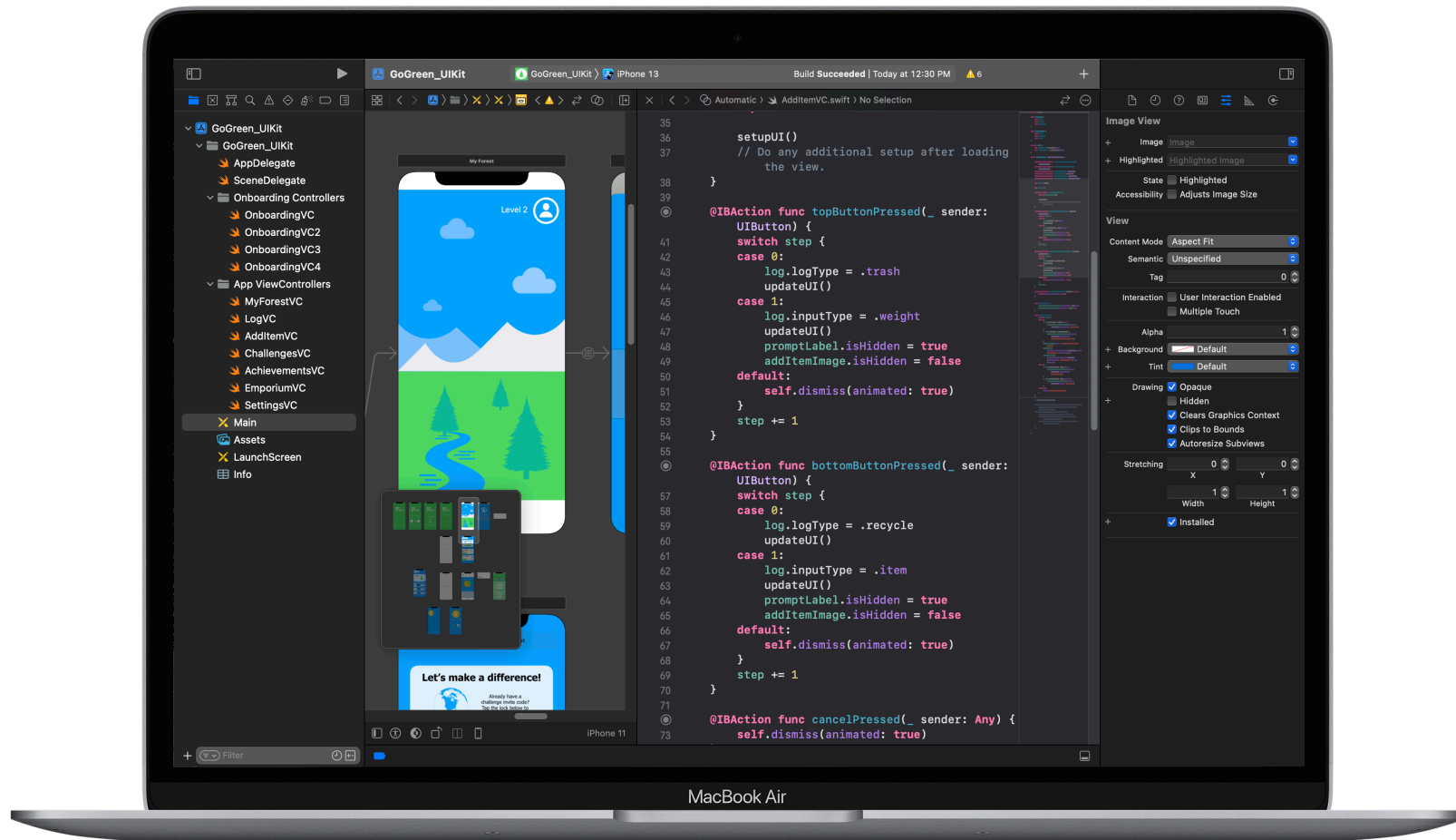


# Apple Develop in Swift

## Curriculum Guide



# Develop in Swift

Develop in Swift is a comprehensive coding offering intended for students in secondary school and above. The curriculum prepares students for college or a career in app development using the Swift programming language, and is complemented with free online professional learning for educators. Swift is designed for Mac—which supports all major programming languages—making it the ideal device for teaching and learning code.

As students move from Develop in Swift Explorations or AP<sup>®</sup> CS Principles to more advanced concepts in Fundamentals and Data Collections, they'll explore designing and building a fully functioning app of their own—and can even earn AP<sup>®</sup> credit or an industry-recognized certification. And for out-of-school coding, the App Design Workbook, App Showcase Guide, and Swift Coding Club help students design, prototype, and celebrate their app ideas.



# Secondary School Curriculum Pathway

## Explorations or AP<sup>®</sup> CS Principles

180 hours

Students learn key computing concepts, building a solid foundation in programming with Swift. They'll learn about the impact of computing and apps on society, economies, and cultures while exploring iOS app development. The AP<sup>®</sup> CS Principles course extends Develop in Swift Explorations to prepare students for the AP<sup>®</sup> Computer Science Principles exam.

**Unit 1:** Values

**Episode 1:** The TV Club

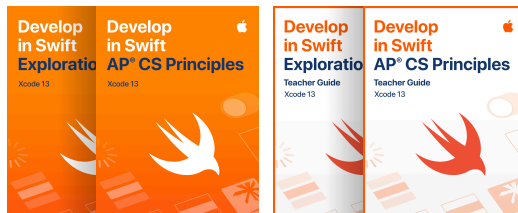
**Unit 2:** Algorithms

**Episode 2:** The Viewing Party

**Unit 3:** Organizing Data

**Episode 3:** Sharing Photos

**Unit 4:** Building Apps



## Fundamentals

180 hours

Students build fundamental iOS app development skills with Swift. They'll master the core concepts and practices that Swift programmers use daily, and build a basic fluency in Xcode source and UI editors. Students will be able to create iOS apps that adhere to standard practices, including the use of stock UI elements, layout techniques, and common navigation interfaces.

**Unit 1:** Getting Started with App Development

**Unit 2:** Introduction to UIKit

**Unit 3:** Navigation and Workflows



## Data Collections

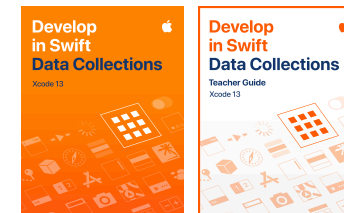
180 hours

Students expand on the knowledge and skills they developed in Fundamentals by extending their work in iOS app development, creating more complex and capable apps. They'll work with data from a server and explore new iOS APIs that allow for much richer app experiences—including displaying large collections of data in multiple formats.

**Unit 1:** Tables and Persistence

**Unit 2:** Working with the Web

**Unit 3:** Advanced Data Display



# Higher Education Curriculum Pathway



Students learn key computing concepts, building a solid foundation in programming with Swift. They'll learn about the impact of computing and apps on society, economies, and cultures while exploring iOS app development.

- Unit 1:** Values
  - Episode 1:** The TV Club
  - Unit 2:** Algorithms
    - Episode 2:** The Viewing Party
  - Unit 3:** Organizing Data
    - Episode 3:** Sharing Photos
  - Unit 4:** Building Apps



Students build fundamental iOS app development skills with Swift. They'll master the core concepts and practices that Swift programmers use daily, and build a basic fluency in Xcode source and UI editors. Students will be able to create iOS apps that adhere to standard practices, including the use of stock UI elements, layout techniques, and common navigation interfaces.

- Unit 1:** Getting Started with App Development
- Unit 2:** Introduction to UIKit
- Unit 3:** Navigation and Workflows



Students expand on the knowledge and skills they developed in Fundamentals by extending their work in iOS app development, creating more complex and capable apps. They'll work with data from a server and explore new iOS APIs that allow for much richer app experiences—including displaying large collections of data in multiple formats.

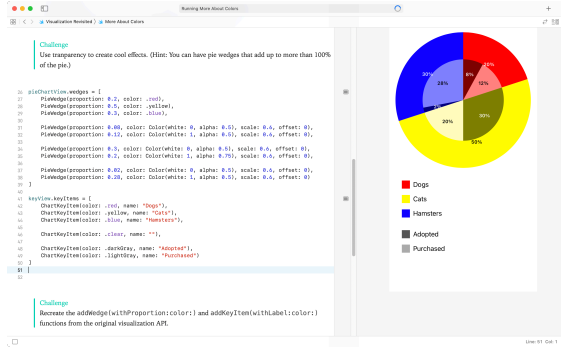
- Unit 1:** Tables and Persistence
- Unit 2:** Working with the Web
- Unit 3:** Advanced Data Display



# Key Features

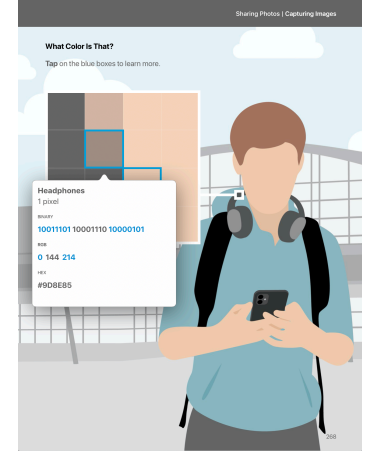
## Xcode playgrounds

Students learn programming concepts as they write code in playgrounds—interactive coding environments that let them experiment with code and see results immediately.



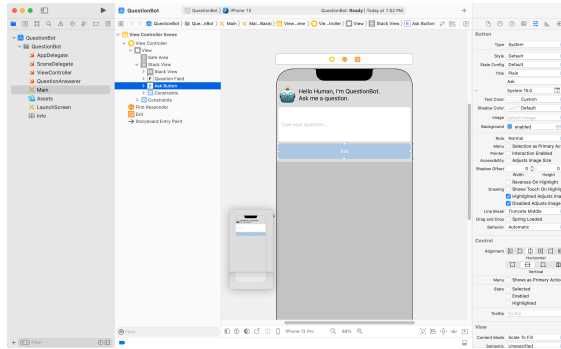
## Connected World episodes\*

Illustrated Connected World episodes allow students to explore everyday activities and tools—from searching on the web and taking photos to interacting on social media—while exploring the technology behind them and their impact on society.



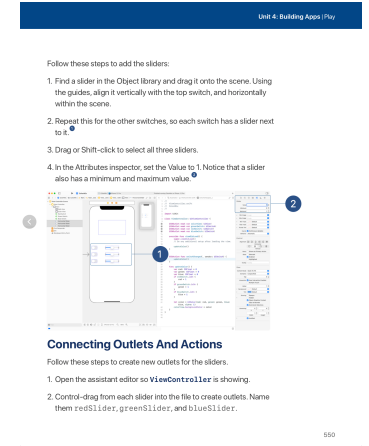
## Guided app projects

Using the included project files, students can work with key concepts without having to build an app from scratch. Supporting images and videos challenge them to apply their knowledge.



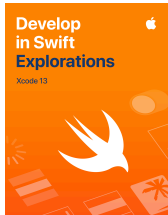
## Step-by-step instructions

Detailed instructions with images and videos guide students through all the steps of building an app in Xcode.

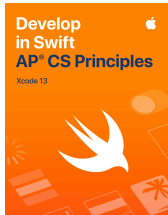


\*Available in Develop in Swift AP® CS Principles and Develop in Swift Explorations courses only.

# Develop in Swift Explorations and AP<sup>®</sup> CS Principles



Apple’s app development curriculum starts with the Develop in Swift Explorations and AP CS Principles books to help students learn key computing concepts, building a solid foundation in programming with Swift. They’ll learn about the impact of computing and apps on society, economies, and cultures while exploring iOS app development. Lessons take students through the app design process: brainstorming, planning, prototyping, and evaluating an app design of their own. While they may still be developing skills to convert prototypes into full apps, designing an app is a critical skill and encourages students to learn to code.



As a College Board–endorsed provider for the 2021–2022 school year, Apple expanded the Explorations course to create AP<sup>®</sup> CS Principles, including material to prepare students for the AP<sup>®</sup> Computer Science Principles exam.

Download: [apple.co/developinswiftexplorations](https://apple.co/developinswiftexplorations)

Download: [apple.co/developinswiftapcsp](https://apple.co/developinswiftapcsp)

**Unit 1: Values.** Students learn about the fundamental units of Swift—the values that flow through their code, including text and numbers. They explore how to associate names with values using variables. The unit culminates in an app project to display a photo.

**Episode 1: The TV Club.** Students follow members of a TV club as they anticipate the new season of their favorite show. They learn how searching on the web and signing up for accounts relates to their personal information, as well as how to think about their privacy while using apps.

**Unit 2: Algorithms.** Students learn how to structure their code using functions to encapsulate repetitive tasks, use if/else statements to represent decisions, and explore how Swift uses types to distinguish different kinds of data. The culminating project is a QuestionBot app that responds to user input from the keyboard.

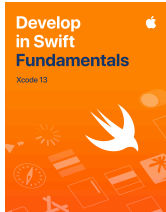
**Episode 2: The Viewing Party.** The TV club story continues as its members stream the episode while texting each other. Students explore how data is represented inside their devices at the lowest level and how it flows across the internet. They also learn more about security and privacy of data.

**Unit 3: Organizing Data.** Students explore how to create custom types using structs, group large quantities of items into arrays, and process them using loops. They also learn how enums represent a set of related values, and in the app project at the end of the unit, they build an interactive game with colorful shapes.

**Episode 3: Sharing Photos.** The TV club concludes as its members share pictures of the viewing party on social media. Students learn about digitizing analog data and parallel computing, and they explore some consequences of sharing data online.

**Unit 4: Building Apps.** Students deepen their skills in Xcode and Interface Builder in guided projects to build apps from the ground up. They learn how to add user interface elements to a screen, connect those elements to their code, and respond to the events generated by user interaction. They use the incremental development process to build their apps one piece at a time, testing as they go. The culmination of the unit is a study app with flash card and quiz modes.

# Develop in Swift Fundamentals

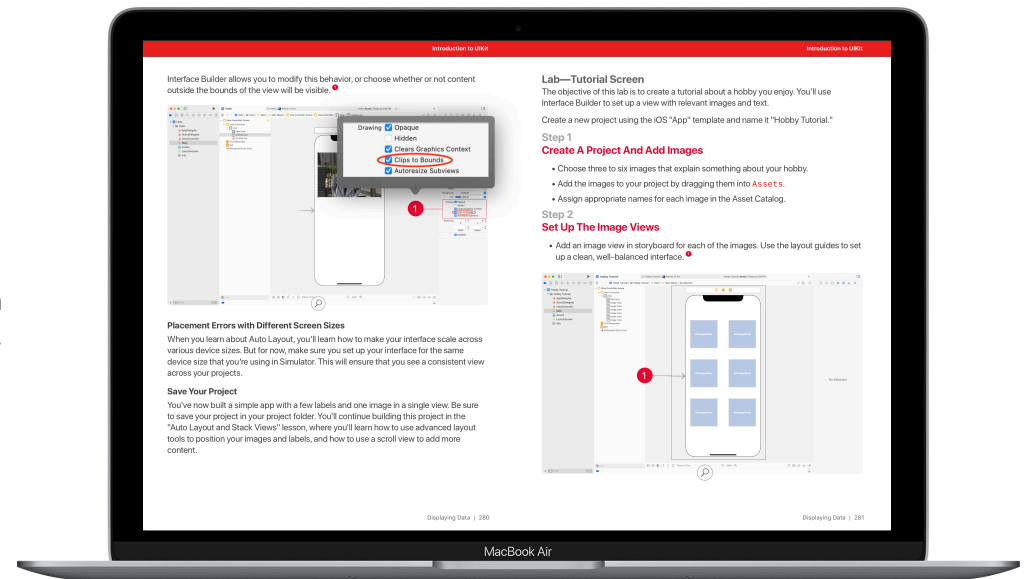


In this course, students build fundamental iOS app development skills. They'll master the core concepts and practices of the Swift programming language that developers use daily and build a basic fluency in Xcode's source and UI editors. Students will dive into the UIKit framework to create iOS apps that adhere to standard practices, including use of stock UI elements and layout techniques, and common navigation interfaces. Three guided app projects help students build an app in Xcode from the ground up with step-by-step instructions. Xcode playgrounds help students learn key programming concepts in an interactive coding environment that lets them experiment with code and see the results immediately. They'll also embark on an app design journey, using the App Design Workbook to define, prototype, and test their own app idea. Download: [apple.co/developinswiftfundamentals](https://apple.co/developinswiftfundamentals)

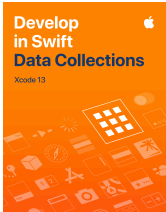
**Unit 1: Getting Started with App Development.** Students find out about the basics of data, operators, and control flow in Swift, as well as documentation, debugging, Xcode, building and running an app, and Interface Builder. They then apply this knowledge to a guided project called Light in which they create a simple flashlight app.

**Unit 2: Introduction to UIKit.** Students explore Swift strings, functions, structures, collections, and loops. They also learn about UIKit—the system views and controls that make up a user interface—and how to display data using Auto Layout and stack views. They put this knowledge to practice in a guided project called Apple Pie, where they build a word-guessing game app.

**Unit 3: Navigation and Workflows.** Students discover how to build simple workflows and navigation hierarchies using navigation controllers, tab bar controllers, and segues. They also examine two powerful tools in Swift: optionals and enumerations. They put this knowledge into practice with a guided project called Personality Quiz—a personalized survey that reveals a fun response to the user.



# Develop in Swift Data Collections

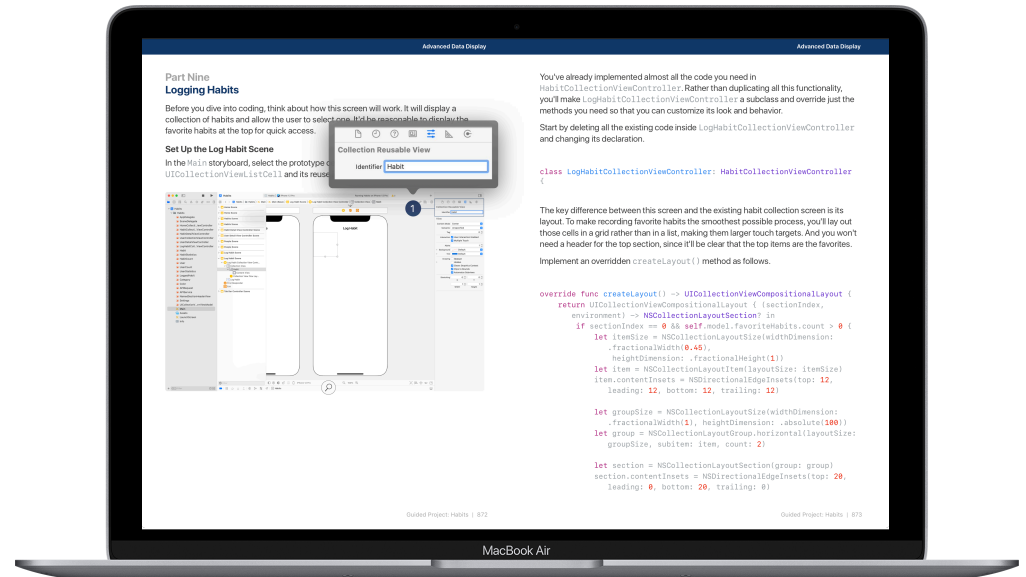


Students expand on the knowledge and skills they developed in *Develop in Swift Fundamentals* by extending their work in iOS app development, creating more complex and capable apps. They'll work with data from a server and explore new iOS APIs that allow for much richer app experiences—including displaying large collections of data in multiple formats. Three guided app projects help students build an app in Xcode from the ground up with step-by-step instructions. Xcode playgrounds help students learn key programming concepts in an interactive coding environment that lets them experiment with code and see the results immediately. Download: [apple.co/developinswiftdatacollections](https://apple.co/developinswiftdatacollections)

**Unit 1: Tables and Persistence.** Students learn scroll views, table views, and building complex input screens. They also explore how to save data, share data to other apps, and work with images in a user's photo library. They use their new skills in a guided project called *List*, a task-tracking app that allows the user to add, edit, and delete items in a familiar table-based interface.

**Unit 2: Working with the Web.** Students learn about animations, concurrency, and working with the web. They apply what they've learned in a guided project called *Restaurant*—a customizable menu app that displays a restaurant's available dishes and allows the user to submit an order. The app uses a web service that lets students set up the menu with their own menu items and photos.

**Unit 3: Advanced Data Display.** Students learn how to use collection views to display data in a highly customizable, two-dimensional layout. They also discover the power of Swift generics and bring all their skills together in an app that manages a complex data set and presents a customizable interface.





# Teaching Code with Apple

When you teach code, you're not only teaching the language of technology. You're also teaching new ways to think and bring ideas to life. And Apple has free resources to help you bring code into your classroom, whether you're just starting or ready to get your students certified in Swift. The [Everyone Can Code](#) curriculum introduces students to coding through a world of interactive puzzles and playful characters with the Swift Playgrounds app. The [Develop in Swift](#) curriculum introduces students to the world of app development by making it easy for them to design and build a fully functioning app of their own design. And Apple supports educators with professional learning offerings to help you get started bringing Everyone Can Code and Develop in Swift offerings to students.

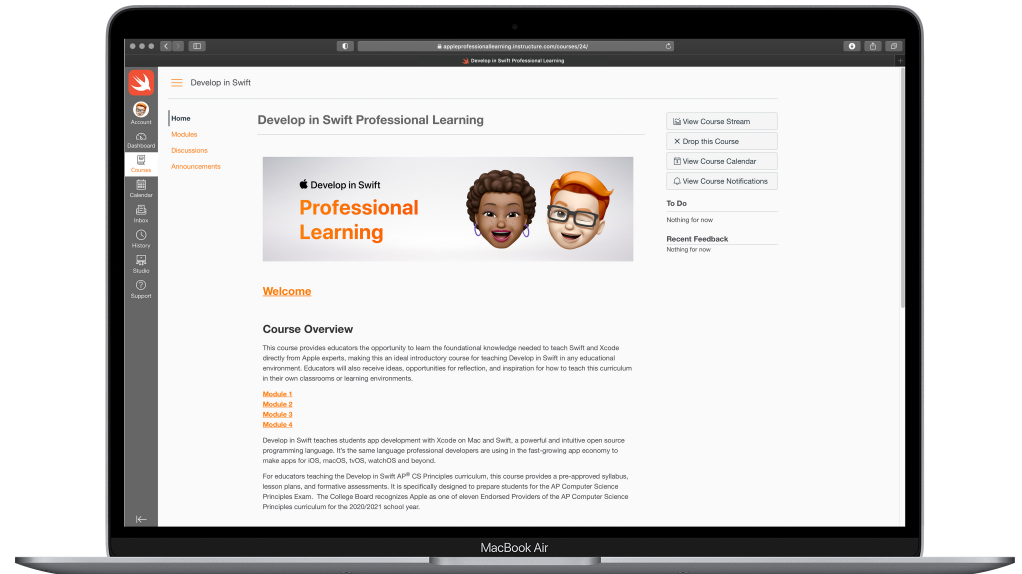
## Free Self-Paced Online Professional Learning

The Develop in Swift Explorations and AP® CS Principles course is available through Canvas by Instructure. Participants will learn the foundational knowledge they need to teach Swift and Xcode directly from Apple education experts, making this the ideal introductory course for teaching Develop in Swift in any educational environment. Learn more at [apple.co/developinswiftexplorationspl](https://apple.co/developinswiftexplorationspl).

## Bring an Apple Professional Learning Specialist to your school

For educators interested in going further, Apple Professional Learning Specialists organize multiple-day training engagements designed to provide hands-on, immersive learning experiences to help faculty develop innovative instructional practices that engage students.

To learn more about Apple Professional Learning, email [sea\\_aels@apple.com](mailto:sea_aels@apple.com).



# App Development with Swift Certifications

Educators who are teaching app development with Swift can help their students prepare for a career in the app economy by earning an industry-recognized certification. App Development with Swift certifications recognize foundational knowledge of Swift, Xcode, and app development tools covered by the free Develop in Swift Explorations and Develop in Swift Fundamentals courses. After successfully completing an App Development with Swift exam students will earn a digital badge they can add to a resume, portfolio, email, or share with professional and social media networks. Learn more: [certiport.com/apple](https://certiport.com/apple)



## APP DEVELOPMENT WITH SWIFT Associate

### App Development with Swift Associate

High school or higher education students who successfully complete the App Development with Swift Associate exam demonstrate knowledge of the impact of computing and apps on society, economies, and cultures while exploring iOS app development. This certification is aligned with Develop in Swift Explorations course.



## APP DEVELOPMENT WITH SWIFT Certified User

### App Development with Swift Certified User

Higher education students who successfully complete the App Development with Swift Certified User exam demonstrate fundamental iOS app development skills with Swift. They have knowledge of core concepts and practices that professional Swift programmers use daily. This certification is aligned with Develop in Swift Fundamentals course.

# Additional Resources



## App Design Workbook

The App Design Workbook uses a design thinking framework to teach students app design—a fundamental skill of iOS app development. They’ll explore the relationship between app design and coding in Swift through each stage of the app design cycle to bring their app ideas to life. Download: [apple.co/developinswiftappdesignworkbook\\_SG](https://apple.co/developinswiftappdesignworkbook_SG)



## App Showcase Guide

Celebrate student ingenuity by encouraging students to share their coding achievements with community events, such as project demonstration events or app showcases. The App Showcase Guide provides practical support to help you host an in-person or virtual app showcase event. Download: [apple.co/developinswiftappshowcaseguide\\_SG](https://apple.co/developinswiftappshowcaseguide_SG)

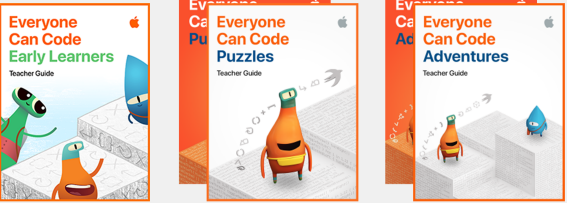










## Swift Coding Club

Swift Coding Clubs are a fun way to design apps. Activities are built on learning Swift programming concepts in Xcode playgrounds on Mac. Students collaborate with peers to prototype apps and think about how code can make a difference in the world around them. Download: [apple.co/swiftcodingclubxcode\\_SG](https://apple.co/swiftcodingclubxcode_SG)

# K–20 Coding Resources

Whether students are beginning coders or are ready to build their first apps, Apple has programs to support teaching and learning with Swift, the same programming language used by professional developers to build some of the world's most powerful apps. Everyone Can Code introduces students to the world of coding and app design through interactive puzzles, playful characters, and activities for both in and out of the classroom using the Swift Playgrounds app. Develop in Swift provides students with practical experience designing and developing apps with Swift in Xcode to prepare them for college, career, and even industry-recognized certification in app development.

PRESCHOOL	PRIMARY SCHOOL	SECONDARY SCHOOL	HIGHER EDUCATION
<h2>Apple Everyone Can Code</h2>		<h2>Apple Develop in Swift</h2>	
			
<p>16 hours      45 hours      45 hours iPad only</p>		<p>180 hours      180 hours      180 hours      180 hours One semester      One semester      One semester      One semester</p>	
			
<p>Swift Playgrounds on iPad and Mac   </p>		<p>Xcode on Mac  </p>	



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. © 2021 Apple Inc. All rights reserved. Apple, the Apple logo, Mac, MacBook Air, Swift, the Swift Logo, Swift Playgrounds, and Xcode are trademarks of Apple Inc., registered in the U.S. and other countries. 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. December 2021