

VERSION

2

PROFESSIONAL

ELYSIUM
ACADEMY
MOBILE APP
DEVELOPER
IOS
**ELYSIUM
ACADEMY
MOBILE APP
DEVELOPER
IOS**

SR. CODE

EAPL/PROF/PRTC12

COURSE CODE

EAPIO

SUB CATEGORY

MOBILE APP DEVELOPMENT



TOTAL DURATION
90
HOURS



THEORY TAKEN
40
HOURS



PRACTICAL TAKEN
50
HOURS

ELYSIUM
ACADEMY
MOBILE APP
DEVELOPER
IOS

COURSE DESCRIPTION



This course begins with the most fundamental matters, introduces you to the iOS platform and the Swift programming language, and then gradually begins to deal with more advanced topics such as cloud servers, concurrency, device hardware, networking, or debugging.

COURSE GOALS



One of the biggest advantages to getting started with iOS development is the ease of learning Swift, the official programming language for iOS apps. It was introduced by Apple in 2014 to replace Objective-C, which had been the standard up until then, and was quickly adopted by most iOS developers.

FUTURE SCOPE



Mobile app development has a bright future and a lot of possibilities. Developers now have the resources necessary to produce really new and transformational mobile apps that have the potential to alter the way we live and work thanks to the rise of AI and ML, cross-platform development, progressive web apps, the Internet of Things, and 5G. iOS is the most feature-rich mobile operating system after Android. With the popularity of iPhones constantly growing, its market share will reach 27.73% in 2022.

01

CHAPTER

GETTING STARTED

O1. About Tools

- a. XCode
- b. iOS
- c. Swift
- d. Installation setup

O2. About Swift

- a. Comments
- b. The println() function
- c. Variables
- d. Constants
- e. Data types
- f. Optional
- g. Type inference
- h. Hands on

O3. Control Flow

- a. For
- b. If else if
- c. Do While
- d. While
- e. Switch
- f. Break and continue statement
- g. Hands on control flow



04
HRS



05
HRS

O4. Operators

- a. Comparison operators
- b. Arithmetic operators
- c. Logical operators
- d. Hands on operators

O5. Strings

- a. Literals
- b. Mutable strings
- c. Comparing strings
- d. Concatenating strings
- e. Hands on strings

02

CHAPTER

FUNCTIONS, OOPS

O1. Functions

- a. What is Swift Functions?
- b. Swift Nested Function
- c. Parameter & Return Value
- d. Swift Recursion
- e. Function Overloading
- f. Hands on functions

O2. Swift collections

- a. Swift Arrays
- b. Swift Dictionary
- c. Swift Sets
- d. Swift Closures



02
HRS



08
HRS

- e. Swift Typealias
- f. Hands on collections

O3. Loops

- a. For loops
- b. While loops
- c. For in loops
- d. Iterating over arrays
- e. Hands on loops

O4. OOPS

- a. Swift Enum
- b. Structures
- c. Classes and objects
- d. Inheritance
- e. Methods
- f. Class Methods
- g. Type check
- h. Any object
- i. Protocols
- j. Hands on Swift OOPS

O5. Closures, Protocol & Delegates

- a. Defining a closure
- b. Closures with parameters
- c. Closures with returned values
- d. Closures as callbacks
- e. Defining & implementing protocol
- f. Delegate design pattern

- g. Implementing & using delegates
- h. Hands on closures, protocol, delegates

03

CHAPTER

ADVANCED OOPS

O1. ENUM and Type Casting

- a. Creating & using enumerations
- b. Type checking
- c. Type casting
- d. Down casting
- e. Hands on enum and type casting

O2. Tuples, ARC

- a. Creating & using tuples
- b. Creating & using type aliases
- c. Strong & weak references
- d. Avoiding strong reference cycles



02
HRS



05
HRS

O3. Memory Management

- a. Reference Counting Basics
- b. Automatic Reference Count
- c. Retain Cycles

O4. Error Handling

- a. Create enum of Errors
- b. Create a Throwing Function
- c. Function Call Using try Keyword
- d. Handling Errors Using do-catch Statement
- e. Hands on

04

CHAPTER

EXTENSION, ACCESS CONTROL

O1. Swift Generic

- a. Swift Generic Function
- b. Swift Generic Class
- c. Type Constraints in Swift Generics
- d. Hands on

O2. Extension

- a. What is extension in swift?
- b. Computed Property In Extension
- c. Protocol Extension
- d. Hands on



O3. Swift Access Control

- a. What is Swift Access Control?
- b. Types of Swift Access Controls
- c. Public Access Control
- d. Private Access Control
- e. file private Access Control
- f. Internal Access Control

O4. Type alias and Hashable

- a. What is Swift Typealias?
- b. How to create a typealias?
- c. Typealias for built-in types
- d. Typealias for user defined types
- e. Typealias for complex types
- f. Swift hashable
- g. Hands on

O5. Swift Equatable

- a. What is equatable?
- b. Strong Reference in Swift
- c. Swift Weak Reference
- d. Hands on

05

CHAPTER

XCODE

01. XCode and Interface builder

- a. Purpose of XCode
- b. IOS Simulator
- c. Interface builder
- d. IOS application architecture
- e. Installation setup

02. COCOA Design Patterns

- a. What is MVC?
- b. Model, View and Controller Classes
- c. Delegate and Data source
- d. Singleton Pattern
- e. Observer pattern
- f. Target-Action
- g. Cocoa coding standards

03. Controls in IOS

- a. What is controls?
- b. Views
- c. Views Hierarchy

04. Common Interactions with UI

- a. Button, label, Text fields
- b. switch, slider, progress bar
- c. Alerts, Action sheet
- d. Table views
- e. Scroll view, Web view, maps



03
HRS



07
HRS

- f. Search Bar, popovers
- g. Picker, Date picker, Image view, Image picker controller
- h. Gestures, Mouse events
- i. Mail, Message, Phone call
- j. Creating outlets and actions
- k. Hands on Basic UI Interactions

O5. Auto Layout

- a. Stacks
- b. Nested Stacks
- c. Down casting
- d. Constraints
- e. Content Hugging Priority
- f. Compression Resistance Priority
- g. Hands on Auto layout

06

CHAPTER

VIEWS, ANIMATIONS

O1. Table View

- a. UITableViewController
- b. Working with multiple TableViews
- c. UITableViewCell
- d. Table View practices
- e. Custom Cell creation.
- f. Multi-View Applications
- g. view to view
(Present model view controller)



02
HRS



10
HRS

- h. Navigation controller
- i. Tab bar controller
- j. Page view controller
- k. Split view controller
- l. Hands on Table View

O2. Create TO DO APP

- a. Design UI
- b. Item Class
- c. Table View Items
- d. Adding Items Statically
- e. Table Header View
- f. Text Field Delegate Method
- g. Select an Item
- h. Break
- i. Deleting an Item
- j. No Duplicates
- k. Refactoring the Code
- l. Sections in Table View
- m. Animating the Transfer

O3. Touches

- a. Touches Began and Ended
- b. Touches Moved
- c. Gestures
- d. Hands on Gestures
- e. Transformations
- f. Gestures and Transform
- g. Hands on touches

O4. Animations

- a. Animation Blocks
- b. View Effects
- c. Hands on View Effects
- d. View Transitions
- e. Hands on animations

O5. Maps and locations

- a. Maps
- b. Annotations
- c. Custom Annotations
- d. Location
- e. Location on the map
- f. Hands on Maps and location

07

CHAPTER

STORYBOARDS, CONCURRENCY

O1. Storyboards

- a. Storyboard File
- b. View Controller and Scene
- c. Segue
- d. Invoking a Segue
- e. XIB and Storyboards
- f. Table View Cell Prototype


02
HRS


08
HRS

O2. Testing and Debugging

- a. Debugging with Print
- b. Breakpoints
- c. Exception Breakpoints
- d. Advanced Breakpoints
- e. Debug Navigator
- f. Unit Testing
- g. UI Testing
- h. Performance Testing
- i. Hands on testing and debugging

O3. Media and Devices

- a. Camera
- b. Play an Audio
- c. Audio Recorder
- d. Play a Video
- e. AV Player Stand Alone
- f. Orientation
- g. Motion
- h. XCode Assets
- i. Hands on

O4. Concurrency

- a. What is GCD?
- b. What is KVO?
- c. GCD Singleton
- d. Operation Queue
- e. Simple Operation

- f. Queued Operations - State and Value
- g. Queued Operations - Table View
- h. Queued Operations - Pending Operations
- i. Queued Operations - Table View Interactions
- j. Hands on

08

CHAPTER

WEB SERVICES

O1. Web Services

- a. What are web services?
- b. Why do we use web-services
- c. How apps function with the help of web-services
- d. Introduction to different web services
- e. JSON parsing, XML parsing
- f. Get and POST methods

O2. Push Notifications

- a. Notification
- b. NSNotification
- c. NSNotificationCenter
- d. UILocalNotifications
- e. Push Notification services



03
HRS



06
HRS

09

CHAPTER

FIREBASE

O1. About Firebase

- a. What is firebase?
- b. Purpose of firebase
- c. Features of firebase
- d. Adding firebase to App
- e. Installation setup

O2. Firebase Integration and User Authentication

- a. Setup Firebase
- b. Setup Authentication View Model
- c. Register Test User With Firebase
- d. Authentication View Model Bug Fix
- e. Keeping User Logged In
- f. Logging In / Signing Out

O3. Store the Data To Firestore

- a. Setup Firebase Firestore
- b. Setup Firebase Storage
- c. Upload User Data To Cloud Firestore
- d. Upload Profile Photo UI
- e. Upload Profile Photo Presentation Logic
- f. Upload Profile Photo To Firebase Storage



03
HRS



10
HRS

O4. Fetching and Display User Data from Firestore

- a. Fetch User Data From Firestore
- b. User Data Model
- c. Populating Settings View With User
- d. Load Profile Image & Authentication Bug Fix
- e. Fetch & Display Users In New Message View

Placement Assistance

100%

135+ Professional Courses

Practical Sessions

90%

67+ Global Pacts

Corporate Placements

65%

170+ IT Companies Tie-Up

ELYSIUM
GROUP OF
COMPANIES

**ELYSIUM
ACADEMY**

**PRIVATE
LIMITED**

AUTHORIZED INTERNATIONAL

Partners

