

[SwiftUI](#) / Commands

Protocol

Commands

Conforming types represent a group of related commands that can be exposed to the user via the main menu on macOS and key commands on iOS.

iOS 14.0+ | iPadOS 14.0+ | Mac Catalyst 14.0+ | macOS 11.0+ | visionOS 1.0+

```
@MainActor @preconcurrency  
protocol Commands
```

Mentioned in

 Building and customizing the menu bar with SwiftUI

Overview

A type conforming to this protocol inherits `@preconcurrency @MainActor` isolation from the protocol if the conformance is included in the type's base declaration:

```
struct MyCustomType: Transition {  
    // `@preconcurrency @MainActor` isolation by default  
}
```

Isolation to the main actor is the default, but it's not required. Declare the conformance in an extension to opt out of main actor isolation:

```
extension MyCustomType: Transition {  
    // `nonisolated` by default  
}
```

Topics

Implementing commands

`var body: Self.Body`

The contents of the command hierarchy.

Required

`associatedtype Body : Commands`

The type of commands that represents the body of this command hierarchy.

Required

Relationships

Conforming Types

CommandGroup

CommandMenu

EmptyCommands

Group

Conforms when `Content` conforms to `Commands`.

ImportFromDevicesCommands

InspectorCommands

SidebarCommands

TextEditingCommands

TextFormattingCommands

ToolbarCommands

See Also

Defining commands

`func commands<Content>(content: () -> Content) -> some Scene`

Adds commands to the scene.

```
func commandsRemoved() -> some Scene
```

Removes all commands defined by the modified scene.

```
func commandsReplaced<Content>(content: () -> Content) -> some Scene
```

Replaces all commands defined by the modified scene with the commands from the builder.

```
struct CommandMenu
```

Command menus are stand-alone, top-level containers for controls that perform related, app-specific commands.

```
struct CommandGroup
```

Groups of controls that you can add to existing command menus.

```
struct CommandsBuilder
```

Constructs command sets from multi-expression closures. Like `ViewBuilder`, it supports up to ten expressions in the closure body.

```
struct CommandGroupPlacement
```

The standard locations that you can place new command groups relative to.