

### Introduction to basic Swift Programming

Teaching Scheme	Credit	Examination Scheme
Theory : 2 Hrs/week Laboratory: 4 Hrs/week	Theory: 02 Practical:04	End Sem : 100 Marks

#### Course Objectives:

1. To learn and understand the basics of the Swift Language.
2. Understanding the programming concepts and features of Swift.
3. To understand the concepts of Variables, Data types and Control flow, functions, structures in Swift.
4. To provide knowledge of Interface Builder basics for Mobile app development using Xcode.

#### Course Outcomes:

Upon completion of the course, students will be able to:

- CO1. Identify and Understand the basic terminologies used in Swift programming Language.  
CO2. Proficient in using the basic constructs of Swift, to develop program.  
CO3. Code and debug Swift programs using Xcode and Playground  
CO4. Understand the fundamentals of Swift and be able to apply it in iOS app development.

Course Contents		
<b>Unit I</b>	<b>Installation Basics &amp; Introduction to Swift</b>	<b>08 Hours</b>

**Introduction to Xcode and Swift Playgrounds:** Installation of Swift on macOS and Linux, creating a package, Building an Executable, Working with multiple Source File. Installation of Xcode, Working with Xcode, create a simple program and execute it using Xcode, Working with swift playgrounds, create a simple program and execute it using swift playgrounds.

**Introduction of Swift:** features of Swift, Data types, constant and variables, operators Type Annotations, Naming Constants and Variables, Printing Constants and Variables,

<b>Unit II</b>	<b>Strings &amp; Control Flow</b>	<b>08 Hours</b>
----------------	-----------------------------------	-----------------

**Strings and Characters:** String Literals, Multiline String Literals, Special Characters in String Literals, Initializing an Empty String, String Mutability, Working with Characters, Concatenating Strings and Characters, String Interpolation, Counting Characters, Substrings, Comparing Strings, Prefix and Suffix Equality

**For-In Loops, While Loops:** While, Repeat-While. Conditional Statements: If-else, Switch, Control Transfer Statements: continue, break, fall through, return, and throw.

<b>Unit III</b>	<b>Functions</b>	<b>07 Hours</b>
-----------------	------------------	-----------------

**Functions:** Defining and Calling Functions, Function Parameters and Return Values: Functions Without Return Values, Functions with Multiple Return Values, Optional Tuple Return Types, Function Argument Labels and Parameter Names: Specifying Argument Labels, Omitting,

Argument Labels, Default Parameter Values, Variadic Parameters, Function Types, Function, Types as Parameter Types.

<b>Unit IV</b>	<b>Structures and Classes</b>	<b>07 Hours</b>
----------------	-------------------------------	-----------------

**Structures:** Definition Syntax, Structure and Class Instances, Accessing Properties, Memberwise Initializers for Structure Types, Value types or Reference Types. Properties: Stored Properties, Lazy Stored Properties, Computed Properties, Property Observers. Global and Local Variables, Type Properties, Type Property Syntax, Querying and Setting Type Properties  
Methods, Instance Methods, self-Property, Mutating Method, Type Methods Inheritance: Base Class, types of Inheritance, Sub classing, Overriding: Accessing Superclass Methods, Properties, and Subscripts, Overriding Methods, Overriding Properties, Overriding Property Getters and Setters, Preventing Overrides.

<b>Unit V</b>	<b>Interface Builder Basics</b>	<b>07 Hours</b>
---------------	---------------------------------	-----------------

**Introduction:** Introduction to iOS, Mobile application development, Overview of iOS platform, setting up Xcode & tools, MVC design pattern.

**Interface Builder Basics:** Common system views, Interface Builder Storyboards, project options, default project, create a new project with label and a greet function.

<b>Unit VI</b>	<b>Introduction to UIKit</b>	<b>07 Hours</b>
----------------	------------------------------	-----------------

**Common system views configuration:** Label (UILabel), Image view, Text view, Toolbars (UIToolbar), Navigation bars, tab bars, Controls, Button, Segmented controls, Sliders, Switches, Date pickers, UIKit User Interface Catalog, Displaying data: Content mode, Unexpected Clipping.

**Auto Layout:** Layout for multiple sizes, Why Auto Layout?, Create alignment constraints, create size constraints, Resolve constraint issues, Safe area layout guide, resolve constraint warnings, Constraints between siblings,

**Stack Views:** Stack views, stack view attributes, Size classes

<b>Learning Resources</b>
---------------------------

**Text Books:**

1. Matthew Mathias, John Gallagher, Swift Programming: The Big Nerd Ranch Guide 2nd edition, 2015.
2. Matt Neuberg, iOS 12 Programming Fundamentals with Swift, O'Reilly; 5th edition.
3. App Development with Swift (as available on iBook Store)

**Reference Books:**

1. Paris Buttfield-Addison, Jonathon Manning, Tim Nugent Learning Swift: Building Apps for macOS, iOS, and Beyond, O'Reilly Media, Inc., 3rd ed, 2018.
2. Jon Hoffman, Mastering Swift 4, Packt Publishing Limited, 4th edition, 2017.
3. Vandad Nahavandipoor. iOS 11 Swift Programming Cookbook, O'Reilly Media, 2017
4. S. Yamacli, Beginner's Guide to iOS 11 App Development Using Swift 4: Xcode, Swift and App Design Fundamentals, (1e), USA: CreateSpace Independent Publishing Platform, 2017.

<b>Swift Laboratory</b>
-------------------------

**Suggested List of Laboratory Experiments/Assignments**

Student should perform at least 15 experiments from group A (**Use suitable programming language/Tool for implementation**)

**Group A:**

1. Installation of Swift, Xcode and Playground.
2. Program to print Hello world (Using terminal and Xcode)
3. Program to demonstrate variable and constant declaration in Swift.
4. Program to demonstrate different arithmetic operators in Swift.
5. Program to demonstrate String Literals, Multiline string and special characters.

6. Program to demonstrate String mutability, Empty String and String Interpolation.
7. Program to demonstrate Characters in Swift.
8. Program to demonstrate various String comparisons in Swift.
9. Program to demonstrate For-In loop in Swift.
10. Program to demonstrate While loop in Swift.
11. Program to demonstrate Repeat-While in Swift.
12. Programs to demonstrate various control statements in Swift.
13. Programs to demonstrate function with and without return type and parameters.
14. Program to demonstrate function returning multiple values.
15. Program to demonstrate function returning optional tuple.
16. Programs to demonstrate function with and without argument label.
17. Program to demonstrate Structure.
18. Program to demonstrate Properties, Member wise and Initializers for Structure Types.
19. Programs to demonstrate Stored Properties, Lazy Stored Properties, and Computed Properties, and Property Observers.
20. Programs to demonstrate different types of Inheritance in Swift.

**Group B :**

1. Lab - Introduction.playground
2. Lab - Constants and Variables.playground
3. Lab - Operators.playground
4. Lab - Control Flow.playground
5. Lab - Strings.playground
6. Lab - Functions.playground
7. Lab - Structures.playground
8. Lab - Classes.playground
9. Lab - Loops.playground

**Group C:**

1. Guided Project: **Light**
2. Guided Project: **Apple Pie**