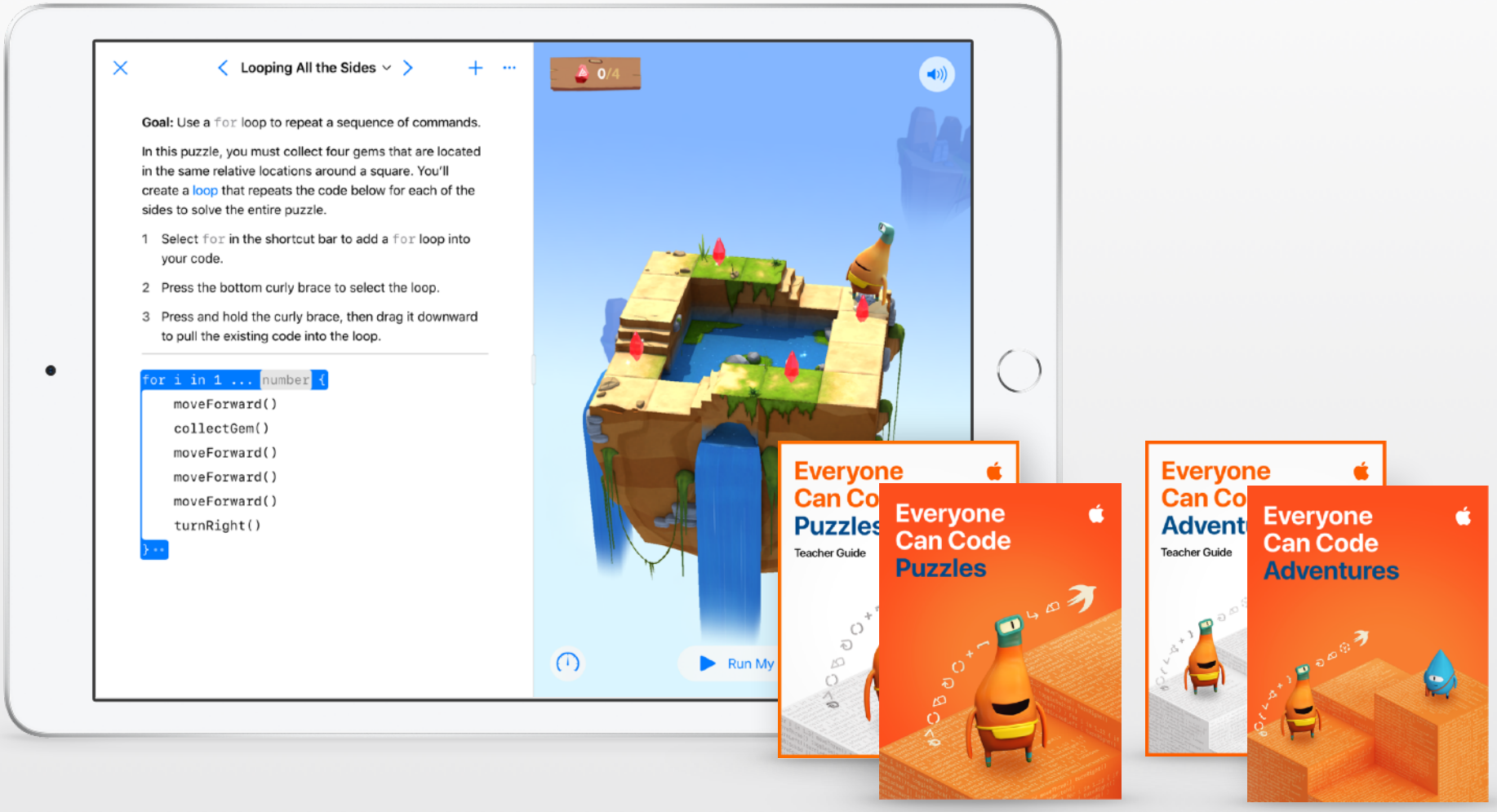
Everyone Can Code

Curriculum Guide



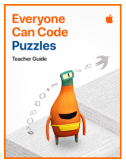
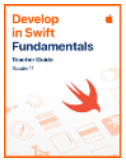
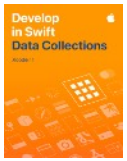
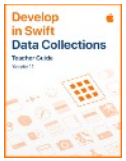
July 2020

Teaching Code with Apple

When you teach code, you're not only teaching the language of technology. You're teaching new ways to think and bring ideas to life. And coding with Swift, Apple's powerful, intuitive, and easy-to-learn programming language, provides students fun and engaging ways to prepare for the future. Every student should have the opportunity to create something that can change the world. Whether students are just getting started with Swift Playgrounds or ready to learn Xcode, Apple provides everything educators need to bring code into the classroom.

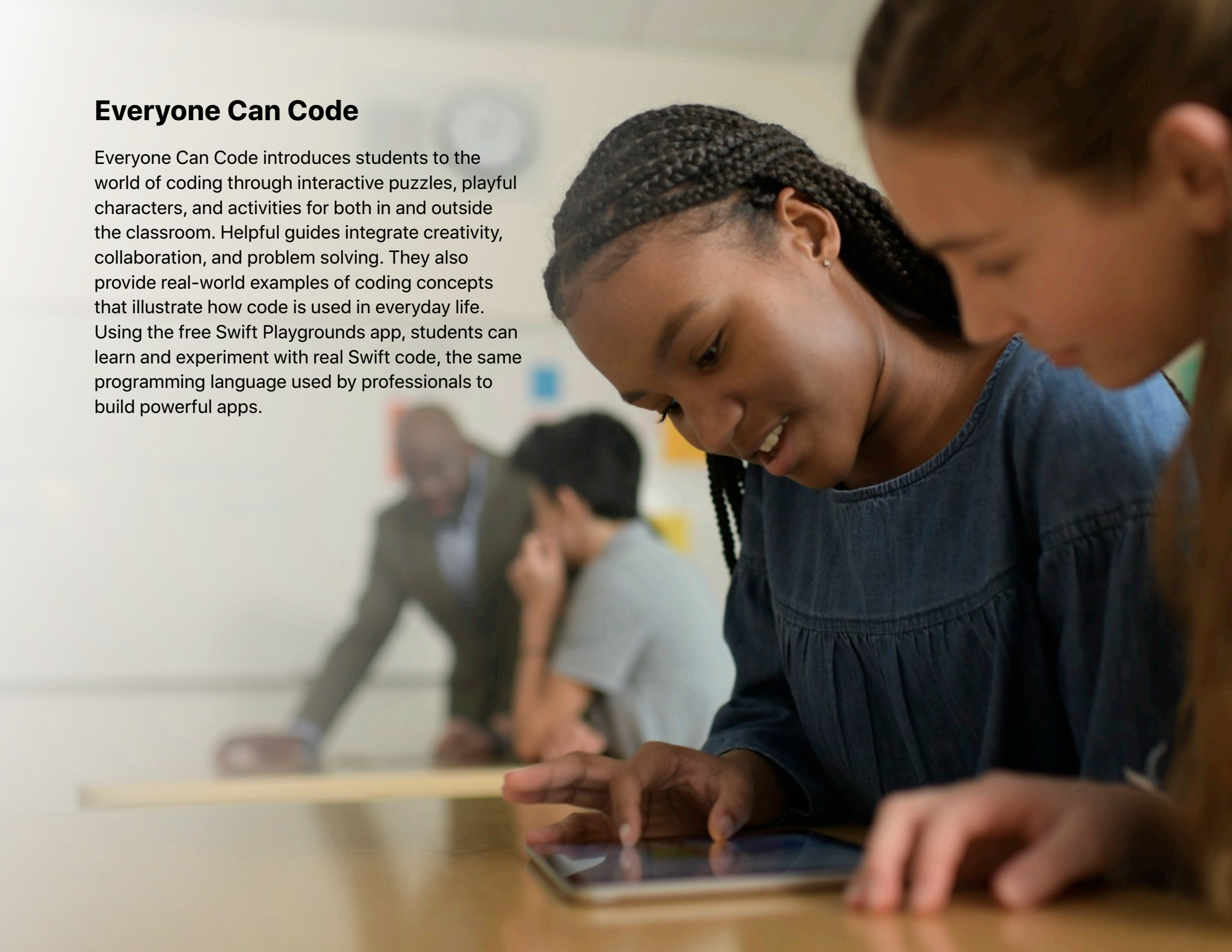
Code Curriculum Pathway

The Everyone Can Code and Develop in Swift curricula take students all the way from writing their first lines of Swift code to building their first apps. The table below provides an overview of the free teaching and learning resources available.

	Student	Teacher	Device	Audience	App	Prerequisites	Overview	Lesson hours
Everyone Can Code								
Everyone Can Code Puzzles			iPad and Mac	Grade 4 and up	Swift Playgrounds	None	Students learn foundational coding concepts, such as loops, variables, and functions. They practice and apply their skills in a variety of ways, from puzzles to open-ended playgrounds.	45 hours
Everyone Can Code Adventures			iPad	Grade 4 and up	Swift Playgrounds	Everyone Can Code Puzzles	Students learn more advanced coding concepts, such as event handling, advanced arrays, and component-based design as they scope and build their own projects.	45 hours
Develop in Swift								
Develop in Swift Explorations			Mac	High school and up	Xcode	None	Students build a solid foundation in Swift programming. An AP® CS Principles version of the course aligned with the College Board requirements is also available.	180 hours
Develop in Swift Fundamentals			Mac	High school and up	Xcode	None	Students build fundamental iOS app development skills with Swift. They'll master core concepts and practices that Swift programmers use and build a basic fluency in Xcode source and UI editors.	180 hours
Develop in Swift Data Collections Coming soon			Mac	High school and up	Xcode	Develop in Swift Fundamentals	Students extend their knowledge and skills in iOS app development, creating more complex, capable apps. They'll work with data from a server and explore iOS APIs that allow for richer app experiences.	180 hours

Everyone Can Code

Everyone Can Code introduces students to the world of coding through interactive puzzles, playful characters, and activities for both in and outside the classroom. Helpful guides integrate creativity, collaboration, and problem solving. They also provide real-world examples of coding concepts that illustrate how code is used in everyday life. Using the free Swift Playgrounds app, students can learn and experiment with real Swift code, the same programming language used by professionals to build powerful apps.



Curriculum Overview

The comprehensive Everyone Can Code curriculum introduces students to core coding concepts using the Swift programming language. Students use real Swift code to solve puzzles and create projects in the fun, interactive Swift Playgrounds app. The app comes with a complete set of Apple-designed lessons called Learn to Code, as well additional coding challenges and playgrounds to program robots and connected devices.

Everyone Can Code includes teacher and student guides that offer flexible, engaging lessons designed for hour-long class periods or to be taught across a whole semester. Each chapter introduces a coding concept through a familiar context, then supports students in solving puzzles with confidence in Swift Playgrounds. They apply their skills to creative coding projects in a range of playgrounds and reinforce their learning by considering how coding concepts shape digital technology in their everyday lives.

The teacher guide is designed to support all teachers—with or without coding experience. It provides inclusive learning experiences, assessment ideas, and tips to support a range of learners.

And if you can't bring it into your school day, or if you want to explore just a piece of this to get started, we've provided a series of modules in our Swift Coding Club kits that leverage all these great resources in a fun and approachable way for out-of-school time.

Explore Swift Playgrounds

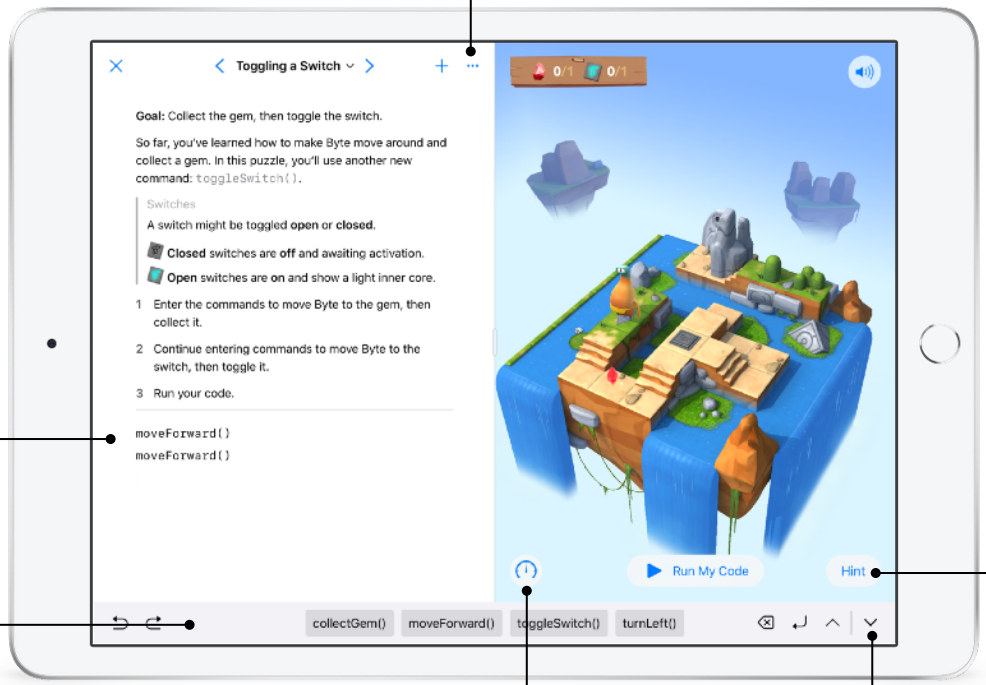
Real Swift code. At the heart of Swift Playgrounds is the same Swift programming language that's used to build many of the leading apps in the App Store today.

Built-in glossary. Definitions help students understand specific terms.

Record and share. Students can record what they do onscreen to demonstrate their work.

Immersive animations. Each section starts with an immersive animation that relates coding concepts to real life, aiding in student understanding.

Interactive environment. Create code on the left side of the screen and instantly see the results on the right, with just a tap.



Touch to edit. Drag complex structures that wrap other code, such as loops and function definitions, around existing code. Just touch the keyword (such as "for") and the drag controls appear onscreen.

Accessibility. Swift Playgrounds was designed with accessibility in mind. It takes advantage of the many powerful accessibility features of iPadOS, including Switch Control and VoiceOver, and even provides additional voice commentary on the actions of characters as students control them with code.

Shortcut Bar. QuickType suggestions for code appear at the bottom of the screen that let students enter the code they need by just tapping the Shortcut Bar.

Helpful hints. Students can get help along the way if they get stuck. In many cases, hints change dynamically as they enter code.

Review code. Run code faster or slower, or step through it to highlight the lines of code as they execute, making it easier for students to identify where errors might occur.

Onscreen keyboard. A keyboard designed for Swift provides quick access to the numbers and symbols most commonly used in Swift code.

Everyone Can Code Scope and Sequence

Everyone Can Code Puzzles

Designed for students in grades 4 and up, *Everyone Can Code Puzzles* is a guide to Swift Playgrounds that reinforces problem-solving strategies and critical-thinking skills through over 45 hours of flexible activities. Each chapter helps students build on what they already know, experiment with new coding concepts, apply their understanding, and creatively communicate how coding impacts their lives.
Download Everyone Can Code Puzzles >

Chapter 1: Commands (3 hours). Students learn about the importance of clear, precise commands. They organize code into sequences to achieve a goal, and investigate the use of commands in everyday digital technologies.

Chapter 2: Functions (3 hours). Students explore the power of functions by grouping commands into a definition they can use over and over. They code a function to choreograph a robot dance routine, and consider the types of functions that everyday digital technologies might use.

Chapter 3: For Loops (3.5 hours). Students recognize patterns in the world around them and in their code, and learn how to write more efficient code using loops.

Chapter 4: Variables (4 hours). Students learn about how computers store information and explore coding using variables to keep track of data and create flexible programs. They investigate how changing the value of variables changes the program output.

Chapter 5: Conditional Code (4 hours). Students explore how Boolean logic helps us make decisions in our everyday lives and in code. They practice writing conditional code to anticipate changing conditions.

Chapter 6: Types and Initialization (5 hours). Students explore how programmers use types to code more efficiently. They learn to describe types according to their methods and properties.

Chapter 7: Functions with Parameters (4 hours). Students investigate procedures that require additional information to ensure the desired outcome. They learn how to make their functions more flexible and powerful using parameters to provide additional detail.

Chapter 8: Logical Operators (6 hours). Students explore scenarios that require them to consider multiple factors before they make a decision. They learn how to use logical operators in their code to respond to multiple conditions.

Chapter 9: While Loops (4.5 hours). Students explore the power of a while loop for repeating a section of code multiple times until a condition is met.

Chapter 10: Arrays and Refactoring (5 hours). Students learn about storing data in arrays and explore how coding using arrays and refactoring can help simplify their code.

Summative Quiz Project. As they learn new skills, students apply them to build their own digital quiz project. They learn firsthand how digital projects are the result of many cycles of ideation, building, testing, and refinement. Across the lessons, they refactor their code to add new features to their quiz, respond to user testing, and improve the efficiency of their code.

Everyone Can Code Adventures

Everyone Can Code Adventures is designed for students in grades 4 and up after they've completed *Everyone Can Code Puzzles*. Approximately 45 hours of hands-on activities let students experiment with hardware features and event-driven programming to express their creative ideas through code. Each chapter covers more advanced content in the Swift Playgrounds app, including Sensor Arcade, Blu's Adventure, Sonic Workshop, and Augmented Reality. Students also learn about app development concepts and the app design process throughout the course as they build up to a final project in Swift Playgrounds.

Download Everyone Can Code Adventures >

Chapter 1: Objects in Views (6 hours). Students explore the use of coordinates to place objects into a view—short for viewable area. They practice coding with coordinate pairs, which are the intersection points between a vertical axis and a horizontal axis.

Chapter 2: Events and Handlers (5 hours). Students learn about the basics of event handling by examining touch events. They review functions with parameters as they add code to make their images and text interactive.

Chapter 3: Arrays (8 hours). To continue learning event functionality, students review and extend their understanding of arrays. Students learn new operators and methods they can use with arrays. They also review some array functionality, operators, and nested for loops.

Chapter 4: More Events and Handlers (5 hours). Students combine their understanding of events and handlers with their understanding of arrays. They learn what kinds of arguments can be passed into an event handler function. They examine the many kinds of input available to them through iPad device hardware.

Chapter 5: Functions as Arguments (8 hours). Students explore how to use functions as arguments within another function using closures. This extension of events supports them as they dive into Augmented Reality.

Chapter 6: Return Types and Outputs (3 hours). Students examine how to use a function to return a specific type. Until this point students have used functions to package procedures and now they can create a type that can be used in other parts of a program.

Chapter 7: Classes and Components (7 hours). Students investigate the basics of components and how to piece them together to create something new. Students will use design thinking and review types. Students will learn about classes to group and organize data in a program. They use these components and connect them together to create an app-like experience.

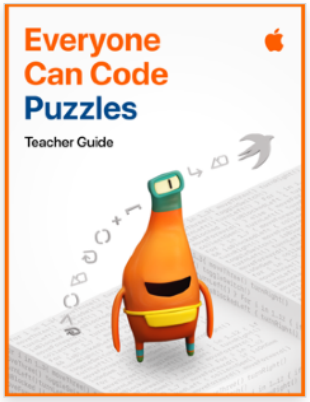
Everyone Can Code Support Materials

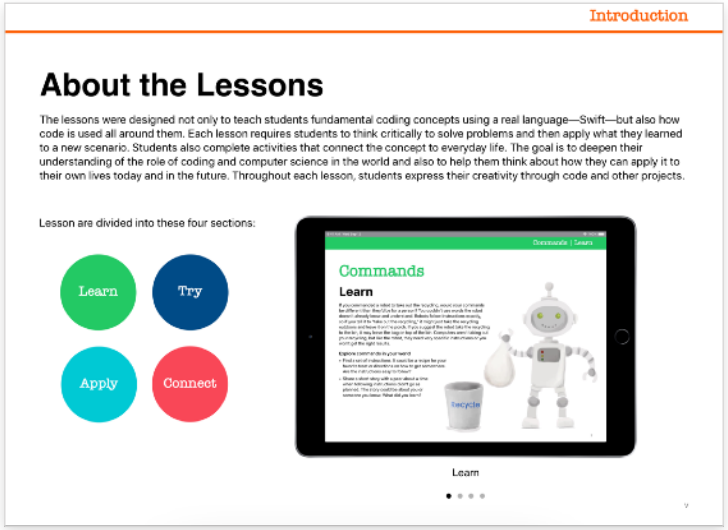
Teaching code isn't just about teaching the language of technology. It's also about teaching logic, creativity, problem-solving, collaboration, and communication—critical skills for the future. As students explore the lessons, they're learning key concepts in computer science, while also learning to appreciate the power of code for designing an equitable, inclusive future.

Everyone Can Code Teacher Guides

The Everyone Can Code curriculum offers teacher guides that support educators in confidently teaching code to deepen student learning:

- **Real-world engagement:** Each lesson uses real-world examples to show students how coding concepts can apply to everyday situations.
- **Communication and creation:** Activities encourage students to create drawings, presentations, music, and more to capture their thinking and showcase their work.
- **Teamwork:** The activities are designed with flexibility so students can complete them in small groups. In some, they create a shared product and use joint reasoning to achieve the best results.
- **Critical thinking:** Completing the activities requires students to analyze, interpret, synthesize, evaluate, and generate solutions.
- **Personalized learning:** The flexible materials are also designed to maximize student choice and address various skill levels. Helping language and useful analogies help teachers explain challenging concepts.





- **Assessment:** The teacher guides also provide assessment ideas for each section of the chapter. The ideas include tips on key behaviors to observe during class activities, indicators of understanding as evidenced in student explanations, and descriptions of how students can apply their understanding of coding concepts in creative projects.

App Design Journal

Students can use the App Design Journal to apply the app design process to solve problems in their school or community. The journal prompts students to brainstorm, plan, prototype, and evaluate their own app ideas, culminating in a pitch presentation of the app prototype. Download the App Design Journal >

App Showcase Guide

Encourage students to share their coding achievements with the broader community through community events, such as project demonstrations or app showcases. The App Showcase Guide provides practical support to help you plan and host a showcase event. Download the App Showcase Guide >

Inclusive Curriculum

At Apple, we believe that everyone should have the opportunity to create something that can change the world. For educators who teach students with disabilities, we've made accessible resources to extend coding to all students.

Accessible Guides

The Everyone Can Code student and teacher guides are optimized for VoiceOver and include closed-caption videos and audio descriptions. Each lesson has been designed to include accessible content, activities, and practice sessions to help bring coding to life for all learners.

Swift Playgrounds and VoiceOver Instructional Videos

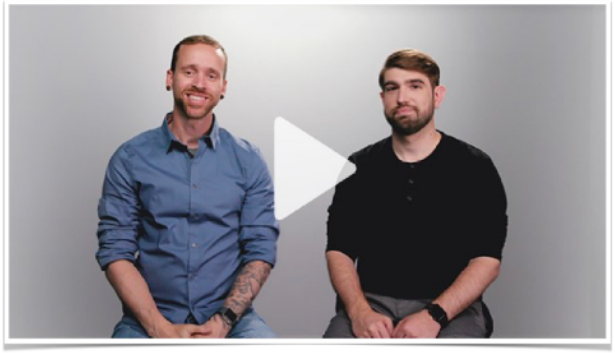
Teachers and students can watch instructional videos from the Hadley Institute for the Blind and Visually Impaired to help them get started using Swift Playgrounds with VoiceOver. Learn how to download Swift Playgrounds, orient to the puzzle worlds, use custom rotor actions, enter code, and solve a simple puzzle.
View the collection of videos here >

Swift Playgrounds Tactile Puzzle Worlds

Tactile Puzzle Worlds provide Unified English Braille, large-print text, and embossed graphics to help students who are blind or have low vision navigate the coding puzzles in Swift Playgrounds. Teachers can print the tactile graphics with their own braille embossers or order printed copies. Download Tactile Puzzle Worlds below for PDF versions of the tactile graphics and instructions on how to print or order copies for your students.
Download Tactile Puzzle Worlds >

Coding Concepts in American Sign Language

Accessible videos in American Sign Language support Deaf students in learning to code. With closed captions and text transcripts, they're designed for everyone to understand and enjoy. The Puzzles teacher guide includes a range of American Sign Language videos that explain coding concept.
View the collection of videos here >



Swift Coding Clubs

Swift Coding Clubs give everyone the opportunity to learn and write code. Designed for an informal setting, Swift Coding Clubs are a great way to introduce code to your school or provide opportunities for students to take their interests further.

The Swift Coding Club's modular design makes it perfect for first-time coders, as well as for those with more experience. Club activities built around Swift foster collaboration and creative problem solving. Each 12-session module includes a project-based activity where students use their skills to design, build, test, and share something that's uniquely theirs.

Club materials include cool Swift Coding Club posters, T-shirt designs, stickers, and certificates to help club leaders build interest and create a club identity. Clubs are encouraged to hold community events or app showcases to celebrate students' achievements and grow the coding community. Download the Swift Coding Club kit >

Additional Resources

Download the Everyone Can Code resources

- Everyone Can Code Puzzles
- Everyone Can Code Puzzles Teacher Guide
- Everyone Can Code Adventures
- Everyone Can Code Adventures Teacher Guide
- Swift Playgrounds app for iPad
- Swift Playgrounds app for Mac

Learn more about Develop in Swift resources

- Develop in Swift Curriculum Guide
- Teaching Code Collection on Apple Books

About Swift

Swift is the powerful and intuitive programming language created by Apple for building apps. Swift is not only great for getting you started with coding, it's also super powerful. It's designed to scale from writing the simplest program, like "Hello, world!", to the world's most advanced software. Learn more about Swift.

About Xcode

Xcode is the Mac app used to build every other Mac app and every iOS app, too. It has all the tools you need to create an amazing app experience. And it's available as a free download from the Mac App Store. Learn more about Xcode.

Apple Professional Learning

Apple Professional Learning Specialists help provide educators with personal support for innovating their instructional practices in ways that will engage students and enable personalized learning.

Apple Professional Learning Specialists organize multiple-day training engagements designed to provide:

- Leadership visioning and planning to help administrators prioritize learning objectives.
- Guidance from an Apple Professional Learning Specialist who's dedicated to your project.
- Customized, research-based professional learning plans to match learning goals.
- Hands-on, immersive learning experiences to help faculty develop innovative instructional practices that engage students.
- An opportunity to work with your professional learning leaders to build sustainability.

To learn more, contact your Apple account executive or send an email to AppleProfessionalLearning@apple.com.

Curriculum Alignment: Grades 3–5

Here's the preliminary alignment of *Everyone Can Code Puzzles* and *Everyone Can Code Puzzles Teacher Guide* with the Computer Science Teachers Association (CSTA) K–12 Computer Science Standards for Level 1B (Grades 3–5). The alignment covers the Algorithms and Programming and Impacts of Computing concepts within the CSTA Standards.

Alignment Puzzles: CSTA K–12 Computer Science Standards Level 1B (Grades 3–5)—Algorithms and Programming Concepts											
	CSTA Standard	**1B-AP-08** Compare algorithms	**1B-AP-9** Variables	**1B-AP-10** Sequences, events, loops, and conditionals	**1B-AP-11** Decomposition	**1B-AP-12** Remix programs	**1B-AP-13** Iterative development	**1B-AP-14** Attribution	**1B-AP-15** Test and debug	**1B-AP-16** Collaborative development	**1B-AP-17** Document programs
	Overall Alignment	●	●	●	●	●	●	●	●	●	●
Puzzles Chapters	**Commands**	●		●	●	●	●		●	●	●
	Functions	●		●	●	●	●	●	●	●	●
	For Loops	●		●	●	●	●		●	●	●
	Variables	●	●	●	●	●	●		●	●	●
	Conditional Code	●		●	●	●	●		●	●	●
	Types and Initialization	●	●	●	●	●	●		●	●	●
	Functions with Parameters	●	●	●	●	●	●	●	●	●	●
	Logical Operators	●	●	●	●	●	●		●	●	●
	While Loops	●	●	●	●	●	●		●	●	●
	Arrays and Refactoring	●	●	●	●	●	●		●	●	●

Key: ● Overall Alignment ● Aligns to Standard

Curriculum Alignment: Grades 3–5 (continued)

Alignment Puzzles: CSTA K–12 Computer Science Standards Level 1B (Grades 3–5)—Impacts of Computing Concept					
	CSTA Standard	**1B-IC-18** Technological change	**1B-IC-19** Accessibility	**1B-IC-20** Diverse perspectives	**1B-IC-21** Ethical use
	Overall Alignment	●	●	●	●
Puzzles Chapters	**Commands**		●	●	
	Functions	●	●	●	●
	For Loops		●	●	●
	Variables	●		●	●
	Conditional Code		●	●	
	Types and Initialization	●	●	●	
	Functions with Parameters				●
	Logical Operators			●	
	While Loops			●	
	Arrays and Refactoring		●	●	

Key: ● Overall Alignment ● Aligns to Standard

Curriculum Alignment: Grades 6–8

Here's the preliminary alignment of *Everyone Can Code Puzzles* and *Everyone Can Code Puzzles Teacher Guide* with the Computer Science Teachers Association (CSTA) K–12 Computer Science Standards for Level 2 (Grades 6–8). The alignment covers the Algorithms and Programming concepts and Impacts of Computing concepts within the CSTA Standards.

Alignment Puzzles: CSTA K–12 Computer Science Standards Level 2 (Grades 6–8)—Algorithms and Programming Concepts										
	CSTA Standard	**2-AP-10** Flowcharts Pseudocode	**2-AP-11** Variables	**2-AP-12** Control structures	**2-AP-13** Decomposition	**2-AP-14** Procedures with parameters	**2-AP-15** Feedback	**2-AP-16** Incorporate existing code	**2-AP-17** Test and refine	**2-AP-18** Collaborative development	**2-AP-19** Document programs
	Overall Alignment	●	●	●	●	●	●	●	●	●	●
Puzzles Chapters	**Commands**	●			●		●	●	●	●	●
	Functions	●			●	●	●	●	●	●	●
	For Loops	●		●	●	●	●	●	●	●	●
	Variables	●	●	●	●	●	●	●	●	●	●
	Conditional Code	●		●	●	●	●	●	●	●	●
	Types and Initialization	●	●	●	●	●	●	●	●	●	●
	Functions with Parameters	●	●	●	●	●	●	●	●	●	●
	Logical Operators	●	●	●	●	●	●	●	●	●	●
	While Loops	●	●	●	●	●	●	●	●	●	●
	Arrays and Refactoring	●	●	●	●	●	●	●	●	●	●

Key: ● Overall Alignment ● Aligns to Standard

Curriculum Alignment: Grades 6–8 (continued)

Alignment Puzzles: CSTA K–12 Computer Science Standards Level 2 (Grades 6–8)—Impacts of Computing Concept					
	CSTA Standard	**2-IC-20** Technological change	**2-IC-21** Accessibility	**2-IC-22** Collaborative surveys	**2-IC-23** Privacy
	Overall Alignment	●	●	●	●
Puzzles Chapters	**Commands**		●		
	Functions	●	●		
	For Loops		●		
	Variables	●			
	Conditional Code		●		
	Types and Initialization	●	●		●
	Functions with Parameters				
	Logical Operators		●	●	●
	While Loops				
	Arrays and Refactoring		●		

Key: ● Overall Alignment ● Aligns to Standard

AP is a registered trademark of the College Board and is used with permission. Features are subject to change. Some features may not be available in all regions or all languages.

© 2020 Apple Inc. All rights reserved. Apple, the Apple logo, iPad, Mac, QuickType, and Xcode are trademarks of Apple Inc., registered in the U.S. and other countries. iPadOS, Swift, the Swift logo, and Swift Playgrounds are trademarks of Apple Inc. App Store is a service mark of Apple Inc., registered in the U.S. and other countries. IOS is a trademark or registered trademark of Cisco in the U.S. and other countries and is used under license. Other product and company names mentioned herein may be trademarks of their respective companies. Product specifications are subject to change without notice. This material is provided for information purposes only; Apple assumes no liability related to its use. July 2020