

< All Technologies

SwiftUI

Font

Getting standard fonts

- static let extraLargeTitle2: Font
- static let extraLargeTitle: Font
- static let largeTitle: Font
- static let title: Font
- static let title2: Font
- static let title3: Font
- static let headline: Font
- static let subtitle: Font
- static let body: Font
- static let callout: Font
- static let caption: Font
- static let caption2: Font
- static let footnote: Font

Getting system fonts

- static func system(Font.TextStyl...
- static func system(size: CGFloat,...

Filter

/

SwiftUI / Font

Structure

Font

An environment-dependent font.

iOS 13.0+ | iPadOS 13.0+ | Mac Catalyst 13.0+ | macOS 10.15+ | tvOS 13.0+ | visionOS 1.0+ | watchOS 6.0+

@frozen  
struct Font

Overview

The system resolves a font's value at the time it uses the font in a given environment because `Font` is a late-binding token.

Topics

Getting standard fonts

`static let extraLargeTitle2: Font`  
Create a font with the second level extra large title text style.

`static let extraLargeTitle: Font`  
Create a font with the extra large title text style.

`static let largeTitle: Font`  
A font with the large title text style.

`static let title: Font`  
A font with the title text style.

`static let title2: Font`  
Create a font for second level hierarchical headings.

`static let title3: Font`  
Create a font for third level hierarchical headings.

`static let headline: Font`  
A font with the headline text style.

`static let subheadline: Font`  
A font with the subheadline text style.

`static let body: Font`  
A font with the body text style.

`static let callout: Font`  
A font with the callout text style.

`static let caption: Font`  
A font with the caption text style.

`static let caption2: Font`  
Create a font with the alternate caption text style.

`static let footnote: Font`  
A font with the footnote text style.

Getting system fonts

`static func system(Font.TextStyle, design: Font.Design?, weight: Font.Weight?) -> Font`  
Gets a system font that uses the specified style, design, and weight.

`static func system(size: CGFloat, weight: Font.Weight?, design: Font.Design?) -> Font`  
Specifies a system font to use, along with the style, weight, and any design parameters you want applied to the text.

`enum Design`  
A design to use for fonts.

`enum TextStyle`  
A dynamic text style to use for fonts.

`struct Weight`  
A weight to use for fonts.

Creating custom fonts

`static func custom(String, fixedSize: CGFloat) -> Font`  
Create a custom font with the given name and a fixed size that does not scale with Dynamic Type.

`static func custom(String, size: CGFloat, relativeTo: Font.TextStyle) -> Font`  
Create a custom font with the given name and size that scales relative to the given textStyle.

`static func custom(String, size: CGFloat) -> Font`  
Create a custom font with the given name and size that scales with the body text style.

Getting a font from another font

`init(CTFont)`  
Creates a custom font from a platform font instance.

Styling a font

`func bold() -> Font`  
Adds bold or emphasized styling to the font.

`func italic() -> Font`  
Adds italics to the font.

`func monospaced() -> Font`  
Returns a fixed-width font from the same family as the base font.

`func monospacedDigit() -> Font`  
Returns a modified font that uses fixed-width digits, while leaving other characters proportionally spaced.

`func smallCaps() -> Font`  
Adjusts the font to enable all small capitals.

`func lowercaseSmallCaps() -> Font`  
Adjusts the font to enable lowercase small capitals.

`func uppercaseSmallCaps() -> Font`  
Adjusts the font to enable uppercase small capitals.

`func weight(Font.Weight) -> Font`  
Sets the weight of the font.

`func width(Font.Width) -> Font`  
Sets the width of the font.

`struct Width`  
A width to use for fonts that have multiple widths.

`func leading(Font.Leading) -> Font`  
Adjusts the line spacing of a font.

`enum Leading`  
A line spacing adjustment that you can apply to a font.

Deprecated symbols

~~`static func system(Font.TextStyle, design: Font.Design) -> Font`~~  
Gets a system font with the given text style and design.  

Deprecated

~~`static func system(size: CGFloat, weight: Font.Weight, design: Font.Design) -> Font`~~  
Specifies a system font to use, along with the style, weight, and any design parameters you want applied to the text.  

Deprecated

Structures

`struct Context`  
Information used to resolve a font.

`struct Resolved`  
A concrete font value.

Instance Methods

`func bold(Bool) -> Font`  
Adds or removes bold or emphasized styling on the font.

`func italic(Bool) -> Font`  
Adds/removes italics on the font.

`func lowercaseSmallCaps(Bool) -> Font`  
Adjusts the font to enable/disable lowercase small capitals.

`func monospaced(Bool) -> Font`  
Returns a font adding or removing fixed-width design from the same family as the base font.

`func pointSize(CGFloat) -> Font`  
Sets the point size of the font explicitly.

`func resolve(in: Font.Context) -> Font.Resolved`  
Evaluates this font to a resolved font given the current context.

`func scaled(by: CGFloat) -> Font`  
Scales the point size of the font.

`func smallCaps(Bool) -> Font`  
Adjusts the font to enable/disable all small capitals.

`func uppercaseSmallCaps(Bool) -> Font`  
Adjusts the font to enable/disable uppercase small capitals.

Relationships

Conforms To

[Equatable](#)  
[Hashable](#)  
[Sendable](#)  
[SendableMetatype](#)

See Also

Setting a font

[Applying custom fonts to text](#)  
Add and use a font in your app that scales with Dynamic Type.

`func font(Font?) -> some View`  
Sets the default font for text in this view.

`func fontDesign(Font.Design?) -> some View`  
Sets the font design of the text in this view.

`func fontWeight(Font.Weight?) -> some View`  
Sets the font weight of the text in this view.

`func fontWidth(Font.Width?) -> some View`  
Sets the font width of the text in this view.

`var font: Font?`  
The default font of this environment.