the final version of their side scroller game.

Lesson 20: The Game Design Process

Game Lab

- Warm Up (15 mins)
- Activity (60 mins)
- Wrap Up (20 mins)

This lesson introduces the process the class will use to design games for the remainder of the unit. The class walks through through this process in a series of levels. As part of this lesson the class also briefly learn to use multi-frame animations in Game Lab. At the end of the lesson they have an opportunity to make improvements to the game to make it their own.
Student Links: Project Guide

Lesson 21: Using the Game Design Process

Game Lab

- Warm Up
- Activity
- Wrap Up

In this multi-day lesson, the class uses the problem solving process from Unit 1 to create a platform jumper game. After looking at a sample game, the class defines what their games will look like and uses a structured process to build them. Finally, the class reflects on how the games could be improved, and implements those changes.
Teacher Links: Exemplar Student Links: Project Guide

Week 7

Lesson 22: Project - Design a Game

Game Lab | Project

- Warm Up (10 mins)
- Activity (80-200 mins)
- Wrap Up (10 mins)

The class plans and builds original games using the project guide from the previous two lessons. Working individually or in pairs, the class plans, develops, and gives feedback on the games. After incorporating the peer feedback, the class shares out the completed games.

Student Links: Project Guide | Rubric | Peer Review

Chapter Commentary

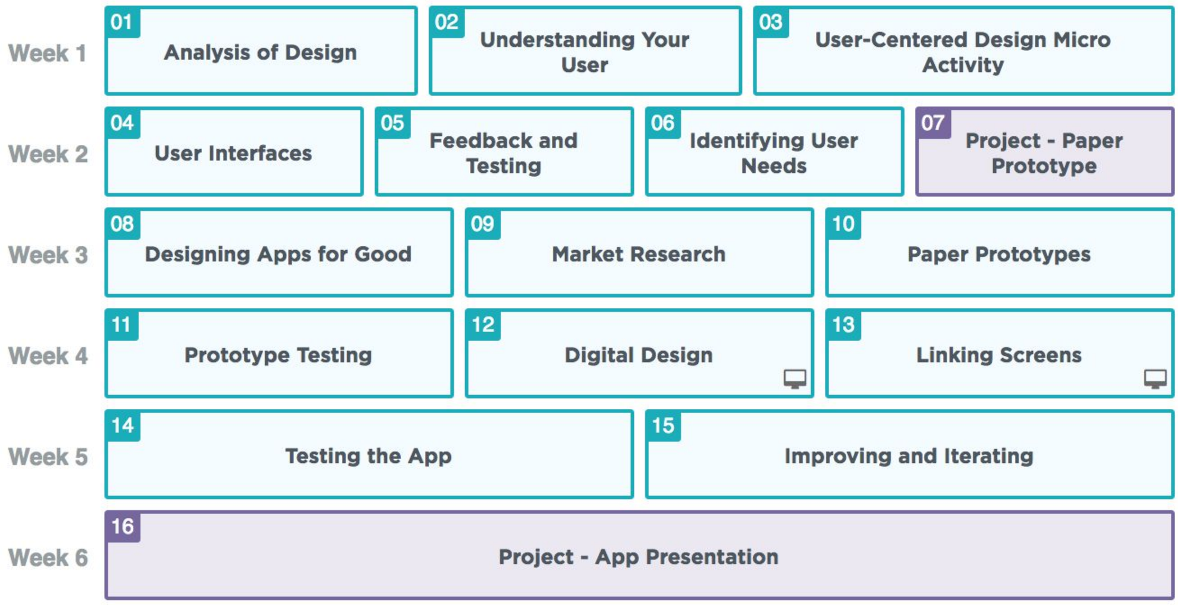
In this chapter students combine the constructs that they learned in the first chapter to program more complex movement and collisions in their sprites. As they create more complex programs, they begin to use functions to organize their code. In the end, students use a design process to create an original game.

CSD Unit 4 - The Design Process

https://curriculum.code.org/csd/unit4/

Unit 4 introduces the broader social impacts of computing. Through a series of design challenges, you will learn how to better understand the needs of others while developing a solution to a problem. The second half of the unit consists of an iterative team project, during which teams have the opportunity to identify a need that they care about, prototype solutions both on paper and in App Lab, and test solutions with real users to get feedback and drive further iteration.

https://studio.code.org/users/sign_in



Week 1	01 Analysis of Design	02 Understanding Your User	03 User-Centered Design Micro Activity	
Week 2	04 User Interfaces	05 Feedback and Testing	06 Identifying User Needs	07 Project - Paper Prototype
Week 3	08 Designing Apps for Good	09 Market Research	10 Paper Prototypes	
Week 4	11 Prototype Testing	12 Digital Design	13 Linking Screens	
Week 5	14 Testing the App	15 Improving and Iterating		
Week 6	16 Project - App Presentation			

Chapter 1: User Centered Design

Big Questions

- How do designers identify the needs of their user?
- How can we ensure that a user's needs are met by our designs?
- What processes will best allow us to efficiently create, test, and iterate upon our designs?

Week 1

Lesson 1: Analysis of Design

Unplugged

- Warm Up (5 min)
- Activity (35 min)
- Wrap Up (5 min)
- Extension Activites

The class explores a variety of different teapot designs to consider how design choices or made and why. Using the teapots as an example, the class will explore the relationship between users, their needs, and the design of objects they use.

Student Links: Activity Guide | Image

Lesson 2: Understanding Your User

Unplugged

- Warm Up (10 min)
- Activity (30 min)
- Wrap Up (5 min)

Using user profiles, the class explores how different users might react to a variety of products. Role playing as a different person, each member of the class will get to experience designs through someone else's eyes.

Teacher Links: Slide Deck Student Links: Activity Guide | User Profile | User Profile | User Profile | User Profile

Lesson 3: User-Centered Design Micro Activity

Unplugged

- Warm Up (5 min)
- Activity (50 min)
- Wrap Up (20 min)

In small groups, the class uses the design process to come up with ideas for smart clothing. From brainstorming, to identifying users, to finally proposing a design, this is the first of several opportunities in this unit to practicing designing a solution for the needs of others.

Student Links: Activity Guide

Week 2

Lesson 4: User Interfaces

Unplugged

- Warm Up (10 min)
- Activity (40 min)
- Wrap Up (5 min)

See how a paper prototype can be used to test and get feedback on software before writing any code. To help out a developer with their idea, the class tests and provides an app prototype made of paper.

Student Links: Activity Guide | Activity Guide | Activity Guide

Lesson 5: Feedback and Testing

Unplugged

- Warm Up (10 mins)
- Activity (40 mins)
- Wrap Up

Users have been testing an app, and they have lots of feedback for the developer. The class needs to sort through all of this feedback, identify the common themes and needs, and start revising the prototype to make it better meet the users' needs.

Student Links: Activity Guide | Activity Guide | Activity Guide

Lesson 6: Identifying User Needs

Unplugged

- Warm Up (5 min)
- Activity (40 min)
- Wrap Up

Up to this point the users that the class has considered have all been remote, and the only information from users has come through text or role playing. Now the class will rely on each other as potential users, and pairs will get to interview each other to identify needs that could be addressed by developing an app.
Student Links: Activity Guide | Project Guide

Lesson 7: Project - Paper Prototype
Unplugged | Project

- Warm Up (5 mins)
- Activity
- Wrap Up

Using the interview information from the previous lesson, the class comes up with app ideas to address the needs of their users. To express those ideas, and test out their effectiveness, each student creates and tests paper prototypes of their own.
Student Links: Rubric | Project Guide

Chapter Commentary

This chapter introduces the design process as a specific version of the problem solving process in which empathy for a user's needs is consistently integrated. Students learn strategies for identifying user needs and assessing how well different designs address them. In particular they learn how to develop a paper prototype, how to gather and respond to feedback about a prototype, and consider ways different user interfaces do or do not affect the usability of their apps.

Chapter 2: App Prototyping

Big Questions

- How do teams effectively work together to develop software?
- What roles beyond programming are necessary to design and develop software?
- How do designers incorporate feedback into multiple iterations of a product?

Week 3

Lesson 8: Designing Apps for Good
Unplugged

- Warm Up (10 min)
- Activity (45 min)
- Wrap Up (5 min)

To kick off the app design project, the class organizes into teams and starts exploring app topics. Several example socially impactful apps serve as inspiration for the project.
Teacher Links: Teacher Resource Student Links: Activity Guide

Lesson 9: Market Research

- Warm Up (5 min)
- Activity (40 min)
- Wrap Up (5 min)

Dive into app development by exploring existing apps that may serve similar users. Each group identifies a handful of apps that address the same topic they are working on, using those apps to help refine the app idea they will pursue.
Student Links: Activity Guide

Lesson 10: Paper Prototypes
Unplugged

- Warm Up (10-15 min)
- Activity (70 - 90 min)
- Wrap Up

Paper prototypes allow developers to quickly test ideas before investing a lot of time writing code. In this lesson teams explore some example apps created in App Lab, using those apps to help inform the first paper prototypes of their apps.
Student Links: Template | Prototype | Activity Guide

Week 4
Lesson 11: Prototype Testing
Unplugged

- Warm Up (5 min)
- Activity 1 - Testing (45 min)
- Wrap Up (5 min)

In this lesson teams test out their paper prototypes with other members of the class. With one student role playing the computer, one narrating, and the rest observing, teams will get immediate feedback on their app designs which will inform the next version of their app prototypes.
Student Links: Activity Guide | Video

Lesson 12: Digital Design
App Lab

- Warm Up (10 min)
- Activity (40-60 min)
- Wrap Up (5 min)

Having developed, tested, and gathered feedback on a paper prototype, teams now move to App Lab to build the next iteration of their apps. Using the drag-and-drop Design Mode, each team member builds out at least one page of their team's app, responding to feedback that was received in the previous round of testing.
Student Links: Activity Guide | Activity Guide | Activity Guide

Lesson 13: Linking Screens
App Lab

- Warm Up (5 min)
- Activity (45-75 min)
- Wrap Up (5 min)
- Extension Activities

Building on the screens that the class designed in the previous lesson, teams combine screens into a single app. Simple code can then be added to make button clicks change to the appropriate screen.

Week 5

Lesson 14: Testing the App

- Warm Up (5 min)
- Activity (45 min)
- Wrap Up (5 min)

Teams run another round of user testing, this time with their interactive prototype. Feedback gathered from this round of testing will inform the final iteration of the app prototypes.
Student Links: Activity Guide

Lesson 15: Improving and Iterating

- Warm Up (5 min)
- Activity 1 (30 min)
- Activity 2 (30 - 90 min)
- Wrap Up (15 min)
- Extension Activities

Using the feedback from the last round of testing, teams implement changes that address the needs of their users. Each team tracks and prioritizes the features they want to add and the bugs they need to fix.
Student Links: Activity Guide

Week 6

Lesson 16: Project - App Presentation
Project

- Warm Up (5 min)
- Activity 1 (40 - 50 min)
- Activity 2 (10 min per team)
- Wrap Up (5 min)
- Extension Activities

Each team prepares a presentation to "pitch" the app they've developed. This is the time to share struggles, triumphs, and plans for the future.
Student Links: Slide Deck | Exemplar | Rubric

Chapter Commentary

This chapter is focused on a long running group project that allows students to apply all they've learned about User-Centered Design to develop an app prototype.

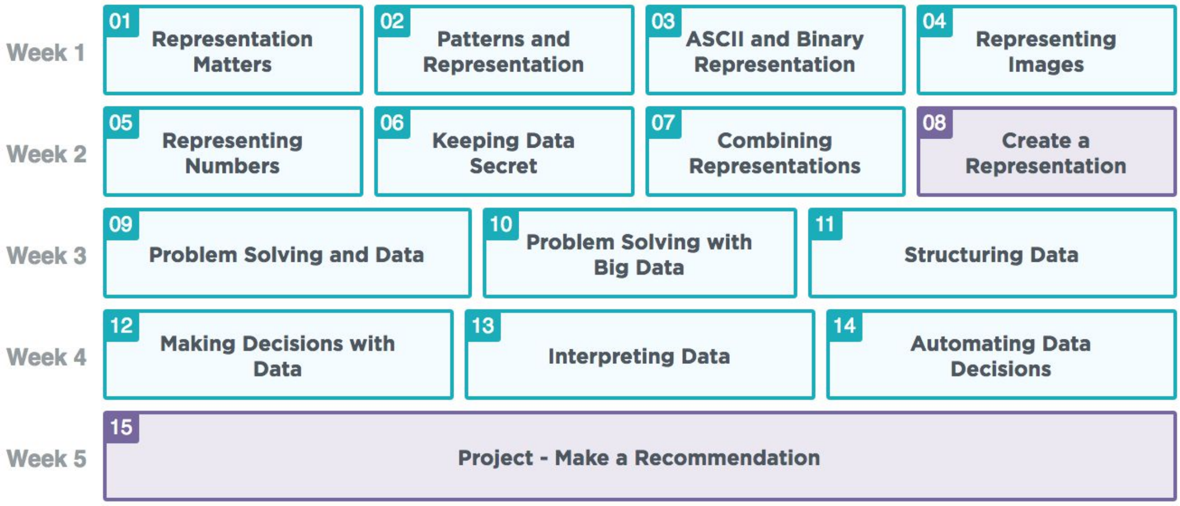
Working in teams, students identify a social issue that they care about and design and prototype an app to address that issue. This is an opportunity for students to explore other roles in software development, such as product management, marketing, design, and testing.

CSD Unit 5 - Data and Society

https://curriculum.code.org/csd/unit5/

Unit 5 is about the importance of data in solving problems and highlights how computers can help in this process. The first chapter explores different systems used to represent information in a computer and the challenges and tradeoffs posed by using them. In the second chapter you'll learn how collections of data are used to solve problems, and how computers help to automate the steps of this process. The chapter concludes by considering how the data problem solving process can be applied to an area of the your choosing.

https://studio.code.org/users/sign_in



Week 1	01 Representation Matters	02 Patterns and Representation	03 ASCII and Binary Representation	04 Representing Images
Week 2	05 Representing Numbers	06 Keeping Data Secret	07 Combining Representations	08 Create a Representation
Week 3	09 Problem Solving and Data	10 Problem Solving with Big Data	11 Structuring Data	
Week 4	12 Making Decisions with Data	13 Interpreting Data	14 Automating Data Decisions	
Week 5	15 Project - Make a Recommendation			

Chapter 1: Representing Information

Big Questions

- Why is representation important in problem solving?
- What features does a representation system need to be useful?
- What is necessary to create useable binary representation systems?
- How can we combine systems together to get more complex information?

Week 1

Lesson 1: Representation Matters

Unplugged

- Warm Up (10 mins)
- Activity (40 mins)
- Wrap Up (5 mins)

This first lesson provides an overview of what data is and how it is used to solve problems. Groups use a data set to make a series of meal recommendations for people with various criteria. Afterwards, groups compare their responses and discuss how the different representations of the meal data affected how the students were able to solve the different problems.
Teacher Links: Resource | Exemplar Student Links: Activity Guide

Lesson 2: Patterns and Representation

Unplugged

- Warm Up (10 mins)
- Activity (40 mins)
- Wrap Up (10 mins)

This lesson looks closer at what is needed to create a system of representation. Groups create systems that can represent any letter in the alphabet using only a single stack of cards, then create messages with their systems and exchange with other groups to ensure the system worked as intended. The class discusses commonalities between working systems while recognizing that there are many possible working solutions.
Teacher Links: Manipulative | Exemplar Student Links: Activity Guide

Lesson 3: ASCII and Binary Representation

Unplugged

- Warm Up (5 mins)
- Activity (40 mins)
- Wrap Up (10 mins)

This lesson introduces a formal binary system for encoding information, the ASCII system for representing letters and other characters. At the beginning of the lesson the teacher introduces the fact that computers must represent information using either "on" or "off". The class is then introduced to the ASCII system for representing text using binary symbols and practices using this system before encoding their own messages using ASCII.
Teacher Links: Exemplar Student Links: Activity Guide

Lesson 4: Representing Images

- Warm Up (5 mins)
- Activity (40 mins)
- Wrap Up (10 mins)

This lesson continues the study of binary representation systems, this time with images. The class is introduced to the concept of splitting images into squares or "pixels" which can then be turned on or off individually to make the entire image. After doing a short set of challenges using the Pixelation Widget, the class makes connections between the system for representing images and the system for representing text they learned in the previous lesson.
Teacher Links: Slides

Week 2

Lesson 5: Representing Numbers

Unplugged

- Warm Up (10 mins)
- Activity (40 mins)
- Wrap Up (10 mins)

This lesson introduces the binary number system. With a set of cards that represent the place values in a binary (base-2) number system, the class turns bits "on" or "off" by turning cards face up and face down, then observes the numbers that result from these different patterns. Eventually, the pattern is extended to a generic 4-bit system.

Teacher Links: Exemplar Student Links: Activity Guide | Manipulative

Lesson 6: Keeping Data Secret

- Warm Up (5 mins)
- Activity (40 mins)
- Wrap Up (5 mins)

Students have a discussion on the different levels of security they would like for personal data. Once the class has developed an understanding of the importance of privacy, they learn about the process of encrypting information by enciphering a note for a partner and deciphering the partner's note. The class concludes with a discussion about the importance of both physical and digital security.

Teacher Links: Exemplar Student Links: Activity Guide

Lesson 7: Combining Representations

- Warm Up (10 mins)
- Activity (40 mins)
- Wrap Up (5 mins)

This lesson combines all three types of binary representation systems (ASCII characters, binary number, and images) to allow for the encode of more complex types information in a record. After seeing a series of bits and being asked to decode them, the class is introduced to the idea that understanding binary information requires understanding both the system that is being used and the meaning of the information encoded.

Teacher Links: Exemplar Student Links: Activity Guide | Reference

Lesson 8: Create a Representation

Unplugged | Project

- Warm Up (5 mins)
- Activity (40 mins)
- Wrap Up (10 mins)

The class designs structure to represent their perfect day using the binary representation systems they've learned in this chapter. After deciding which pieces

of information the record should capture, the class will decide how a punch card of bytes of information will be interpreted to represent those pieces of information. Afterwards, everyone will use the ASCII, binary number, and image formats they have learned to represent their perfect days try to decipher what a partner's perfect day is like.
Teacher Links: Exemplar Student Links: Project Guide | Rubric

Chapter Commentary

This chapter focuses on data representation and its role in solving information problems. Students learn what a representation system needs to be useful, and how computers are able to represent different types of information using binary systems. For the chapter project, students represent their perfect day in a binary punch card and trade with classmates to decipher.

Chapter 2: Solving Data Problems

Big Questions

- How does data help us to solve problems?
- How do computers and humans use data differently?
- What parts of the data problem solving process can be automated?
- What kinds of problems do computers use data to solve in the real world?

Week 3

Lesson 9: Problem Solving and Data

- Warm up (5 mins)
- Problem Solving with Data (70 min)
- Wrap Up

This lesson covers how the problem solving process can be tailored to deal with data problems, in particular. The class is tasked with deciding what a city most needs to spend resources on. They must find and use data from the Internet to support their decision.
Teacher Links: Exemplar Student Links: Activity Guide

Lesson 10: Problem Solving with Big Data

- Warm Up (5 mins)
- Activity (40 mins)
- Wrap Up (10 mins)

This lesson covers how data is collected and used by a organizations to solve problems in the real world. The class looks at three scenarios that could be solved using data and brainstorms the types of data they would want to solve them and how they could collect the data. Each scenario also includes a video about a real-world service that has solved a similar problem with data.
Student Links: Activity Guide

Lesson 11: Structuring Data

- Warm Up (5 mins)
- Visualizing Data (70 mins)
- Wrap Up (15 min)

This lesson goes further into the interpretation of data, including cleaning and visualizing raw data sets. The class first looks at the how presenting data in different ways can help people to understand it better. After seeing how cleaning and visualization can help people make better decisions, the class looks at what parts of this process can be automated, and what need a human.

Teacher Links: Exemplar Student Links: Activity Guide

Week 4

Lesson 12: Making Decisions with Data

Unplugged

- Warm Up (5 mins)
- Activity (40 mins)
- Wrap Up (10 mins)

This lesson gives the class a chance to practice the data problem solving process introduced in the last lesson. Not all questions have right answers and in some cases the class can and should decide that they should collect more data. The lesson concludes with a discussion of how different people could draw different conclusions from the same data, or how collecting different data might have affected the decisions they made.

Teacher Links: Exemplar Student Links: Activity Guide

Lesson 13: Interpreting Data

- Warm Up (10 mins)
- Activity (40 mins)
- Wrap Up (5 mins)

Students begin the lesson by looking at a cake preference survey that allows respondents to specify both a cake and an icing flavor. They discuss how knowing the relationship between cake and icing preference helps them better decide which combination to recommend. They are then introduced to cross tabulation, which allows them to graph relationships to different preferences. They use this technique to find relationships in a preference survey, then brainstorm the different types of problems that this process could help solve.

Student Links: Activity Guide | Resource

Lesson 14: Automating Data Decisions

- Warm Up (5 mins)
- Activity (40 mins)
- Wrap Up (5 mins)

In this lesson students look at a simple example of how a computer could be used to complete the decision making step of the data problem solving process. Students are given the task of creating an algorithm that could suggest a vacation spot.

Students then create rules, or an algorithm, that a computer could use to make this decision automatically. Students share their rules and what choices their rules would make with the class data. They then use their rules on data from their classmates to test whether their rules would make the same decision that a person would. The lesson concludes with a discussion about the benefits and drawbacks of using computers to automate the data problem solving process.
Student Links: Resource | Activity Guide

Week 5
Lesson 15: Project - Make a Recommendation
Project

- Warm Up (10 mins)
- Activity (150 mins)
- Wrap Up (10 mins)
- Programming Extension (120 mins)

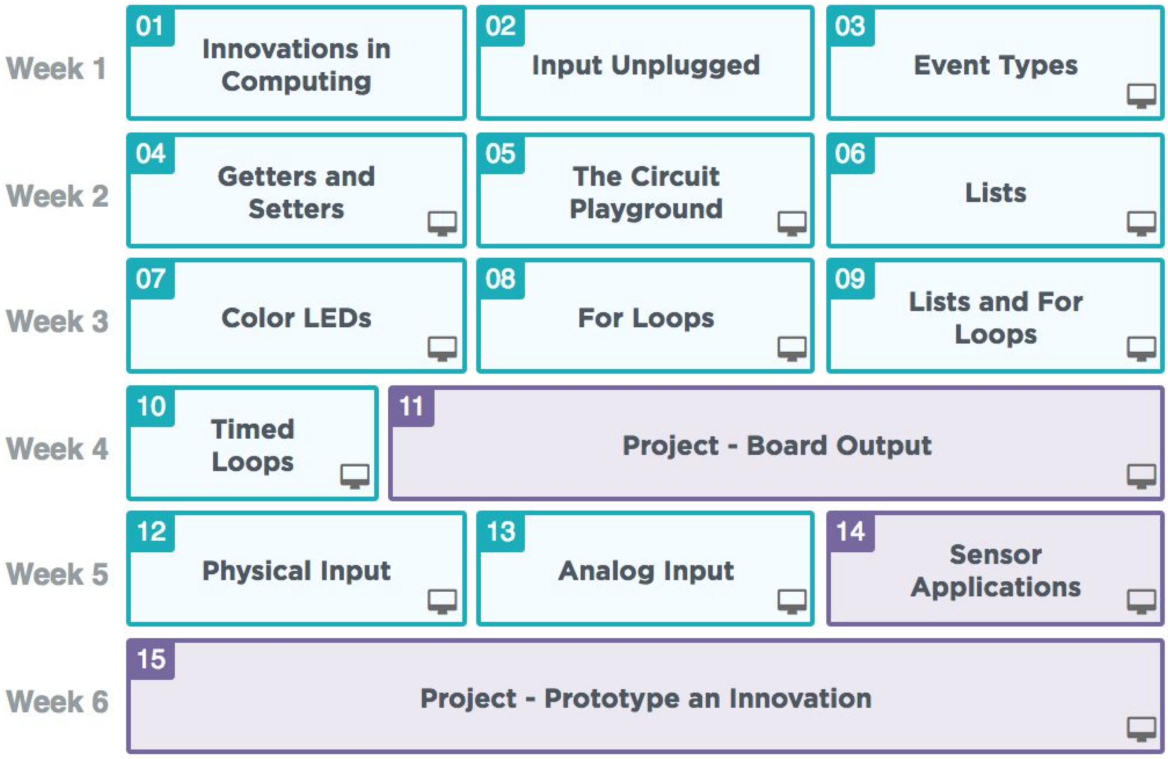
To conclude this unit the class designs ways to use data to make a recommendations or predictions to help solve a problem. In the first several steps the class brainstorms problems, performs simple research, and defines a problem of their choosing. They then decide what kind of data they want to collect, how it could be collected, and how it could be used, before exchanging feedback and giving a final presentation.
Teacher Links: Exemplars Student Links: Project Guide | Peer Review | Rubric

Chapter Commentary
Students explore how data can be used to answer interesting questions and solve problems. Using a modified version of the general Problem Solving Process, students look at how computers and humans use data differently and the pros and cons of automating problem solving. After learning ways that computers use data in the real world, students choose their own problem and use data to address it.

CSD Unit 6 - Physical Computing

https://curriculum.code.org/csd/unit6/

Unit 6 explores the role of hardware platforms in computing and how different sensors can provide more effective input and output than the traditional keyboard, mouse, and monitor. Using App Lab and Adafruit's Circuit Playground, you'll develop programs that utilize the same hardware inputs and outputs that you see in the smart devices, looking at how a simple rough prototype can lead to a finished product. The unit concludes with a design challenge to use the Circuit Playground as the basis for an innovation of your own design.



Week 1	01 Innovations in Computing	02 Input Unplugged	03 Event Types 🖥
Week 2	04 Getters and Setters 🖥	05 The Circuit Playground 🖥	06 Lists 🖥
Week 3	07 Color LEDs 🖥	08 For Loops 🖥	09 Lists and For Loops 🖥
Week 4	10 Timed Loops 🖥	11 Project - Board Output 🖥	
Week 5	12 Physical Input 🖥	13 Analog Input 🖥	14 Sensor Applications 🖥
Week 6	15 Project - Prototype an Innovation 🖥		

https://curriculum.code.org/csd/unit6/

Chapter 1: Hardware Output

Big Questions

- How does software interact with hardware?
- How can computers communicate information with simple hardware outputs?
- How can programs be made to repeat tasks?

Week 1

Lesson 1: Innovations in Computing

Research | Unplugged

Explore a wide variety of new and innovative computing platforms while expanding your understanding of what a computer can be.
Student Links: Activity Guide | Video | Video

Lesson 2: Input Unplugged
Unplugged
Experience two different ways that an app can take input from a user, while learning more about the event-driven programming model used in App Lab.
Student Links: Activity Guide

Lesson 3: Event Types
App Lab
Review the basics of programming in App Lab and explore some new event types that can be used in your programs.
Teacher Links: Teacher Guide Student Links: Activity Guide
Week 2

Lesson 4: Getters and Setters
App Lab
By reading and changing the content on the screen of an app, the class starts to build apps that only need a single screen. Even with just one screen, these techniques allow for lots of user interaction and functionality.

Lesson 5: The Circuit Playground
App Lab | Maker Toolkit
Get to know the Circuit Playground, the circuit board that will be used throughout the rest of this unit. Using App Lab, develop programs that use the Circuit Playground for output.

Lesson 6: Lists
App Lab
Learn how lists can be used to store multiple values in a single variable name.

Week 3
Lesson 7: Color LEDs
App Lab | Maker Toolkit
Using the concept of lists from the previous lesson, the class writes programs that control the ten color LEDs on the Circuit Playground.

Lesson 8: For Loops
Unplugged | App Lab
The class learns about repeating instructions using a for loop, first by controlling a "robot" through a grid, and then in App Lab.
Student Links: Activity Guide

Lesson 9: Lists and For Loops
App Lab
Combining lists and for loops allows you to write code that impacts every element of a list, regardless of how long it is. The class uses this structure to write programs that process all of the elements in lists, include the list of color LEDs.

Week 4
Lesson 10: Timed Loops
App Lab | Maker Toolkit
In this lesson we build on students understanding of for loops to learn about a more sophisticated structure called a timed loop. Unlike for loops, you can control how quickly a timed loop repeats, and other code (such as event handlers) can be run between iterations of the loop. This allows us to create interesting animations that repeat infinitely, while still allowing other code to run alongside it.

Lesson 11: Project - Board Output
App Lab | Maker Toolkit | Project
Students take what they've learned through chapter one, and develop an app of their own design that uses the board to output information.
Student Links: Project Guide | Rubric

Chapter Commentary
This unit begins with a review of programming and an introduction to the event-driven model that students will use in the unit. Using the Circuit Playground, they explore ways to output information through the elements on the board. Students learn about for loops, lists, and timed loops to better control the output. In the chapter project, they create an app that uses the board to create an app that uses hardware as an output.

Chapter 2: Hardware Input
Big Questions

- How can computers sense and respond to their environment?
- How can complex real-world information be represented in code?
- How can simple hardware be used to develop innovative new products?

Week 5
Lesson 12: Physical Input
App Lab | Maker Toolkit
Using the hardware buttons and switch, the class develops programs that use the Circuit Playground as an input.

Lesson 13: Analog Input
App Lab | Maker Toolkit

Explore the analog inputs on the Circuit Playground, writing programs that respond to the environment through sensors.
Student Links: Video

Lesson 14: Sensor Applications
App Lab | Maker Toolkit | Project
Work through three small apps to better understand the uses of the sensors from the previous lesson.
Student Links: Project Guide

Week 6
Lesson 15: Project - Prototype an Innovation
App Lab | Maker Toolkit | Project
Develop innovative computing devices of your own design, using everything you've learned throughout this course.
Student Links: Project Guide | Peer Review | Rubric

Chapter Commentary
Students transition from using hardware as simply a tool for output and begin using the buttons and sensors as a form of input for their programs. After exploring the different ways in which the board can be used for user interaction, students wrap up the course be designing and prototyping an innovative computing device of their own.

Projects:

To access projects for various ages:

https://studio.code.org/projects/public

Links:

Create a login at

https://studio.code.org/courses

For more information/content and resources:
www.code.org

SYLLABUS:
https://studio.code.org/courses?view=teacher

Creating an iPad app

http://gethopscotch.com/

Learning to make a Website

https://thimble.mozilla.org/en-US/

Learning to program with robots

https://www.lego.com/en-us/mindstorms/

Hour of Code Tutorials

https://code.org/hourofcode/overview