

[SwiftUI](#) / [Scene](#) / `commandsReplaced(content:)`

## Instance Method

# commandsReplaced(content:)

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

iOS 16.0+ | iPadOS 16.0+ | Mac Catalyst 16.0+ | macOS 13.0+ | visionOS 1.0+

```
nonisolated
func commandsReplaced<Content>(@Commands
Builder content: () -> Content) -> some Scene where Content : Commands
```

## Parameters

### content

A Commands builder whose output will be used to replace the commands normally provided by the modified scene.

## Return Value

A scene that replaces any commands defined by its children with alternative content.

## Discussion

WindowGroup, Window, and other scene types all have an associated set of commands that they include by default. Apply this modifier to a scene to replace those commands with the output from the given builder.

For example, the following code adds a scene for showing the contents of the pasteboard in a dedicated window. We replace the scene's default Window > Clipboard menu command with a custom Edit > Show Clipboard command that we place next to the other pasteboard commands.

```

@main
struct Example: App {
    @Environment(\.openWindow) var openWindow

    var body: some Scene {
        ...

        Window("Clipboard", id: "clipboard") {
            ClipboardContentView()
        }
        .commandsReplaced {
            CommandGroup(after: .pasteboard) {
                Section {
                    Button("Show Clipboard") {
                        openWindow(id: "clipboard")
                    }
                }
            }
        }
    }
}

```

## 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.

```
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.

```
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.