

Curriculum for IOS Swift

Introducing Swift:

- About Swift
- XCode Overview
- Projects and Playgrounds
- Basic Swift Syntax

Data Types:

- Primitives
- Integers, Floats, Booleans, Tuples
- Working with strings
- Constants and Variables
- Literals
- Type Aliases
- Optional

Basic Syntax:

- Math and Logic operations
- Defining Arrays and Collections
- Controlling Program Flow with conditionals and loops
- Enumerations
- Nested Types

Functions and Closures:

- Defining a Function
- Passing parameters
- Returning single and multiple return values
- Parameter Naming
- Variadic, constant, variable and In-Out parameters
- Closures

Classes and Structures:

- Value types explained
- Defining structures
- Assigning structure instances
- Reference Types
- Defining Classes
- Differences between Value and reference types and when to use
- Setting up Initializers and de-initializers

Properties in detail:

- Stored Properties
- Lazy and Computed properties

- Working with Property Observers
- Encapsulation?

Methods:

- Defining Instance and Type Methods
- Method naming
- Mutating Methods
- Subscripts

Using Inheritance and Polymorphism:

- Creating Subclasses
- Overriding Methods, Properties, and Observers
- Initialization of inherited types - Chaining
- Casting and subtype polymorphism

Automatic Reference Counting:

- ARC Under the hood
- Strong and Weak references
- Unowned references
- ARC and Closures

Protocols and Extensions:

- What is a protocol
- Implementing a protocol
- Delegation
- Protocol Inheritance
- Checking conformance
- Extending types with Extensions

Generics:

- Why Generics
- Generic Functions
- Generic Types
- Constraints

Getting Started:

- Registering as an Apple developer
- Apple iOS Developer Program

Environment Setup:

- Interface Builder
- iOS simulator

First iPhone Application:

- Digging deep into the code of the First iOS application

Actions And Outlets:

- Steps Involved

UI Elements:

- How to add UI elements
- Our Focus
- Our Approach
- List of UI elements
- Use of text field
- Important properties of text field are
- Updating properties in xib
- Text field delegates
- Steps in using delegates
- Sample code and steps
- Why Input types?
- Keyboard Input types
- Add a custom method addTextFieldWithDifferentKeyboard
- Update viewDidLoad in ViewController.m as follows
- Output
- Buttons
- Label
- Tool bar
- Status Bar
- Navigation Bar
- Tab bar
- Image View
- Scroll View
- Table View
- Split View
- Text View
- View Transition
- Pickers
- Switches
- Sliders
- Alerts
- Icons

Accelerometer:

- Introduction
- Steps Involved
- Output

Universal Applications:

- Introduction
- Steps Involved
- Output

Camera Management:

- Introduction
- Steps Involved
- Output

Location Handling:

- Introduction
- Steps Involved
- Output

SQLite Database:

- Introduction
- Steps Involved
- Output

Sending Email:

- Introduction
- Steps Involved
- Output

Audio And Video:

- Introduction
- Steps Involved
- Note
- Output

File Handling:

- Methods used in File Handling
- Check if file exists at a path
- Comparing two file contents
- Check if writable, readable and executable
- Move file
- Copy file
- Remove file
- Read file
- Write file
- What next?

Accessing Maps:

- Introduction
- Output

In-App Purchase:

- Introduction
- Steps Involved
- Note
- Output

IAD Integration:

- Introduction
- Steps Involved
- Output

GameKit:

- Introduction
- Steps Involved
- Output

Storyboards:

- Introduction
- Steps Involved
- Output

Auto Layout:

- What we did here?
- Output

Twitter And Facebook:

- Output

Memory Management:

- Problems faced
- Memory Management rules
- Handling memory in ARC
- Using memory management tools
- Steps for analyzing memory allocations

Application Debugging:

- Selecting a Debugger
- How to find coding mistakes?
- Set Breakpoints

Understanding Swift:

- Introduction to Swift
- Introduction to XCode IDE
- Structure of Swift program
- Compiling and building a Hello world
- Swift Playground, Basic Syntax
- Variables and data types
- Structs and Classes.

Getting Started with Swift:

- Working with collections
- Protocols and Extensions, Generics
- Tuple, Functions
- Subscripts, Closures
- Automatic Reference Counting

iOS Development Basics:

- Installing XCode and iOS SDK
- Understanding XCode
- Creating a simple iOS app
- Project templates
- Introduction to pillars of iOS app development-Tools-XCode, iOS Simulator, Instruments
- Language- Swift, Design patterns, frameworks, Design-CorrectUI, AppleHIG, Testing, Provisioning, Signing, Submitting, Using iOS Simulator, MVC pattern in iOS app development

Storyboard Controllers and Layouts:

- The lifecycle of iOS Application
- Understanding Delegation
- iOS View Controllers, Creating UI
- Introduction to Layout and Views
- Designing responsive interfaces with Auto layout
- Create a sample UI
- Walkthrough of controllers
- Debugging iOS application

TableView, collectionView & Various View Controllers:

- Introduction to Table View
- Creating a basic table view and data source
- Loading data into TableView
- Reusing table view cells, Customizing table views
- UI collectionView
- UI RefreshControl
- Introduction to multiple-view applications
- Using the navigation controller
- Introduction to a master-detail application

- Creating a tab-bar-driven application
- UI PageViewController

Universal App, Size Classes & Camera:

- UISplitViewController
- Master-detail application
- Using Size Classes for Adaptive Layout
- Using Camera & Gallery to Capture Images

Animation, Location, Maps & Social Sharing:

- Animations- Core Animation, UIKit Dynamics, Facebook Pop Animation, Working with Maps- MKMapItem and MKPlacemark, CLLocationManager, MKMapView, UIActivityViewController to perform Social Sharing

JSON Handling, Network Operation & Image Caching:

- Networking with Alamofire
- SwiftyJSON & HanekeSwift
- Introduction to networking in iOS application, Adding/Managing frameworks using Carthage
- JSON Handling using SwiftyJSON
- Performing Networking using Alamofire
- Downloading & Caching images using HanekeSwift

Data Persistence and Storage:

- iOS data persistence
- Using NS User Defaults to store and read information
- Property Lists-Working with Core data, Introduction to Core Data, Creating models with entities, Saving models, Fetching model, Edit and remove models
- Sample application implementing Core data

CloudKit, Parse and Submitting Apps to App Store:

- Implementing CloudKit Storage
- **Interact with Parse backend**
- **Localizing Application**
- **Submitting an App to Apple App Store.**