

App Development with Swift

Associate

Objective Domains

Individuals who earn the App Development with Swift Associate certification exemplify knowledge of key computing concepts and a solid foundation in programming with Swift. They'll demonstrate knowledge of the impact of computing and apps on society, economies, and cultures while exploring iOS app development.

Planning, Design and Theory

- 11 Summarize the design cycle
 - 11.1 Brainstorm, plan, prototype, evaluate
- 12 Summarize how sensitive data can be protected and compromised
 - 12.1 Sharing personal and application information
 - 12.2 Security challenges
 - 12.3 Legal, ethical and socioeconomic impacts

Project Navigation

- 21 Differentiate between basic file types
- 22 Recognize the assets available in a project
- 23 Define how assets are used
- 24 Import an asset to a project and use it correctly
- 25 Select the appropriate actions to hide or show different areas of the user interface

Interface Builder/iOS

- 31 Given a scenario, select the appropriate object(s) on the storyboard or the Document Outline
- 32 Use the Attributes inspector to non-programmatically modify the properties of objects and/or a view
- 33 Connect UIKit objects on storyboard to a Swift file
 - 33.1 Differentiate between an IBOutlet and an IBAction
 - 33.2 Determine when to connect an object as an IBOutlet or an IBAction
- 34 Programmatically modify the properties of objects and/or a view



APP DEVELOPMENT
WITH SWIFT

App Development with Swift

Associate

App Development with Swift

Associate

Swift Language Usage

- 41 Write, call and/or evaluate the execution of functions
 - 41.1 Evaluate the use of argument labels, parameters and returns
- 42 Calculate the results when using various operators
- 43 Create and evaluate structures
 - 43.1 Declare the properties of a structure
 - 43.2 Initialize the properties of a structure
 - 43.3 Define methods
 - 43.4 Create an instance of a structure
 - 43.5 Use an instance of a structure
- 44 Create and manipulate arrays
 - 44.1 Declare and/or initialize an array with values
 - 44.2 Identify and/or modify an array element using its index
 - 44.3 Use and/or evaluate array properties and/or methods
- 45 Demonstrate how to control the flow of execution
 - 45.1 Create, analyze and predict loop structures and their results
 - 45.2 Create and interpret the outcome of conditional statements

- 46 Create, use and/or compare custom enumerations
- 47 Declare and/or evaluate constants and variables of different data types
 - 47.1 Differentiate between constants and variables
 - 47.2 Apply type inference
 - 47.3 Use explicit typing
- 48 Use the appropriate naming conventions
 - 48.1 Use appropriate camel casing
 - 48.2 Apply Swift identifier rules

Debugging

- 51 Use the Connections inspector to evaluate whether a connection error has occurred
- 52 Given a connection error scenario, determine a solution
- 53 Differentiate between syntax and run-time errors when building and running an app
- 54 Interpret console error messages
- 55 Recognize the purpose of breakpoints

