

Welcome to Computer Science Fundamentals

Welcome to Computer Science Fundamentals, the Code.org curriculum designed for students in kindergarten through fifth grade (K-5), which includes students 5-11 years old! This guide begins with an introduction to the CS Fundamentals curriculum, provides a look into our core values and methods, and includes a detailed overview of each course offering.

We built CS Fundamentals for elementary school educators and students. No prior experience is assumed. These courses work best in a classroom setting, led by an engaged teacher. For scenarios where students will be learning independently or asynchronously, we recommend our CS Fundamentals Express courses. Teachers do not need previous computer science experience and are encouraged to learn along with their students.

CS Fundamentals Curriculum At-A-Glance

Which course is right for my students?

Grade-Aligned Courses - There are six CS Fundamentals courses, each aligned with an elementary grade level. Each course is an entry point, and there is no requirement to do all (or any) of the previous courses first.

CS Fundamentals Express Courses - In addition to courses A-F, CS Fundamentals also offers two express courses designed for situations where the teacher is a less active role in engaging students. An express course might be used, for example, if a student is learning CS on her own.

Express courses consist primarily of skill-building lessons done on the computer. These courses can be delivered with or without an actively involved teacher, making them a good fit for self-paced classrooms, stations, or independent learning. The table below maps courses A-F and the two express courses:

Course A <i>Kindergarten</i>	Course B <i>1st Grade</i>	Course C <i>2nd Grade</i>	Course D <i>3rd Grade</i>	Course E <i>4th Grade</i>	Course F <i>5th Grade</i>
Pre-reader Express Course <i>Grades K-1</i> Built with lessons from Courses A - B		Express Course <i>Grades 2-5</i> Built with lessons from Courses C - F			

Our Vision

Code.org's vision is that every student in every school should have the opportunity to learn computer science (code.org/about). Our curriculum is designed so that an empowered teacher can lead a diverse group of students through experiences that are supportive, equitable, engaging, and lead to valuable learning (code.org/educate/curriculum/values).

Historically this vision has contrasted sharply with reality. Until recently, most schools did not offer computer science at all, and what classes there were notoriously lacked in diversity. Additionally, many students found these classes unengaging, intimidating, or simply disconnected from their lived experiences with technology. Thanks to efforts by many organizations and individuals, this world is beginning to change: many more schools now offer computer science courses; more diverse students take those courses; and more engaging, relevant, and equitable pedagogy has become the established norm. Even so, there is much work still to be done. This course is designed to continue this momentum as the collective CS education community moves towards this vision of an equitable CS education system.

How We Support Our Vision

Many aspects of Code.org's CS Discoveries curriculum are designed to bring about the eventual change we aim to see more broadly in CS education. Some of the most significant features are listed below.

Free and open: We make our curriculum, videos, and tools free and open for anyone to adopt.

Prioritize New-to-CS Teachers: Historically, only a few schools could hire trained computer scientists as teachers, which severely limited which schools could offer a CS course. Reaching all schools has meant developing our CS Discoveries course with the understanding that most of our teachers are new-to-CS and prioritizing their needs. As such, our course includes some distinctive features.

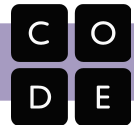
- Comprehensive lesson plans and resources designed to ensure new-to-CS teachers have everything they need to implement the course
- Clear and consistent pedagogy to help new-to-CS teachers develop best practices as CS teachers
- High-quality videos that help teachers introduce and explain CS concepts
- An associated professional learning program that pays particular attention to the needs of new-to-CS teachers

Equitable Pedagogy: Our curriculum is designed to promote an equitable classroom environment for all students, with particular attention paid to the experiences of historically excluded groups, most notably young women and Black, Hispanic, and Native American students. Drawing from extensive feedback from our classrooms, as well as CS education research, our course includes many features designed to support and prioritize these students:

- Pedagogy that develops a collaborative and supportive classroom environment
- Specific attention paid to language demands of our lessons
- Projects and activities that highlight a variety of applications of computing and frequently ask students to incorporate their own backgrounds and interests.
- Curriculum videos that feature a cast of diverse role models in terms of race, gender, and profession who empower our diverse students to "see themselves" as part of the world of computing
- A professional learning program that highlights these features and helps teachers reflect on how best to implement them within their own classroom

Join Us in this Vision

We think our vision is audacious and deeply motivating. If you feel the same, the best way to join us in this vision is to teach this course! We know that for many teachers this represents a significant undertaking, and we have aimed to do our best to help share the load. Based on the feedback of many teachers we know it will be a challenging, but ultimately



gratifying experience. Code.org is here to support you, and we look forward to your feedback so that we can continue to make CS Discoveries an even better experience for our students and teachers.

Materials and Resources

Our materials and tools are specifically created with a focus on foundational concepts and are designed to support exploration and discovery. This allows students to develop an understanding of these concepts through “play” and experimentation. From our coding tools to our non-coding tools and videos, all our resources have been engineered to support the lessons in our curriculum, and, thus our philosophy about student engagement and learning. Together, these resources typically allow the teacher to act in the role of facilitator and coach when addressing unfamiliar material. In instances when the teacher acts as the primary source of information, generous supports are provided.

All resources below can be accessed free of charge at code.org/csf.

Lesson Plans

- Instructional guides for every lesson
- Activity Guides and handouts for students
- Formative and summative assessments
- Exemplars, rubrics, and teacher dashboard

Slide Decks

- Slide decks provide step-by-step instructions, examples, and interactive activities that align with curricular objectives. The slides designed to be student facing to enhance learning and understanding.

Videos

- Tutorials, instructional videos, and inspirational videos

Tools

Maze/Collector Puzzles - Recognizable by the 8x8 square grid, these puzzles ask students to move an on-screen character to complete a specified goal, sometimes collecting objects or performing other actions along the way. These levels feature characters from popular media franchises and others that are unique to Code.org.

Debugging Feature Step Button - Most Maze and Collector puzzles include a step button, which allows you to step through a program block by block to see what happens each step of the way. This debugging feature enables students to slow their programs down, making it easier to spot coding mistakes.

Play Lab - Featuring a diverse cast of fun actors, Play Lab allows students to create their own stories and simple games. Actors can speak, move and even express emotions. Play Lab projects are interactive, and controlled by the user’s keyboard and mouse.

Sprite Lab - Sprite Lab is the most open-ended programming tool found in CS Fundamentals. It enables students to create fun animations in no time but is also a powerful tool for creating everything from interactive posters to games to simulations.

Other Online Tools - Created as part of Code.org’s Hour of Code tutorials, more unique tools are found in various lessons across CS Fundamentals. These tools allow students to explore everything from making games to the basics of machine learning and artificial intelligence.

Tech Requirements and Required Materials

Technical Requirements

The course requires and assumes that each student has access to an internet-connected computer every day in class. Each computer must have a modern web browser installed. The course tools and resources (lesson plans, teacher dashboard, videos, student tools, programming environment, etc.) are online and accessible through a web browser. For more information about tech setup, go to: code.org/educate/it.

While the course features many “unplugged” activities away from the computer, daily access to a computer is essential for every student. PDFs of handouts, worksheets, and readings are available on the course website.

Suggested Materials / Supplies

Many lessons have handouts that guide students through activities. These resources can be printed or assigned digitally.

Some lessons call for typical classroom supplies and manipulatives. See below for information on specific lessons:

Course	Lesson	Materials
Course C	My Robotic Friends Jr.	Plastic cups (10/group of 2-3)*
	Binary Bracelets	Markers. Optional – 18 black/18 white beads, one pipe cleaner per student
	My Loopy Robotic Friends Jr.	Paper cups (20/group of 4)
Course D	Dance Party	Optional - Headphones (1/student)
	Conditionals with Cards	Deck of cards or something similar (1/group of 4-6)*
Course E	Digital Sharing	Smartphone or tablet
Course F	The Power of Words	Colored Pencils, string the length of the classroom
	Blank Space Stories	Envelopes (1-4/group of 2-4)
	For Loop Fun	Dice (3 dice/group of 2-4)*

* You can easily re-use these items between multiple classes

Getting Help

The curriculum is completely free for anyone to teach anywhere. For support, click on the question mark in the upper right-hand corner of the website.

Here, you'll find our “Help and support” forum, where you can email us or find how-to articles. You'll also see a link to our “Teacher community” forums where you can connect to other teachers for support, teaching tips, or best practices.

When you're in a puzzle, you'll see an additional “Report a problem” link for that puzzle. Thank you for helping us find and fix any issues.

CS Fundamentals Curriculum Overview

Concepts

At the highest level, CS Fundamentals courses are organized into **concept chunks**. Concept chunks provide structure around the big ideas in a course. Concept chunks make it easy for teachers to see at a glance what is covered in a course.


Course A (13 lessons)	Course B (13 lessons)	Course C (16 lessons)
Digital Citizenship (2 lessons)	Digital Citizenship (1 lesson)	Digital Citizenship (1 lesson)
Sequencing (4 lessons)	Sequencing (4 lessons)	Sequencing (5 lessons)
Loops (4 lessons)	Loops (4 lessons)	Loops (4 lessons)
Events (2 lessons)	Impacts of Computing (1 lesson)	Events (3 lessons)
End of Course Project (1 lesson)	Events (2 lessons)	Data (2 lessons)
	End of Course Project (1 lesson)	End of Course Project (1 lesson)
Course D (17 lessons)	Course E (17 lessons)	Course F (16 lessons)
Digital Citizenship (1 lesson)	Digital Citizenship (4 lessons)	Digital Citizenship (1 lesson)
Sequencing (3 lessons)	Sprites (3 lessons)	Sprites (4 lessons)
Computing Systems (1 lesson)	Game Design (5 lessons)	Variables (6 lessons)
Events (4 lessons)	Functions (3 lessons)	Data & Simulations (4 lessons)
Loops (3 lessons)	Conditionals (3 lessons)	End of Course Project (1 lesson)
Conditionals (5 lessons)	Impacts of Computing (2 lessons)	
End of Course Project (1 lesson)	End of Course Project (1 lesson)	

Each CS Fundamentals course has several concept chunks, and each chunk can have one or several lessons. In the next section, you can find more details about the types of lessons in CS Fundamentals.

Course Overviews

Course A


Course A offers a computer science curriculum for beginning readers, including Kindergarten students. Students will learn to program using commands like loops and events. The lessons featured in this course also teach students to meaningfully collaborate with others, investigate different problem-solving techniques, persist in the face of challenging tasks, and learn about internet safety.

Concept	#	Lesson Name	Overview
Digital Citizenship	1	 common sense education® Safety in My Online Neighborhood	The power of the internet allows students to experience and visit places they might not be able to see in person. On this virtual field trip, kids can practice staying safe on online adventures.
	2	Learn to Drag and Drop	This skill-building lesson will give students an idea of what to expect when they head to the computer lab. It begins with a brief discussion introducing them to computer lab manners. Then they will progress into using a computer to complete online puzzles.
Sequencing	3	Happy Maps	This context-setting lesson brings together teams with a simple task: get the "flurb" to the fruit. Students will practice writing precise instructions as they work to translate instructions into the symbols provided.
	4	Sequencing with Scrat	In this skill-building lesson, students will develop sequential algorithms to move a squirrel character from one side of a maze to the acorn on the other side. To do this, they will stack code blocks together in a linear sequence.
	5	Programming with Scrat	In this skill-building lesson, students will continue to develop sequential algorithms.
	6	Programming with Rey and BB-8	In this skill-building lesson, students will use their newfound programming skills in more complicated ways to navigate a tricky course with the Star Wars character BB-8.
Loops	7	Happy Loops	This context-setting lesson revisits Happy Maps. This time, students will use loops to solve bigger, longer puzzles with their code.
	8	Loops with Scrat	In this skill-building lesson, students will practice loops in programming puzzles where the goal is to help the squirrel reach the acorn.
	9	Loops with Laurel	In this skill-building lesson, students continue learning the concept of loops. Here, students use loops to collect treasure in open cave spaces.

Loops (cont.)	10	Ocean Scene with Loops	In this skill-building lesson, students learn to draw images by looping simple sequences of instructions. Here, loops are creating patterns. At the end of this lesson, students will create their own images.
Events	11	The Big Event Jr.	In this context-setting lesson, the class will experience the concept of events through a game where they move or shout when you press buttons on a giant remote.
	12	Mini-Project: On the Move with Play Lab	In this mini-project , students will use events in Play Lab and apply all of the coding skills they've learned to create an animated game. It's time to get creative and make a story in the Play Lab!
Project	13	End of Course Project	In this project lesson, students apply what they have learned about sequencing and loops with the Artist.

Course B

We developed Course B with first graders in mind. While the concepts in Course B parallel those in Course A, students learn more sophisticated unplugged activities and work through a greater variety of 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 create games in the Play Lab programming environment.

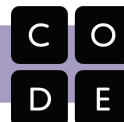
Concept Chunk	#	Lesson Name	Description
Digital Citizenship	1	 common sense education Digital Trails	Students learn that the information they share online leaves a digital footprint or "trail." Students compare different trails and think critically about what kinds of information they want to leave behind.
Sequencing	2	Move It, Move It	Students will program their classmates to step carefully around a large grid on the floor in this context-setting lesson.
	3	Sequencing with Angry Birds	In this skill-building lesson, students will develop sequential algorithms to move a bird from one side of a maze to the pig on the other side. To do this, they will stack code blocks together in a linear sequence.
	4	Programming with Angry Birds	In this skill-building lesson, students will continue to develop sequential algorithms.
	5	Programming with Harvester	Students will apply the programming concepts that they have learned to the Harvester environment in this skill-building lesson. Students will continue to develop sequential algorithm skills and start using the debugging process.
Loops	6	Getting Loopy	In this context-setting lesson, students will learn to use loops to more easily communicate instructions by looking at the repeated patterns of a dance.
	7	Loops with Harvester	In this skill-building lesson, students will help the harvester collect crops by using loops.
	8	Loops with Laurel	In this skill-building lesson, students continue learning the concept of loops. Here, Laurel the Adventurer uses loops to collect treasure in open cave spaces.
	9	Drawing Gardens with Loops	In this skill-building lesson, students learn to draw images by looping simple sequences of instructions. Here, students use loops to create patterns.
Impacts of Computing	10	The Right App	In this exploratory lesson , students learn about different computing technologies that exist. They also compare how people lived before the invention of apps and computers with how people live now and then empathize with several fictional smartphone users to help them design the "right app" that addresses their needs.

Events	11	The Big Event Jr.	In this context-setting lesson, the class will experience the concept of events through a game where they move or shout when you press buttons on a giant remote.
	12	Mini-Project: A Royal Battle with Events	In this mini-project , students will have the opportunity to learn how to use events in Play Lab and apply all of the coding skills that they've learned to create an animated game.
Project	13	End of Course Project	In this project lesson, students apply what they have learned about sequencing and events in an open-ended project with Play Lab.

Course C

We developed Course C for students in second grade. Students will create programs with sequencing, loops, and events. They will investigate problem-solving techniques and develop strategies for building positive communities both online and offline. By the end of the course, students will create interactive games that they can share.


Concept Chunk	#	Lesson Name	Description
Digital Citizenship	1	 Putting a STOP to Online Meanness	The internet is filled with all kinds of interesting people, but sometimes, some of them can be mean to each other. With this role play, help your students understand why it's often easier to be mean online than in person and how to deal with online meanness when they see it.
Sequencing	2	My Robotic Friends Jr.	In this context-setting lesson, students will use a set of symbols to instruct a "robot" to stack cups in different patterns. Students will take turns participating as the robot, responding only to the algorithm defined by their peers.
	3	Programming with Angry Birds	In this skill-building lesson, students will develop sequential algorithms to move a bird from one side of a maze to the pig on the other side. To do this, they will stack code blocks together in a linear sequence, making them move straight, turn left, or turn right.
	4	Debugging in Maze	In this skill-building lesson, students will encounter pre-written code that contains mistakes. They will need to step through the existing code to identify errors.
	5	Collecting Treasure with Laurel	In this skill-building lesson, students will continue to develop their understanding of algorithms and debugging. With a new character, Laurel the Adventurer, students will create sequential algorithms to get Laurel to pick up treasure as she walks along a path.
	6	Creating Art with Code	In this skill-building lesson, students will take control of the Artist to complete drawings on the screen.
Loops	7	My Loopy Robotic Friends Jr.	This context-setting lesson builds on the initial "My Robotic Friends" activity, featuring larger and more complicated designs.
	8	Loops with Rey and BB-8	This skill-building lesson has students using loops to help the Star Wars character BB-8 efficiently traverse a maze.
	9	Harvesting Crops with Loops	Students loop new actions to help the harvester collect multiple veggies growing in large bunches in this skill-building lesson.
	10	Mini-Project: Sticker Art	This mini-project lesson builds on the understanding of loops from previous lessons. Students will create unique artwork with the Artist.
Events	11	The Big Event	In this context-setting lesson, students will learn that events are a great way to add flexibility to a pre-written algorithm.
	12	Build a Flappy Game	This lesson combines skill-building around events with a mini-project where students get to build their own Flappy Bird game. At the end of the lesson, students will customize their game by changing the visuals or rules.
	13	Mini-Project: Chase Game	This lesson combines skill-building around events with a mini-project where students get to build their own animated game.
Data	14	Picturing Data	Students will collect data from a Play Lab project and visualize it using different graphs in this exploratory lesson.



Data (cont.)	15	Binary Bracelets	This exploratory lesson introduces the concept of binary which is how computers today store all sorts of information.
Project	16	End of Course Project	This project lesson takes students through designing, developing, and showcasing new Play Lab projects.

Course D

We created course D for students in third grade. The course begins with a review of the concepts found in earlier courses, including loops and events. Afterward, students will develop their understanding of algorithms, nested loops, while loops, conditionals, and more.

Concept Chunk	#	Lesson Name	Description
Digital Citizenship	1	 Password Power-Up	Stronger, more secure online passwords are a good idea for everyone. Use the tips in this lesson to help kids make passwords that are both secure and memorable.
Sequencing	2	Graph Paper Programming	In this context-setting lesson, students use symbols to instruct each other to color squares on graph paper. By "programming" one another to draw pictures, students get an opportunity to experience some of the core concepts of programming in a fun and accessible way.
	3	Introduction to Online Puzzles	In this skill-building lesson, students will practice their sequencing and debugging skills in maze puzzles.
	4	Debugging with Laurel	In this skill-building lesson, students will practice debugging in the "collector" environment. Students will get to practice reading and editing code to fix puzzles with simple algorithms, loops, and nested loops.
Computing Systems	5	Hardware and Software Skits	In this unplugged lesson students will act out the ways internal and external parts of a computer work together. They will take on roles such as: CPU, Keyboard, Monitor, User and more.
Events	6	Events in Bounce	In this context-setting/skill-building lesson, students will learn what events are and how programmers use them in video games. Students will build a game that they can customize with different speeds and sounds.
	7	Build a Star Wars Game	In this skill-building lesson, students will practice using events to build a game that they can share.
	8	Dance Party	In this skill-building lesson, students will program an interactive dance party.
	9	Music Lab Intro Tutorial	In this skill-building lesson, students will use Music Lab to discover the power of programming to make their own unique sounds, compose songs, and perform live music.
Loops	10	Loops in Ice Age	This context-setting / skill-building lesson will quickly introduce students to loops.
	11	Drawing Shapes with Loops	This skill-building lesson builds on the understanding of loops from the previous lesson and doubles as a debugging exercise for extra problem-solving practice.
	12	Nested Loops in Maze	In this skill-building lesson, students will learn how to program a loop inside of another loop.
Conditionals	13	Conditionals with Cards	In this context-setting lesson, students will write conditional (if/else) statements to state the rules of simple card games.

Conditionals (cont.)	14	Looking Ahead with Minecraft	This skill-building lesson gives students the chance to practice concepts that they have learned up to this point and get their first experience with conditionals!
	15	If/Else with Bee	In this skill-building lesson, your class will continue to code with conditionals, allowing them to write code that functions differently depending on the specific conditions the program encounters.
	16	While Loops in Farmer	In this skill-building lesson, students will be working to fill holes and dig dirt in Farmer, but they will not know the size of the holes or the height of the mounds of dirt. To solve these puzzles, students will use a new kind of loop.
	17	Until Loops in Maze	In this skill-building lesson, students will learn about until loops. Students will build programs that have the main character repeat actions until they reach their desired stopping point.
Project	18	End of Course Project	This project lesson takes students through the process of designing, developing, and showcasing new projects!

Course E


We created Course E for students in fourth grade. The course begins with an introduction to the Sprite Lab programming tool. Students will learn to make fun, interactive projects that reinforce what they'll learn about online safety. Following these lessons, students will engage in more complex coding. Students will learn about nested loops, functions, and conditionals. By the end of the course, students will combine these concepts to solve challenging puzzles. The course ends with an open-ended project where students create a game or drawing.

Concept Chunk	#	Lesson Name	Description
Digital Citizenship	1	common sense education Be a Super Digital Citizen	Online tools are empowering for kids, and they also come with big responsibilities. Show your students appropriate ways to take action and resolve conflicts, from being upstanders to helping others in need.
	2	common sense education Private and Personal Information	It's in our students' nature to share and connect with others. Help your students learn the difference between what's personal and what's best left private.
Sprites	3	Swimming Fish with Sprite Lab	Students will program a simple animated underwater scene in this skill-building lesson.
	4	Hello World	In this skill-building lesson, students will learn to create and animate sprites and make them interactive using events.
	5	Mini-Project: About Me	Students will apply their understanding of sharing personal and private information on the web by creating an interactive poster in this mini-project .
Game Design	6	Game Design	In this skill-building lesson, students will continue to build on animating sprites and groups of sprites with events.
	7	Variables as Score	In this skill-building lesson, students will learn how to use a variable to keep track of important, changing data in a game setting.
	8	Environment and Players	In this skill-building lesson, students will be introduced to a new game mechanic to simulate gravity and blocking environments for the purpose of creating a game.
	9	Game Jam: Create	This mini-project lesson, students will use their new skills and work together to create a short, simple game based on a randomly generated theme.
	10	Game Jam: Play	This mini-project lesson, students will conduct user tests of their new game to gather data to inform decisions to improve the game they coded previously.
Functions	11	Songwriting	This context-setting lesson will help students understand why combining chunks of code into functions can be a helpful practice.
	12	Music Lab Intro Tutorial	In this skill-building lesson, students will use Music Lab to discover the power of programming to make their own unique sounds, compose songs, and perform live music.
	13	Functions in Minecraft	In this skill-building lesson, students will begin to understand how functions can be helpful!
	14	Functions with Artist	In this skill-building lesson, students will use functions with the Artist.
Conditionals	15	Conditionals in Minecraft: Voyage Aquatic	In this context-setting lesson, students will get the chance to practice content that they have learned up to this point, as well as getting a sneak peek at conditionals!
	16	Conditionals with the Farmer	This skill-building lesson focuses on a new type of programming puzzle, featuring a farmer who needs to prepare her fields for planting crops.

	17	Functions with Harvester	In this skill-building lesson, students will use conditionals with functions to harvest crops in Harvester.
Impacts of Computing	18	Designing for Accessibility	In this exploratory lesson, students will learn about accessibility and the value of empathy through brainstorming and designing accessible solutions for hypothetical apps.
	19	Digital Sharing	This exploratory lesson helps students understand the challenges and benefits of respecting ownership and copyright in digital environments.
Project	20	End of Course Project	This project lesson takes students through the process of designing, developing, and showcasing new projects!

Course F

We created Course F for students in the fifth grade. The course begins by looking at how users make choices in the apps they use. Students then learn to make a variety of Sprite Lab apps that also offer choices for the user. In the later lessons in the course, students will learn more advanced concepts, including variables and “for” loops. Students are given greater autonomy and more choices throughout this course. By the end of the course, students will have created interactive projects and artwork to share with their friends and family.

Concept Chunk	#	Lesson Name	Description
Digital Citizenship	1	 The Power of Words	As kids grow, they'll naturally start to communicate more online. But some of what they see could make them feel hurt, sad, angry, or even fearful. Help your students build empathy for others and learn strategies to use when confronted with cyberbullying.
Sprites	2	Introducing Sprite Lab	In this context-setting lesson, students will discuss the role of computers and technology in their lives, focusing on how apps and tools give users choices about how to use them. Students will also explore apps similar to those they'll create themselves later on.
	3	Making Sprites	In this skill-building lesson, students will work through a series of programming levels on the computer, finishing with an open-ended “free play” task where they can build whatever they like. Students will write programs and learn about the two concepts at the heart of Sprite Lab: sprites and behaviors.
	4	Sprites in Action	In this skill-building lesson, students will work through a series of programming levels on the computer, finishing with an open-ended “free play” task where they can build whatever they like. Students will write programs that respond to timed events and user input.
	5	Mini-Project: Virtual Pet	Students will create an interactive Virtual Pet that looks and behaves how they wish in this mini-project lesson. Students will use Sprite Lab's "Costumes" tool to customize their pet's appearance. They will then use events, behaviors, and other concepts they have learned to bring their project to life.
Variables	6	Blank Space Stories	In this context-setting lesson, students will use fill-in-the-blank stories (similar to Mad Libs®) as a context for understanding how computers take and store input from a user, then use it later as a program runs.
	7	Text and Prompts	In this skill-building lesson, students will get practice with variables in Sprite Lab.
	8	Mini-Project: User Input Programs	Students will apply their new skills with variables to one of three open-ended mini-projects . Students will explore sample programs, create a plan for their own project, and program it from scratch.
	9	Lots of Sprites	In this context-setting/skill-building lesson , students will go deeper into Sprite Lab's capabilities and learn how to manage groups of sprites with the same costume.

Variables (cont.)	10	Counting with Variables	In this skill-building lesson , students will use variables to track a value that changes over time, like a counter. This lesson also includes a short mini-project in which students create a simple game.
	11	Mini-Project: Collector Game	In this mini-project lesson , students will create a customized "collector game", in which the user moves a sprite around to collect objects.
Data & Simulations	12	Simulating Experiments	In this context-setting lesson, students will run and collect data from a simple simulation in Sprite Lab. After running the simulation multiple times, students will have an opportunity to predict how changing a variable in the simulation might impact the outcome and test that hypothesis.
	13	Outbreak	In this skill-building lesson, students will model an outbreak simulator. Students will study the simulation to understand how a virus can spread and how to slow it down.
	14	AI for Oceans	In this exploratory lesson, students will train a machine learning model by classifying fish and other objects.
	15	The Internet	Students will pretend to flow through the internet while learning about connections, URLs, IP Addresses, and DNS in this exploratory lesson.
Project	16	End of Course Project	This project lesson takes students through the process of designing, developing, and showcasing new projects!