

[SwiftUI](#) / WidgetBundle

Protocol

WidgetBundle

A container used to expose multiple widgets from a single widget extension.

iOS 14.0+ | iPadOS 14.0+ | Mac Catalyst 14.0+ | macOS 11.0+ | visionOS 1.0+ | watchOS 9.0+

```
@MainActor @preconcurrency
protocol WidgetBundle
```

Overview

To support multiple types of widgets, add the `@main` attribute to a structure that conforms to `WidgetBundle`. For example, a game might have one widget to display summary information about the game and a second widget to display detailed information about individual characters.

```
@main
struct GameWidgets: WidgetBundle {
    var body: some Widget {
        GameStatusWidget()
        CharacterDetailWidget()
    }
}
```

Topics

Implementing a widget bundle

```
var body: Self.Body
```

Declares the group of widgets that an app supports.

Required

```
associatedtype Body : Widget
```

The type of widget that represents the content of the bundle.

Required

```
struct WidgetBundleBuilder
```

A custom attribute that constructs a widget bundle's body.

Running a widget bundle

```
init()
```

Creates a widget bundle using the bundle's body as its content.

Required

```
static func main()
```

Initializes and runs the widget bundle.

See Also

Creating widgets

 [Building Widgets Using WidgetKit and SwiftUI](#)

Create widgets to show your app's content on the Home screen, with custom intents for user-customizable settings.

 [Creating a widget extension](#)

Display your app's content in a convenient, informative widget on various devices.

 [Keeping a widget up to date](#)

Plan your widget's timeline to show timely, relevant information using dynamic views, and update the timeline when things change.

 [Making a configurable widget](#)

Give people the option to customize their widgets by adding a custom app intent to your project.

`protocol Widget`

The configuration and content of a widget to display on the Home screen or in Notification Center.

`struct LimitedAvailabilityConfiguration`

A type-erased widget configuration.

`protocol WidgetConfiguration`

A type that describes a widget's content.

`struct EmptyWidgetConfiguration`

An empty widget configuration.