**Swift vs SwiftUI**

Swift and SwiftUI are two distinct, yet complementary, technologies in Apple's ecosystem for developing iOS, macOS, watchOS, and tvOS applications. Here’s a detailed comparison of Swift and SwiftUI to help understand their roles, differences, and how they can be used together.

**Swift**

**Type: Programming Language**

* Purpose: General-purpose, multi-paradigm, compiled programming language.
* Introduced: 2014 by Apple as a successor to Objective-C.
* Usage: Can be used to develop applications across different Apple platforms (iOS, macOS, watchOS, tvOS) as well as for general-purpose programming.

**Key Features**

Syntax and Safety: Swift offers a clean and modern syntax with features that help avoid common programming errors (e.g., optionals to handle nil values safely).

**Usage Example**

Swift is used to write the entire application logic, from handling data processing, network requests, and business logic to creating user interfaces with UIKit or AppKit.

**SwiftUI**

**Overview**

* Type: UI Framework
* Purpose: A modern framework for building user interfaces across all Apple platforms using declarative syntax.
* Introduced: 2019 by Apple.
* Usage: Focused on building UI components and managing the user interface state in a declarative manner.

**Key Features**

Declarative Syntax: SwiftUI uses a declarative syntax to define user interfaces. You describe what the UI should look like, and SwiftUI takes care of the rendering and state management.

**Usage Example**

SwiftUI is used to define the user interface and handle the layout and visual aspects of the application.

**When to Use Swift and SwiftUI**

Swift is the core language for developing any application logic, including data processing, network communication, and integrating with APIs. You need to know Swift to use SwiftUI effectively because SwiftUI itself is built with Swift and relies on its syntax and features.

SwiftUI is ideal for building user interfaces quickly and declaratively. It is especially useful for new projects or when targeting multiple Apple platforms with minimal code differences.