iOS App Life Cycle

What is the App Life Cycle?

The App Life Cycle defines the sequence of states an app goes through from launch to termination. These states help you manage resources, save user data, and optimize performance.

App States in iOS:

1. Not Running - The app is not launched or has been terminated by the system or user.
2. Inactive - The app is in the foreground but not receiving events.
3. Active - The app is in the foreground and receiving user events.
4. Background - The app is running in the background and may still execute code.
5. Suspended - The app is in the background but not executing code (iOS freezes it to save resources).

App Life Cycle Methods (UIKit):

1. Application Launch

```swift
func application(_ application: UIApplication, didFinishLaunchingWithOptions launchOptions: [UIApplication.LaunchOptionsKey: Any]?) -> Bool {
    print("App Launched")
    return true
}
```

2. Moving to Inactive

```swift
func applicationWillResignActive(_ application: UIApplication) {
```

```
    print("App Will Resign Active")
}
```

3. Moving to Background

```swift
func applicationDidEnterBackground(_ application: UIApplication) {
    print("App Entered Background")
}
```

4. Moving to Foreground

```swift
func applicationWillEnterForeground(_ application: UIApplication) {
    print("App Will Enter Foreground")
}
```

5. Becoming Active Again

```swift
func applicationDidBecomeActive(_ application: UIApplication) {
    print("App Became Active")
}
```

6. App Termination

```swift
func applicationWillTerminate(_ application: UIApplication) {
    print("App Will Terminate")
}
```

Life Cycle in SwiftUI:

In SwiftUI, use @Environment(.scenePhase) to track state changes:

```swift
import SwiftUI

struct ContentView: View {
    @Environment(\.scenePhase) private var scenePhase

    var body: some View {
        Text("Hello, World!")
            .onChange(of: scenePhase) { newPhase in
                switch newPhase {
                case .active:
                    print("App Became Active")
                case .inactive:
                    print("App is Inactive")
                case .background:
                    print("App Moved to Background")
                default:
                    break
                }
            }
    }
}
```

Key Takeaways:

- The iOS App Life Cycle has 5 states: Not Running, Inactive, Active, Background, and Suspended.

- Use UIApplicationDelegate in UIKit-based apps to track state transitions.

- Use scenePhase in SwiftUI apps for life cycle management.

- Always save user data when the app enters the background to prevent data loss.