

Swift Tutorial

Swift Tutorial

Welcome to Swift Tutorial. In this series of Swift Tutorials, you will learn all the basics of Swift programming language with detailed examples.

Introduction to Swift

- Swift is a new programming language developed by Apple Inc.
- Swift is used to develop applications for iOS and OS X.

Pre-requisites for Swift Tutorial

This Swift Tutorial is targeted for beginners. So, every topic shall be explained in detail. You will find it better if you know basic programming concepts or any other programming language.

Swift Tutorials

In this Swift Tutorial series, we shall go through the following topics.

Swift Basic Tutorials

These basic Swift Tutorials gets you introduced with the datatypes available in Swift, how to declare variables, constants, etc, what are the keywords available in swift, and operators like logical, arithmetic, boolean, etc.

- [Swift Hello World Program](#)
- Swift Data Types
- Swift Variables
- [Swift Comments](#)
- Swift Optionals
- [Swift Tuples](#)
- [Swift Keywords](#)
- Swift Constants
- Swift Literals
- Swift Operators

Some how-tos..

- [How to create an Integer variable in Swift?](#)
- [How to create a String variable in Swift?](#)
- [How to get Type of variable in Swift?](#)

Swift Datatypes

The following Swift Tutorials deal with each of the datatype in Swift separately with examples.

- [Swift Boolean](#)

Swift Operators

The following Swift Tutorials deal with each of the operator in Swift separately with examples.

- [Swift Ternary Operator](#)
- [Swift AND Operator](#)
- [Swift OR Operator](#)
- [Swift NOT Operator](#)

Swift Decision Making Tutorials

The decision making statements are those that execute a block of statements based on the evaluation of an expression (or condition).

- [Swift if](#)
- [Swift if else](#)
- [Swift else if](#)
- [Swift nested if](#)
- [Swift switch](#)
- [Swift Ternary Operator](#)

Swift Loops Tutorials

Looping statements help to execute a set of statements repeatedly based on the evaluation of a condition. Also, there are loop control statements like break and continue to break a loop abruptly and continue with the other statements or skip the execution of statements for an iteration.

- [Swift for loop](#)
- [Swift while loop](#)
- [Swift repeat while loop](#)
- [Swift continue](#)
- [Swift break](#)

- [Swift break](#)
- [Swift fallthrough](#)
- [Swift forEach](#)
- [Swift forEach – break](#)

[Swift Integer Tutorials](#)

- [How to Generate Random Number in Swift?](#)

[Swift Strings Tutorials](#)

Strings are the basic data types that one uses for most of the real-time applications. Following series of Swift Tutorials help you on how to work with Strings in Swift, how to find the length of a string, how to check if two strings are equal, how to convert a string to integer or vice-versa, and other useful string operations.

- [How to find String Length in Swift?](#)
- [How to Concatenation Strings in Swift?](#)
- [How to Append a String to Another String in Swift?](#)
- [How to check if two Strings are Equal in Swift?](#)
- [How to check if String is Empty in Swift?](#)
- [How to find Sub-String in Swift?](#)
- [How to check if a String contains Another String in Swift?](#)
- [How to Split a String by Character Separator in Swift?](#)
- [How to Reverse a String in Swift?](#)
- [How to Insert a Character in String at Specific Index in Swift?](#)
- [How to Remove a Character in String at Specific Index in Swift?](#)
- [How to check if a String starts with Another String in Swift?](#)
- [How to check if a String ends with Another String in Swift?](#)
- [How to convert a String to Uppercase in Swift?](#)
- [How to convert a String to Lowercase in Swift?](#)
- [How to Convert String to Integer in Swift?](#)
- [How to include Double Quotes inside a String in Swift?](#)

[Swift Array Tutorials](#)

Arrays store like kind of elements in a sequential manner and can be accessed using a single variable name and of-course the index. Following Swift Tutorials deal with how to work with Arrays in Swift, starting from initialization and proceeding with different operations like printing, checking, appending, removing elements from the array.

Creating Swift Arrays

1. [Swift Array](#)
2. [Create Empty Array in Swift](#)
3. [Create Empty String Array in Swift](#)

4. [Create an Array of Specific Size in Swift](#)
5. [Create an Array with Default Value for Each Item in Swift](#)
6. [Create an Array with Initial Values in Swift](#)
7. [Swift Array Initialisation](#)
8. [Swift Integer Array](#)
9. [Swift String Array](#)
10. [Print Swift Array](#)
11. [Check if an Array is Empty](#)
12. [Get Swift Array Size](#)

Modify Swift Arrays

1. [Swift Array – Append an Element](#)
2. [Swift Array – Append Another Array](#)
3. [Append an integer to the Swift Array](#)
4. [Append String to Swift Array](#)
5. [Concatenate two Swift Arrays using Addition \(+\) Operator](#)
6. [Remove an element from Swift Array](#)

Accessing Swift Arrays

1. [Swift Array – Access Elements using Index](#)
2. [Swift Array – Loop over Elements using For Loop](#)
3. [Swift Array – Loop over Elements using While Loop](#)
4. [Swift Array – Loop over Elements using forEach Loop](#)

Swift Set Tutorials

Sets are collection of items that are stored not necessarily in sequential manner.

1. [Swift Set](#)
2. [Check if an element is present in a Set](#)
3. [Add or insert an element to Swift Set](#)
4. [Get Set Size](#)
5. [Print elements of Set](#)

Swift Dictionary Tutorials

1. [Swift Dictionary](#)
2. [Create Swift Dictionary](#)
3. [Create Dictionary using Arrays for keys and values](#)
4. [Convert Dictionary into Arrays of keys and values](#)
5. [Get Dictionary Size](#)
6. [Iterate Swift Dictionary](#)
7. [Append Element to Dictionary](#)
8. [Get Value using Key in Dictionary](#)

- o. [Get value using key in Dictionary](#)
- 9. [Check if a specific key is present in the Dictionary](#)
- 10. [Check if the Dictionary is empty](#)
- 11. [Print all keys in a Dictionary](#)
- 12. [Merge Dictionaries](#)

Other Useful Swift Tutorials

- [Swift Characters](#)
- [Swift Functions](#)
- [Swift Closures](#)
- [Swift Enum](#)
- [Swift Structures](#)
- [Swift Classes](#)
- [Swift Properties](#)
- [Swift Methods](#)
- [Swift Subscript](#)
- [Swift Inheritance](#)
- [Swift Initialization](#)
- [Swift Deinitialization](#)
- [Swift ARC](#)
- [Swift Optional Chaining](#)
- [Swift Type Casting](#)
- [Swift Extensions](#)
- [Swift Protocols](#)
- [Swift Generics](#)
- [Swift Access Control](#)

Swift Programs

- [How to generate a Random Number in Swift?](#)

Swift Tutorial

⇒ [Swift Tutorial](#)

◆ [Swift Keywords](#)

◆ [Swift Comments](#)

◆ [Swift If](#)

◆ [Swift If-Else](#)

◆ [Swift For Loop](#)

◆ [Swift While Loop](#)

◆ [Swift forEach](#)

◆ [Swift Repeat While Loop](#)

◆ [Swift Break](#)

◆ [Swift Continue](#)

◆ [Swift Tuple](#)

◆ [Swift Enum](#)

◆ [Swift Structure](#)

Strings

◆ [Swift - Substring](#)

◆ [Swift - Concatenate Strings](#)

Arrays

◆ [Swift Array Initialization](#)

◆ [Swift Print Array](#)

◆ [Swift Integer Array](#)

◆ [Swift append Integer to Array](#)

◆ [Swift String Array](#)

◆ [Swift append String to Array](#)

◆ [Swift Get Array Size - count](#)

◆ [Swift Remove an Element from Array](#)

◆ [Swift Append / Concatenate Arrays](#)

◆ [Swift Check if an Array is Empty](#)

Dictionaries

◆ [Swift Dictionary](#)

◆ [Swift - Create Dictionary](#)

◆ [Swift - Create Dictionary using Arrays](#)

◆ [Swift - Iterate through Dictionary](#)

◆ [Swift - Get Dictionary Size](#)

◆ [Swift - Check if Dictionary is Empty](#)

◆ [Swift - Add or Append Element to Dictionary](#)

◆ [Swift - Get value using key in Dictionary](#)

◆ [Swift - Check if a key is present in Dictionary](#)

▼ Swift - Check if a key is present in Dictionary

◆ Swift - Merge Two Dictionaries

◆ Swift - Convert Dictionary into Arrays of Keys and Values

◆ Swift - Print all Keys in a Dictionary

Sets

◆ Swift Print Set

◆ Swift Get Set Size

◆ Swift Insert Element to Set

◆ Swift Check if Element is present in Set

File Operations

◆ Swift Read Text File

Swift Errors [Solved]

◆ Swift error: return from initializer without initializing all stored properties

◆ Swift – struct error: missing argument labels in call