SwiftUI / View / accessibilityChildren(children:)

Instance Method

# accessibilityChildren(children:)

Replaces the existing accessibility element's children with one or more new synthetic accessibility elements.

iOS 15.0+ | iPadOS 15.0+ | Mac Catalyst 15.0+ | macOS 12.0+ | tvOS 15.0+ | visionOS 1.0+ | watchOS 8.0+

```
nonisolated
func accessibilityChildren<V>(@View
Builder children: () -> V) -> some View where V : View
```

## Parameters

`children`

A [ViewBuilder](#) that represents the replacement child views the framework uses to generate accessibility elements.

## Discussion

Use this modifier to replace an existing element's children with one or more new synthetic accessibility elements you provide. This allows for synthetic, non-visual accessibility elements to be set as children of a visual accessibility element.

SwiftUI creates an accessibility container implicitly when needed. If an accessibility element already exists, the framework converts it into an accessibility container.

In the example below, a `Canvas` displays a graph of vertical bars that don't have any inherent accessibility elements. You make the view accessible by adding the `accessibilityChildren(children:)` modifier with views whose accessibility elements represent the values of each bar drawn in the canvas:

```swift
var body: some View {
    Canvas { context, size in
        // Draw Graph
        for data in dataSet {
            let path = Path(
                roundedRect: CGRect(
                    x: (size.width / CGFloat(dataSet.count))
                    * CGFloat(data.week),
                    y: 0,
                    width: size.width / CGFloat(dataSet.count),
                    height: CGFloat(data.lines),
                cornerRadius: 5)
            context.fill(path, with: .color(.blue))
        }
        // Draw Axis and Labels
        ...
    }
    .accessibilityLabel("Lines of Code per Week")
    .accessibilityChildren {
        HStack {
            ForEach(dataSet) { data in
                RoundedRectangle(cornerRadius: 5)
                    .accessibilityLabel("Week \(data.week)")
                    .accessibilityValue("\(data.lines) lines")
            }
        }
    }
}
```

SwiftUI hides any views that you provide with the `children` parameter, then the framework uses the views to generate the accessibility elements.

---

# See Also

## Creating accessible elements

```swift
func accessibilityElement(children: AccessibilityChildBehavior) -> some
View
```

Creates a new accessibility element, or modifies the __AccessibilityChildBehavior__ of the existing accessibility element.

```
func accessibilityRepresentation<V>(representation: () -> V) -> some
View
```
   Replaces one or more accessibility elements for this view with new accessibility elements.

```
struct AccessibilityChildBehavior
```
   Defines the behavior for the child elements of the new parent element.

```
func accessibilityRepresentation<V>(representation: () -> V) -> some
View
```
   Replaces one or more accessibility elements for this view with new accessibility elements.

```
struct AccessibilityChildBehavior
```
   Defines the behavior for the child elements of the new parent element.