

□ Documentation

[Xcode](#) / [Localization](#) / Previewing localizations

Article

Previewing localizations

Test localizations in the SwiftUI preview or the Interface Builder preview.



Overview

You can preview localizations early in development while you lay out your app's interface for both SwiftUI and Interface Builder apps. For SwiftUI apps, you need to add the language or import the localization before you preview it, described in [Adding support for languages and regions](#) and [Importing localizations](#). For Interface Builder apps, you can first preview the interface in pseudolanguages and then later in the localizations you add.

Add localizations to a SwiftUI preview

For SwiftUI apps, you can preview a localization by setting the `locale` environment variable in your code. Use the `environment(_ : :)` function to set the locale for all views in the view hierarchy of a SwiftUI preview. For example, if you add German to your project, you can set the locale to German (de):

```
struct ContentView_Previews: PreviewProvider {
    static var previews: some View {
        ContentView()
            .environment(\.locale, .init(identifier: "de"))
    }
}
```

To preview right-to-left languages, also set the `layoutDirection` key to `LayoutDirection.rightToLeft`.

To preview multiple localizations, add additional previews to your code and set the `locale` environment value for each. The localizations appear in the SwiftUI preview.

The screenshot shows the Xcode interface with the project navigation bar at the top. Below it is the file browser showing the project structure, including files like `HelloWorldApp.swift`, `ContentView.swift`, `Assets.xcassets`, `Info.plist`, and test files. The main editor area displays the `ContentView.swift` code:

```
1 //  
2 // ContentView.swift  
3 // HelloWorld  
4 //  
5 // Created by Ravi Patel on 9/8/20.  
6 //  
7  
8 import SwiftUI  
9  
10 struct ContentView: View {  
11     var body: some View {  
12         Text("Hello, world!", comment:"Friendly greeting")  
13             .font(.system(size: 50))  
14             .padding()  
15     }  
16 }  
17  
18 struct ContentView_Previews: PreviewProvider {  
19     static var previews: some View {  
20         Group {  
21             ContentView()  
22                 .environment(\.locale, .init(identifier:"de"))  
23         }  
24     }  
25 }  
26 }  
27
```

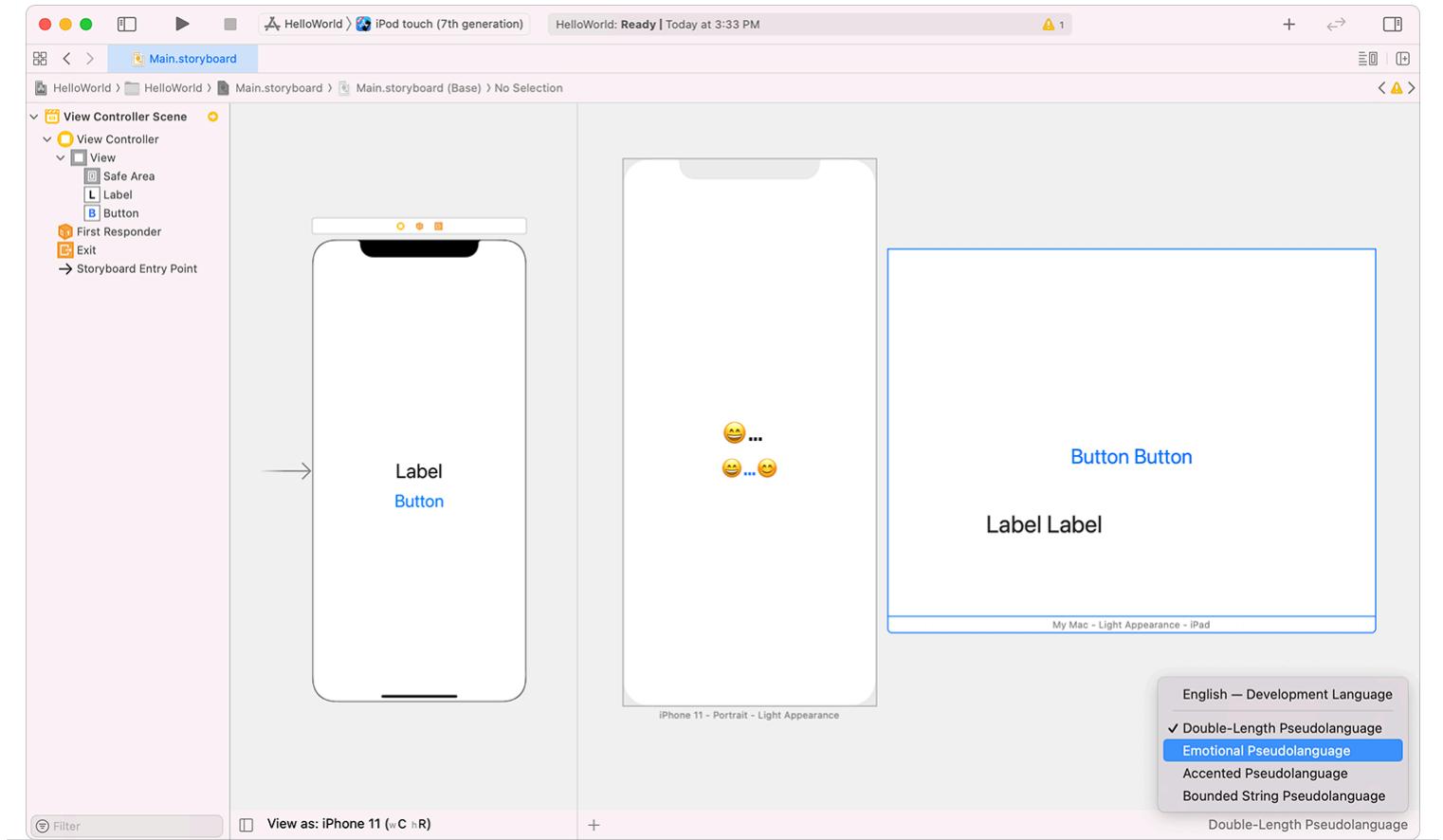
To the right of the code editor is a preview window showing a blue iPhone with the text "Hallo, Welt!" displayed on its screen. The bottom right corner of the preview window shows a language selection button with "de" selected.

Preview localizations in Interface Builder

For Interface Builder, you can preview localizations any time during development by choosing localizations including pseudolanguages in the preview. You don't need to build and run your app to see the preview.

In Interface Builder, select the desired view controller. Click the Adjust Editor Options button in the upper-right corner, then choose Preview. A preview of the layout appears to the right of the canvas.

In the preview area, select a preview or click in the background to deselect all previews. If you don't select a preview, you'll change the language of all previews. Click the language button—for example, English—in the lower-right corner. In the pop-up menu that appears, choose a localization or pseudolanguage.



See Also

Related Documentation

[Adding support for languages and regions](#)

Select the resources that you want to localize for each language and region you support.

[Importing localizations](#)

Import the files that you translate or adapt for a language and region into your project.

`struct EnvironmentValues`

A collection of environment values propagated through a view hierarchy.

Testing

[Testing localizations when running your app](#)

Run your app in each language and region you support to thoroughly test your app.