#WWDC18

TextKit Best Practices

Session 221

Donna Tom, TextKit
Emily Van Haren, Authoring Tools

© 2018 Apple Inc. All rights reserved. Redistribution or public display not permitted without written permission from Apple.

Key concepts

Examples

Best practices

Key concepts

Examples

Best practices

What is TextKit?

```
import TextKit
```

```
import TextKit
```

No such module 'TextKit'

TextKit

UIKit
UITextField
UITextView
UILabel
AppKit
NSTextField
NSTextView
NSSecureTextField
TextKit

UIKit
UITextField
UITextView
UILabel
AppKit
NSTextField
NSTextView
NSSecureTextField
TextKit
Core Text
Core Graphics
Foundation

Emoji

Glyphs

Tracking

Kerning

Ligatures

Orphan avoidance

Tightening

Hyphenation

Bidirectional text

Line breaking

Truncation

Font fallback

Autocorrect

Paste

Copy

Touch Bar

Dictation

Drag & Drop

Accessibility

App Extensions

Marked text attributes

Key bindings

Redo

Lookup

Selection

Undo

Choosing the Right Control

Choosing the Right Control

UIKit

Choosing the Right Control

UIKit

Choosing the Right Control

UIKit

Choosing the Right Control

UIKit

Choosing the Right Control

UIKit

Choosing the Right Control

UIKit

Choosing the Right Control

UIKit

Choosing the Right Control

UIKit

Choosing the Right Control

UIKit

Choosing the Right Control

UIKit

Choosing the Right Control

AppKit

Choosing the Right Control

AppKit

Choosing the Right Control

AppKit

Choosing the Right Control

AppKit

Choosing the Right Control

AppKit

Choosing the Right Control

AppKit

Choosing the Right Control

AppKit

Choosing the Right Control

AppKit

What about string drawing?

What about string drawing?

```
func draw(at: CGPoint)

func draw(in: CGRect)

func draw(with: CGRect,
          options: NSStringDrawingOptions = [],
          context: NSStringDrawingContext?)
```

Choosing the Right Control

String drawing

Use for small amounts of text

Limit drawing frequency

Limit number of custom attributes

Choosing the Right Control

	Label	Text Field	Text View	String Drawing
Display text on screen	✅	✅	✅	✅
Respond to touch and/or mouse events	✅	✅	✅	
Accessibility support	✅	✅	✅	
Vibrancy	✅	✅	✅	
Respond to focus changes		✅	✅	
Selection and editing		✅	✅	
Secure text entry		✅		

Choosing the Right Customization Point

Controller
Model
View

Controller
Layout
Model
Storage
View
Display

Layout
Storage
NSTextStorage
NSTextContainer
Display

Choosing the Right Customization Point

NSTextStorage

Holds string data and attributes

`Waffles` are *very* delicious.

Choosing the Right Customization Point

NSTextContainer

Defines geometric area for text layout

Documentation

Text System Storage Layer Overview

Attributed String Programming Guide

Text Attribute Programming Topics

Introduction to Attributed Strings for iOS	WWDC 2012
Advanced Attributed Strings for iOS	WWDC 2012

Layout
Storage
NSTextStorage
NSTextContainer
Display

Layout
Storage
Display
UILabel
UITextField
UITextView
NSTextField
NSTextView
NSSecureTextField

Documentation

Text Programming Guide for iOS

Text System User Interface Layer Programming Guide

Layout
Storage
Display
UILabel
UITextField
UITextView
NSTextField
NSTextView
NSSecureTextField

Layout
Storage
Display

Layout
NSLayoutManager
Storage
Display

Choosing the Right Customization Point

NSLayoutManager

Coordinates between storage and display

Controls layout process

Choosing the Right Customization Point

Layout process

Before:

Tempura (天麩羅) is a tasty Japanese food. 🍤

Times New Roman

Choosing the Right Customization Point

Layout process

After:

Tempura (天麩羅) is a tasty Japanese food. 🍤

Choosing the Right Customization Point

Layout process

Choosing the Right Customization Point

Layout process

Choosing the Right Customization Point

Glyphs

Choosing the Right Customization Point

Glyphs

Choosing the Right Customization Point

Layout process

Choosing the Right Customization Point

Layout process

Choosing the Right Customization Point

Layout process

Documentation

Text Layout Programming Guide

Advanced Layouts and Effects With Text Kit WWDC 2013

Layout
NSLayoutManager
Storage
Display

Controller
Layout
Model
Storage
View
Display

Choosing the Right Configuration

This is the standard configuration. The text container holds a weak reference to the text view, and the text view retains the whole tree by retaining the root text storage.

As text is added, it fills the region defined by the first text container. The text is displayed in the text view that is paired with the text container.

When there's no more room, another text container (and associated text view) is added, and the text flows onto the second page.

Multiple layout managers allow you to have multiple presentations of the same text. The text within each view can have separate layout and selections.

Multiple layout managers allow you to have multiple presentations of the same text. The text within each view can have separate layout and selections.

Related Previous Sessions

Advanced Cocoa Text Tips and Tricks	WWDC 2010

Choosing the Right Customization Approach

Choose the right tool for the job

Choosing the Right Customization Approach

Delegation

Notifications

Subclassing

Choosing the Right Customization Approach

Delegation

Notifications

Subclassing

Choosing the Right Customization Approach

Delegation

Notifications

Subclassing

Key concepts

Examples

Best practices

TextKit is everywhere.

Apple News on iOS

9:41
Search
LTE
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
FEATURED
Students succeed with iPad and Mac
When you walk into Wilder Elementary School, just outside of Boise, Idaho, it's what
For You
Spotlight
Following
Search
Saved

9:41
Search
LTE
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
FEATURED
Students succeed with iPad and Mac
When you walk into Wilder Elementary School, just outside of Boise, Idaho, it's what
For You
Spotlight
Following
Search
Saved

9:41
LTE
Search
Spotlight
WEEKEND PURSUITS
From the Apple News Editors

Choosing the right control.

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
2
AppleNewsWWDC
AppleNewsWWDC
Main.storyboard
Main.storyboard (Base)
View Controller Scene
View Controller
View
View Controller Scene
View Controller
View
Safe Area
First Responder
Exit
Storyboard Entry Point
Objects
View Controller - A controller that manages a view.
Storyboard Reference - Provides a placeholder for a view controller in an external storyboard.
Navigation Controller - A controller that manages navigation through a hierarchy of views.
Table View Controller - A controller that manages a table view.
Collection View Controller - A controller that manages a collection view.
Tab Bar Controller - A controller that manages a set of view controllers that represent tab bar items.
Split View Controller - A composite view controller that manages left and
View
Content Mode
Scale To Fill
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
1
Custom
Default
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews
0
0
X
Y
1
1
Width
Height
Filter
View as: iPhone X (wC hR)
77%

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
AppleNewsWWDC
AppleNewsWWDC
Main.storyboard
Main.storyboard (Base)
View Controller Scene
View Controller
View
View Controller Scene
View Controller
View
Safe Area
First Responder
Exit
Storyboard Entry Point
Objects
View Controller - A controller that manages a view.
Storyboard Reference - Provides a placeholder for a view controller in an external storyboard.
Navigation Controller - A controller that manages navigation through a hierarchy of views.
Table View Controller - A controller that manages a table view.
Collection View Controller - A controller that manages a collection view.
Tab Bar Controller - A controller that manages a set of view controllers that represent tab bar items.
Split View Controller - A composite view controller that manages left and
View
Content Mode
Scale To Fill
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
1
Custom
Default
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews
0
0
X
Y
1
1
Width
Height
Filter
View as: iPhone X (wC hR)
77%

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
AppleNewsWWDC
AppleNewsWWDC
Main.storyboard
Main.storyboard (Base)
View Controller Scene
View Controller
View
View Controller Scene
View Controller
View
Safe Area
First Responder
Exit
Storyboard Entry Point
label
Label Label - A variably sized amount of static text.
View
Content Mode
Scale To Fill
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
1
Custom
Default
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews
Stretching
0
0
X
Y
1
1
Width
Height
Filter
View as: iPhone X (wC hR)
77%

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
AppleNewsWWDC
AppleN...WWDC
Main.storyboard
Main.st...(Base)
View C...r Scene
View Controller
View
Label
View Controller Scene
View Controller
View
Safe Area
Label
Constraints
First Responder
Exit
Storyboard Entry Point
Label
Text
Plain
Label
Color
Dark Text Color (De...
Font
System Semibold 17.0
Dynamic Type
Automatically Adjusts Font
Alignment
Lines
1
Behavior
Enabled
Highlighted
Baseline
Align Baselines
Line Break
Truncate Tail
Autoshrink
Fixed Font Size
Tighten Letter Spacing
Highlighted
Default
Shadow
Default
Shadow Offset
0
-1
Width
Height
View
Content Mode
Left
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
Alpha
1
Background
Custom
Tint
Default
Drawing
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews
Filter
View as: iPhone X (wC hR)
77%

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
AppleNewsWWDC
AppleN...WWDC
Main.storyboard
Main.st...(Base)
View C...r Scene
View Controller
View
Label
View Controller Scene
View Controller
View
Safe Area
Label
Constraints
First Responder
Exit
Storyboard Entry Point
Label
View as: iPhone X (wC hR)
77%
Filter
Label
Text Plain
Label
Color Dark Text Color (De...
Font System Semibold 17.0
Dynamic Type Automatically Adjusts Font
Alignment
Lines 1
Behavior Enabled
Highlighted
Baseline Align Baselines
Line Break Truncate Tail
Autoshrink Fixed Font Size
Tighten Letter Spacing
Highlighted Default
Shadow Default
Shadow Offset 0 -1
Width Height
View
Content Mode Left
Semantic Unspecified
Tag 0
Interaction User Interaction Enabled
Multiple Touch
Alpha 1
Background Custom
Tint Default
Drawing Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Spotlight
View Controller Scene
View Controller
View
Safe Area
Spotlight
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
View as: iPhone X (wC hR)
77%
Label
Text
Plain
Spotlight
Color
Dark Text Color (De...
Font
System Semibold 17.0
Dynamic Type
Automatically Adjusts Font
Alignment
Lines
1
Behavior
Enabled
Highlighted
Baseline
Align Baselines
Line Break
Truncate Tail
Autoshrink
Fixed Font Size
Tighten Letter Spacing
Highlighted
Default
Shadow
Default
Shadow Offset
0
-1
Width
Height
View
Content Mode
Left
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
Alpha
1
Background
Custom
Tint
Default
Drawing
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Spotlight
View Controller Scene
View Controller
View
Safe Area
Spotlight
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
View as: iPhone X (wC hR)
77%
Label
Text
Plain
Spotlight
Color
Dark Text Color (De...
Font
System Semibold 17.0
Dynamic Type
Automatically Adjusts Font
Alignment
Lines
1
Behavior
Enabled
Highlighted
Baseline
Align Baselines
Line Break
Truncate Tail
Autoshrink
Fixed Font Size
Tighten Letter Spacing
Highlighted
Default
Shadow
Default
Shadow Offset
0
-1
Width
Height
View
Content Mode
Left
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
Alpha
1
Background
Custom
Tint
Default
Drawing
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Spotlight
View Controller Scene
View Controller
View
Safe Area
Spotlight
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
View as: iPhone X (wC hR)
77%
Label
Text
Plain
Spotlight
Color
Dark Text Color (De...
Font
Body
Dynamic Type
Automatically Adjusts Font
Alignment
Lines
1
Behavior
Enabled
Highlighted
Baseline
Align Baselines
Line Break
Truncate Tail
Autoshrink
Fixed Font Size
Tighten Letter Spacing
Highlighted
Default
Shadow
Default
Shadow Offset
0
-1
Width
Height
View
Content Mode
Left
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
Alpha
1
Background
Custom
Tint
Default
Drawing
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Spotlight
View Controller Scene
View Controller
View
Safe Area
Spotlight
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
View as: iPhone X (wC hR)
77%
Label
Text
Plain
Spotlight
Color
Dark Text Color (De...
Font
Body
Dynamic Type
Automatically Adjusts Font
Alignment
Lines
1
Behavior
Enabled
Highlighted
Baseline
Align Baselines
Line Break
Truncate Tail
Autoshrink
Fixed Font Size
Tighten Letter Spacing
Highlighted
Default
Shadow
Default
Shadow Offset
0
-1
Width
Height
View
Content Mode
Left
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
Alpha
1
Background
Custom
Tint
Default
Drawing
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Spotlight
View Controller Scene
View Controller
View
Safe Area
Spotlight
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
View as: iPhone X (wC hR)
77%
Filter
Label
Text
Plain
Spotlight
Color
Dark Text Color (De...
Font
Body
Dynamic Type
Automatically Adjusts Font
Alignment
Lines
1
Behavior
Enabled
Highlighted
Baseline
Align Baselines
Line Break
Truncate Tail
Autoshrink
Fixed Font Size
Tighten Letter Spacing
Highlighted
Default
Shadow
Default
Shadow Offset
0
-1
Width
Height
View
Content Mode
Left
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
Alpha
1
Background
Custom
Tint
Default
Drawing
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Spotlight
View Controller Scene
View Controller
View
Safe Area
Spotlight
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
View as: iPhone X (wC hR)
77%
Label
Text
Plain
Spotlight
Color
Dark Text Color (De...
Font
Body
Dynamic Type
Automatically Adjusts Font
Alignment
Lines
1
Behavior
Enabled
Highlighted
Baseline
Align Baselines
Line Break
Truncate Tail
Autoshrink
Fixed Font Size
Tighten Letter Spacing
Highlighted
Default
Shadow
Default
Shadow Offset
0
-1
Width
Height
View
Content Mode
Left
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
Alpha
1
Background
Custom
Tint
Default
Drawing
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews

UIKit

```
@IBOutlet weak var spotlightLabel: UILabel!

override func viewDidLoad() {
    super.viewDidLoad()

    spotlightLabel.text = "Spotlight"

    spotlightLabel.textColor = .darkText
    spotlightLabel.font = UIFont.preferredFont(forTextStyle: .body)
    spotlightLabel.adjustsFontForContentSizeCategory = true
}
```

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
4
AppleNewsWWDC
AppleNewsWWDC
Main.storyboard
Main.stor...ard (Base)
View Con...ller Scene
View Controller
View
View Controller Scene
View Controller
View
Safe Area
Spotlight
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
View as: iPhone X (wC hR)
77%
Filter
View
Content Mode
Scale To Fill
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
Alpha
1
Background
Custom
Tint
Default
Drawing
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews
Stretching
0
0
X
Y
1
1
Width
Height

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
5
AppleNewsWWDC
Appl...WDC
Main...oard
Main...ase)
View...cene
View...roller
View
From the Apple News Editors
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
WEEKEND PURSUITS
Filter
View as: iPhone X (wC hR)
77%
Label
Text
Plain
From the Apple News Editors
Color
Dark Text Color (De...
Font
System 12.0
Dynamic Type
Automatically Adjusts Font
Alignment
Lines
1
Behavior
Enabled
Highlighted
Baseline
Align Baselines
Line Break
Truncate Tail
Autoshrink
Fixed Font Size
Tighten Letter Spacing
Highlighted
Default
Shadow
Default
Shadow Offset
0
-1
Width
Height
View
Content Mode
Left
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
Alpha
1
Background
Custom
Tint
Default
Drawing
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews

9:41
LTE
Search
Spotlight
WEEKEND PURSUITS
From the Apple News Editors

NSAttributedString

NSAttributedString

From the Apple News Editors

NSAttributedString

From the Apple News Editors

NSAttributedString

From the Apple News Editors

Font .preferredFont(forTextStyle: .body)

NSAttributedString

From the Apple News Editors

Font .preferredFont(forTextStyle: .body)

Color .darkText .white

UIKit

```
@IBOutlet weak var label: UILabel!

override func viewDidLoad() {
    super.viewDidLoad()

    let mutableAttrString = NSMutableAttributedString(string: label.text)

    let sourceRange = // Range of “Apple News Editors”
    mutableAttrString.addAttribute(.foregroundColor, value: UIColor.white, range: sourceRange)

    label.attributedText = mutableAttrString
}
```

UIKit

```
@IBOutlet weak var label: UILabel!

override func viewDidLoad() {
    super.viewDidLoad()

    let mutableAttrString = NSMutableAttributedString(string: label.text)

    let sourceRange = // Range of “Apple News Editors”
    mutableAttrString.addAttribute(.foregroundColor, value: UIColor.white, range: sourceRange)

    label.attributedText = mutableAttrString
}
```

UIKit

```
@IBOutlet weak var label: UILabel!

override func viewDidLoad() {
    super.viewDidLoad()

    let mutableAttrString = NSMutableAttributedString(string: label.text)

    let sourceRange = // Range of “Apple News Editors”
    mutableAttrString.addAttribute(.foregroundColor, value: UIColor.white, range: sourceRange)

    label.attributedText = mutableAttrString
}
```

9:41
LTE
Spotlight
WEEKEND PURSUITS
From the Apple News Editors

9:41
LTE
Search
Spotlight
WEEKEND PURSUITS
From the Apple News Editors

FEATURED

Students succeed with iPad and Mac

When you walk into Wilder Elementary School, just outside of Boise, Idaho, it's what

For You | Spotlight | Following | Search | Saved

FEATURED

Students succeed with iPad and Mac

When you walk into Wilder Elementary School, just outside of Boise, Idaho, it's what

For You | Spotlight | Following | Search | Saved

FEATURED

Copy | Select All | Look Up

Students succeed with iPad and Mac

When you walk into Wilder Elementary School, just outside of Boise, Idaho, it's what

For You | Spotlight | Following | Search | Saved

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
5
AppleNewsWWDC
AppleNewsWWDC
Main.storyboard
Main.stor...ard (Base)
View Con...ller Scene
View Controller
View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Constraints
First Responder
Exit
Storyboard Entry Point
Filter
Spotlight
WEEKEND PURS
Objects
View Controller - A controller that manages a view.
Storyboard Reference - Provides a placeholder for a view controller in an external storyboard.
Navigation Controller - A controller that manages navigation through a hierarchy of views.
Table View Controller - A controller that manages a table view.
Collection View Controller - A controller that manages a collection view.
Tab Bar Controller - A controller that manages a set of view controllers that represent tab bar items.
Split View Controller - A composite view controller that manages left and
View as: iPhone X (wC hR)
77%
View
Content Mode
Scale To Fill
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
1
Custom
Default
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews
0
0
X
Y
1
1
Width
Height

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
AppleNewsWWDC
AppleNewsWWDC
Main.storyboard
Main.stor...ard (Base)
View Con...ller Scene
View Controller
View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
WEEKEND PURS
Objects
View Controller - A controller that manages a view.
Storyboard Reference - Provides a placeholder for a view controller in an external storyboard.
Navigation Controller - A controller that manages navigation through a hierarchy of views.
Table View Controller - A controller that manages a table view.
Collection View Controller - A controller that manages a collection view.
Tab Bar Controller - A controller that manages a set of view controllers that represent tab bar items.
Split View Controller - A composite view controller that manages left and
View
Content Mode
Scale To Fill
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
1
Custom
Default
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews
0
0
X
Y
1
1
Width
Height
Filter
View as: iPhone X (wC hR)
77%

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | Today at 10:47 AM
AppleNewsWWDC
AppleNewsWWDC
Main.storyboard
Main.stor...ard (Base)
View Con...ller Scene
View Controller
View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
WEEKEND PURS
textview
Text View - Displays multiple lines of editable text and sends an action message to a target object when Return is tapped.
View
Content Mode
Scale To Fill
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
1
Custom
Default
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews
Stretching
0
0
X
Y
1
1
Width
Height
Filter
View as: iPhone X (wC hR)
77%

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
5
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Filter
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Lorem ipsum dolor sit er elit lamet, consectetaur cillium adipisicing pecu, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla
View as: iPhone X (wC hR)
77%
Text View
Text
Plain
Lorem ipsum dolor sit er elit lamet, consectetaur cillium adipisicing pecu, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Nam liber te conscient to factor tum poen legum odioque civiuda.
Color
Default
Font
System 14.0
Dynamic Type
Automatically Adjusts Font
Alignment
Behavior
Editable
Selectable
Data Detectors
Phone Number
Link
Address
Calendar Event
Shipment Tracking Number
Flight Number
Lookup Suggestion
Text Input Traits
Content Type
Unspecified
Capitalization
Sentences
Correction
Default
Smart Dashes
Default
Smart Insert
Default
Smart Quotes
Default
Spell Checking
Default

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
5
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Filter
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Students succeed with iPad and Mac
View as: iPhone X (wC hR)
77%
Text View
Text
Plain
Students succeed with iPad and Mac
Color
Default
Font
System 14.0
Dynamic Type
Automatically Adjusts Font
Alignment
Behavior
Editable
Selectable
Data Detectors
Phone Number
Link
Address
Calendar Event
Shipment Tracking Number
Flight Number
Lookup Suggestion
Text Input Traits
Content Type
Unspecified
Capitalization
Sentences
Correction
Default
Smart Dashes
Default
Smart Insert
Default
Smart Quotes
Default
Spell Checking
Default
Keyboard Type
Default
Keyboard Look
Default
Return Key
Default
Auto-enable Return Key
Secure Text Entry
Scroll View
Indicators
Default Style
Show Horizontal Indicator
Show Vertical Indicator
Scrolling
Scrolling Enabled
Paging Enabled

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
5
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Filter
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Students succeed with iPad and Mac
View as: iPhone X (wC hR)
77%
Text View
Text
Plain
Students succeed with iPad and Mac
Color
Default
Font
System 14.0
Dynamic Type
Automatically Adjusts Font
Alignment
Behavior
Editable
Selectable
Data Detectors
Phone Number
Link
Address
Calendar Event
Shipment Tracking Number
Flight Number
Lookup Suggestion
Text Input Traits
Content Type
Unspecified
Capitalization
Sentences
Correction
Default
Smart Dashes
Default
Smart Insert
Default
Smart Quotes
Default
Spell Checking
Default
Keyboard Type
Default
Keyboard Look
Default
Return Key
Default
Auto-enable Return Key
Secure Text Entry
Scroll View
Indicators
Default Style
Show Horizontal Indicator
Show Vertical Indicator
Scrolling
Scrolling Enabled
Paging Enabled

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
5
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Filter
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Students succeed with iPad and Mac
View as: iPhone X (wC hR)
77%
Text View
Text
Plain
Students succeed with iPad and Mac
Color
Default
Font
Georgia Bold 37.0
Dynamic Type
Automatically Adjusts Font
Alignment
Behavior
Editable
Selectable
Data Detectors
Phone Number
Link
Address
Calendar Event
Shipment Tracking Number
Flight Number
Lookup Suggestion
Text Input Traits
Content Type
Unspecified
Capitalization
Sentences
Correction
Default
Smart Dashes
Default
Smart Insert
Default
Smart Quotes
Default
Spell Checking
Default
Keyboard Type
Default
Keyboard Look
Default
Return Key
Default
Auto-enable Return Key
Secure Text Entry
Scroll View
Indicators
Default Style
Show Horizontal Indicator
Show Vertical Indicator
Scrolling
Scrolling Enabled
Paging Enabled

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Students succeed with iPad and Mac
View as: iPhone X (wC hR)
77%
Text View
Text
Plain
Students succeed with iPad and Mac
Color
Default
Font
Georgia Bold 37.0
Dynamic Type
Automatically Adjusts Font
Alignment
Behavior
Editable
Selectable
Data Detectors
Phone Number
Link
Address
Calendar Event
Shipment Tracking Number
Flight Number
Lookup Suggestion
Text Input Traits
Content Type
Unspecified
Capitalization
Sentences
Correction
Default
Smart Dashes
Default
Smart Insert
Default
Smart Quotes
Default
Spell Checking
Default
Keyboard Type
Default
Keyboard Look
Default
Return Key
Default
Auto-enable Return Key
Secure Text Entry
Scroll View
Indicators
Default Style
Show Horizontal Indicator
Show Vertical Indicator
Scrolling
Scrolling Enabled
Paging Enabled

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Students succeed with iPad and Mac
View as: iPhone X (wC hR)
77%
Text View
Text
Plain
Students succeed with iPad and Mac
Color
Default
Font
Georgia Bold 37.0
Dynamic Type
Automatically Adjusts Font
Alignment
Behavior
Editable
Selectable
Data Detectors
Phone Number
Link
Address
Calendar Event
Shipment Tracking Number
Flight Number
Lookup Suggestion
Text Input Traits
Content Type
Unspecified
Capitalization
Sentences
Correction
Default
Smart Dashes
Default
Smart Insert
Default
Smart Quotes
Default
Spell Checking
Default
Keyboard Type
Default
Keyboard Look
Default
Return Key
Default
Auto-enable Return Key
Secure Text Entry
Scroll View
Indicators
Default Style
Show Horizontal Indicator
Show Vertical Indicator
Scrolling
Scrolling Enabled
Paging Enabled

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Students succeed with iPad and Mac
View as: iPhone X (wC hR)
77%
Text View
Text
Plain
Students succeed with iPad and Mac
Color
Default
Font
Georgia Bold 37.0
Dynamic Type
Automatically Adjusts Font
Alignment
Behavior
Editable
Selectable
Data Detectors
Phone Number
Link
Address
Calendar Event
Shipment Tracking Number
Flight Number
Lookup Suggestion
Text Input Traits
Content Type
Unspecified
Capitalization
Sentences
Correction
Default
Smart Dashes
Default
Smart Insert
Default
Smart Quotes
Default
Spell Checking
Default
Keyboard Type
Default
Keyboard Look
Default
Return Key
Default
Auto-enable Return Key
Secure Text Entry
Scroll View
Indicators
Default Style
Show Horizontal Indicator
Show Vertical Indicator
Scrolling
Scrolling Enabled
Paging Enabled

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Students succeed with iPad and Mac
View as: iPhone X (wC hR)
77%
Text View
Text
Plain
Students succeed with iPad and Mac
Color
Default
Font
Georgia Bold 37.0
Dynamic Type
Automatically Adjusts Font
Alignment
Behavior
Editable
Selectable
Data Detectors
Phone Number
Link
Address
Calendar Event
Shipment Tracking Number
Flight Number
Lookup Suggestion
Text Input Traits
Content Type
Unspecified
Capitalization
Sentences
Correction
Default
Smart Dashes
Default
Smart Insert
Default
Smart Quotes
Default
Spell Checking
Default
Keyboard Type
Default
Keyboard Look
Default
Return Key
Default
Auto-enable Return Key
Secure Text Entry
Scroll View
Indicators
Default Style
Show Horizontal Indicator
Show Vertical Indicator
Scrolling
Scrolling Enabled
Paging Enabled

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Students succeed with iPad and Mac
View as: iPhone X (wC hR)
77%
Smart Insert Default
Smart Quotes Default
Spell Checking Default
Keyboard Type Default
Keyboard Look Default
Return Key Default
Auto-enable Return Key
Secure Text Entry
Scroll View
Indicators Default Style
Show Horizontal Indicator
Show Vertical Indicator
Scrolling Scrolling Enabled
Paging Enabled
Direction Lock Enabled
Bounce Bounce On Scroll
Bounce On Zoom
Bounce Horizontally
Bounce Vertically
Zoom 1 Min 1 Max
Content Touch Delay Touch Down
Can Cancel On Scroll
Keyboard Do not dismiss
View
Content Mode Scale To Fill
Semantic Unspecified
Tag 0
Interaction User Interaction Enabled
Multiple Touch
Alpha 1
Background Default
Tint Default

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Students succeed with iPad and Mac
View as: iPhone X (wC hR)
77%
Smart Insert Default
Smart Quotes Default
Spell Checking Default
Keyboard Type Default
Keyboard Look Default
Return Key Default
Auto-enable Return Key
Secure Text Entry
Scroll View
Indicators Default Style
Show Horizontal Indicator
Show Vertical Indicator
Scrolling Scrolling Enabled
Paging Enabled
Direction Lock Enabled
Bounce Bounce On Scroll
Bounce On Zoom
Bounce Horizontally
Bounce Vertically
Zoom 1 Min 1 Max
Content Touch Delay Touch Down
Can Cancel On Scroll
Keyboard Do not dismiss
View
Content Mode Scale To Fill
Semantic Unspecified
Tag 0
Interaction User Interaction Enabled
Multiple Touch
Alpha 1
Background Default
Tint Default

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Students succeed with iPad and Mac
View as: iPhone X (wC hR)
77%
Smart Insert Default
Smart Quotes Default
Spell Checking Default
Keyboard Type Default
Keyboard Look Default
Return Key Default
Auto-enable Return Key
Secure Text Entry
Scroll View
Indicators Default Style
Show Horizontal Indicator
Show Vertical Indicator
Scrolling Scrolling Enabled
Paging Enabled
Direction Lock Enabled
Bounce Bounce On Scroll
Bounce On Zoom
Bounce Horizontally
Bounce Vertically
Zoom 1 Min 1 Max
Content Touch Delay Touch Down
Can Cancel On Scroll
Keyboard Do not dismiss
View
Content Mode Scale To Fill
Semantic Unspecified
Tag 0
Interaction User Interaction Enabled
Multiple Touch
Alpha 1
Background White Color
Tint Default

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Students succeed with iPad and Mac
View as: iPhone X (wC hR)
77%
Smart Insert Default
Smart Quotes Default
Spell Checking Default
Keyboard Type Default
Keyboard Look Default
Return Key Default
Auto-enable Return Key
Secure Text Entry
Scroll View
Indicators Default Style
Show Horizontal Indicator
Show Vertical Indicator
Scrolling Scrolling Enabled
Paging Enabled
Direction Lock Enabled
Bounce Bounce On Scroll
Bounce On Zoom
Bounce Horizontally
Bounce Vertically
Zoom 1 Min 1 Max
Content Touch Delay Touch Down
Can Cancel On Scroll
Keyboard Do not dismiss
View
Content Mode Scale To Fill
Semantic Unspecified
Tag 0
Interaction User Interaction Enabled
Multiple Touch
Alpha 1
Background White Color
Tint Default

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Students succeed with iPad and Mac
View as: iPhone X (wC hR)
77%
Smart Insert
Default
Smart Quotes
Default
Spell Checking
Default
Keyboard Type
Default
Keyboard Look
Default
Return Key
Default
Auto-enable Return Key
Secure Text Entry
Scroll View
Indicators
Default Style
Show Horizontal Indicator
Show Vertical Indicator
Scrolling
Scrolling Enabled
Paging Enabled
Direction Lock Enabled
Bounce
Bounce On Scroll
Bounce On Zoom
Bounce Horizontally
Bounce Vertically
Zoom
1
Min
1
Max
Content Touch
Delay Touch Down
Can Cancel On Scroll
Keyboard
Do not dismiss
View
Content Mode
Scale To Fill
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
Alpha
1
Background
White Color
Tint
Default

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
5
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Filter
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Students succeed with iPad and Mac
View as: iPhone X (wC hR)
77%
Indicators
Default Style
Show Horizontal Indicator
Show Vertical Indicator
Scrolling
Scrolling Enabled
Paging Enabled
Direction Lock Enabled
Bounce
Bounce On Scroll
Bounce On Zoom
Bounce Horizontally
Bounce Vertically
Zoom
1
Min
1
Max
Content Touch
Delay Touch Down
Can Cancel On Scroll
Keyboard
Do not dismiss
View
Content Mode
Scale To Fill
Semantic
Unspecified
Tag
0
Interaction
User Interaction Enabled
Multiple Touch
Alpha
1
Background
White Color
Tint
Default
Drawing
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews
Stretching
0
X
0
Y
1
Width
1
Height
Installed

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Students succeed with iPad and Mac
View as: iPhone X (wC hR)
77%
Indicators
Default Style
Show Horizontal Indicator
Show Vertical Indicator
Scrolling
Scrolling Enabled
Paging Enabled
Direction Lock Enabled
Bounce
Bounce On Scroll
Bounce On Zoom
Bounce Horizontally
Bounce Vertically
Zoom
Min
Max
Content Touch
Delay Touch Down
Can Cancel On Scroll
Keyboard
Do not dismiss
View
Content Mode
Scale To Fill
Semantic
Unspecified
Tag
Interaction
User Interaction Enabled
Multiple Touch
Alpha
Background
Default
Tint
Default
Drawing
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews
Stretching
X
Y
Width
Height
Installed

AppleN...WWDC
iPhone 8 Plus
AppleNewsWWDC | Build Succeeded | 5/29/18 at 9:53 PM
AppleNewsWWDC
Apple...sWWDC
Main.s...yboard
Main.s...(Base)
View C...Scene
View Controller
View
Text View
View Controller Scene
View Controller
View
Safe Area
Spotlight
WEEKEND PURSUITS
Source Label
Text View
Constraints
First Responder
Exit
Storyboard Entry Point
Spotlight
WEEKEND PURSUITS
From the Apple News Editors
Students succeed with iPad and Mac
View as: iPhone X (wC hR)
77%
Indicators
Default Style
Show Horizontal Indicator
Show Vertical Indicator
Scrolling
Scrolling Enabled
Paging Enabled
Direction Lock Enabled
Bounce
Bounce On Scroll
Bounce On Zoom
Bounce Horizontally
Bounce Vertically
Zoom
Min
Max
Content Touch
Delay Touch Down
Can Cancel On Scroll
Keyboard
Do not dismiss
View
Content Mode
Scale To Fill
Semantic
Unspecified
Tag
Interaction
User Interaction Enabled
Multiple Touch
Alpha
Background
Default
Tint
Default
Drawing
Opaque
Hidden
Clears Graphics Context
Clip to Bounds
Autoresize Subviews
Stretching
X
Y
Width
Height
Installed

UIKit

```
@IBOutlet weak var headlineTextView: UITextView!

override func viewDidLoad() {
    super.viewDidLoad()

    headlineTextView.text = "Students succeed with iPad and Mac"
    headlineTextView.textColor = .darkText
    headlineTextView.font = UIFont(name: "Georgia", size: 37)
    headlineTextView.backgroundColor = nil
    headlineTextView.isEditable = false
    headlineTextView.isScrollEnabled = false
}
```

UIKit

```
@IBOutlet weak var headlineTextView: UITextView!

override func viewDidLoad() {
    super.viewDidLoad()

    headlineTextView.text = "Students succeed with iPad and Mac"
    headlineTextView.textColor = .darkText
    headlineTextView.font = UIFont(name: "Georgia", size: 37)
    headlineTextView.backgroundColor = nil
    headlineTextView.isEditable = false
    headlineTextView.isScrollEnabled = false
}
```

Choosing the right configuration.

TextEdit on macOS

Document
Helvetica
Regular
12
1.0
0
1
2
3
4
5
6
7
Here's to the crazy ones. The misfits. The rebels. The trouble-makers. The round pegs in the square holes. The ones who see things differently. They're not fond of rules, and they have no respect for the status-quo. You can quote them, disagree with them, glorify, or vilify them. But the only thing you can't do is ignore them. Because they change things. They push the human race forward. And while some may see them as the crazy ones, we see genius. Because the people who are crazy enough to think they can change the world, are the ones who do.

Document

Helvetica Regular 12 1.0

Here's to the crazy ones. The misfits. The rebels. The trouble-makers. The round pegs in the square holes. The ones who see things differently. They're not fond of rules, and they have no respect for the status-quo. You can quote them, disagree with them, glorify, or vilify them. But the only thing you can't do is ignore them. Because they change things. They push the human race forward. And while some may see them as the crazy ones, we see genius. Because the people who are crazy enough to think they can change the world, are the ones who do.

Document

Helvetica Regular 12 1.0

Here's to the crazy ones. The misfits. The rebels. The trouble-makers. The round pegs in the square holes. The ones who see things differently. They're not fond of rules, and they have no respect for the status-quo. You can quote them, disagree with them, glorify, or vilify them. But the only thing you can't do is ignore them. Because they change things. They push the human race forward. And while some may see them as the crazy ones, we see genius. Because the people who are crazy enough to think they can change the world, are the ones who do.

Document
Helvetica
Regular
12
1.0
0
1
2
3
4
5
6
7
Here's to the crazy ones. The misfits. The rebels. The trouble-makers. The round pegs in the square holes. The ones who see things differently. They're not fond of rules, and they have no respect for the status-quo. You can quote them, disagree with them, glorify, or vilify them. But the only thing you can't do is ignore them. Because they change things. They push the human race forward. And while some may see them as the crazy ones, we see genius. Because the people who are crazy enough to think they can change the world, are the ones who do.

Document
Helvetica
Regular
12
1.0
Here's to the crazy ones. The misfits. The rebels. The trouble-makers. The round pegs in the square holes. The ones who see things differently. They're not fond of rules, and they have no respect for the status-quo. You can quote them, disagree with them, glorify, or vilify them. But the only thing you can't do is ignore them. Because they change things. They push the human race forward. And while some may see them as the crazy ones, we see genius. Because the people who are crazy enough to think they can change the world, are the ones who do.
Text Storage
Layout Manager
Text Container
weak
Text View

Document

Helvetica | Regular | 12 | 1.0

Here's to the crazy ones. The misfits. The rebels. The trouble-makers. The round pegs in the square holes. The ones who see things differently. They're not fond of rules, and they have no respect for the status-quo. You can quote them, disagree with them, glorify, or vilify them. But the only thing you can't do is ignore them. Because they change things. They push the human race forward. And while some may see them as the crazy ones, we see genius. Because the people who are crazy enough to think they can change the world, are the ones who do.

Document
Helvetica
Regular
12
1.0
Here's to the crazy ones. The misfits. The rebels. The trouble-makers. The round pegs in the square holes. The ones who see things differently. They're not fond of rules, and they have no respect for the status-quo. You can quote them, disagree with them, glorify, or vilify them. But the only thing you can't do is ignore them. Because they change things. They push the human race forward. And while some may see them as the crazy ones, we see genius. Because the people who are crazy enough to think they can change the world, are the ones who do.
Table
Rows
2
Merge Cells
Columns
3
Split Cells
Nest Table
Alignment
px
Cell Border
1
px
Cell Background
None

Document

Helvetica | Regular | 12 | 1.0

Here's to the crazy ones. The misfits. The rebels. The trouble-makers. The round pegs in the square holes. The ones who see things differently. They're not fond of rules, and they have no respect for the status-quo. You can quote them, disagree with them, glorify, or vilify them. But the only thing you can't do is ignore them. Because they change things. They push the human race forward. And while some may see them as the crazy ones, we see genius. Because the people who are crazy enough to think they can change the world, are the ones who do.

Etiam mollis dolor sed cursus dapibus. Phasellus venenatis leo lectus, a eleifend tortor lobortis dapibus. Mauris vulputate volutpat euismod. Fusce suscipit, urna ut consectetur volutpat, mi purus bibendum diam, nec eleifend risus risus quis nisl. Suspendisse leo ligula, pulvinar eget magna a, malesuada rhoncus risus. Suspendisse in diam ornare, elementum felis eget, consequat felis. Duis posuere quam sit amet bibendum hendrerit. Nulla ultricies nisi a enim vestibulum, vehicula consequat est placerat. Aenean posuere nisi et ipsum vestibulum porta. Mauris viverra tincidunt felis, a tempor sem. Praesent rutrum ullamcorper lectus sed aliquet. Fusce hendrerit enim elit, sit amet euismod sapien dictum id. Etiam varius, tortor non auctor rutrum, lorem leo mattis neque, sit amet facilisis nisi dui quis tellus. Integer non odio sagittis, consectetur enim eu, efficitur tortor.

Vivamus non efficitur neque. Sed tristique pretium auctor. Ut imperdiet sem id lorem hendrerit, id sollicitudin ex tempus. Suspendisse malesuada quis elit maximus semper. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin porta at lacus dignissim ullamcorper. Curabitur ullamcorper magna id nulla congue, ut eleifend purus maximus. Nulla ut porta massa. Aliquam volutpat velit diam, ac tincidunt arcu commodo vel.

Nullam porta metus est, id dapibus dolor feugiat sit amet. Donec ut eros sem. Suspendisse a scelerisque nisi. Quisque sodales interdum venenatis. Ut sit amet felis vel augue congue pellentesque eu nec risus. Sed vel nisl ut quam fringilla malesuada sed quis magna. Sed eget sapien nec purus

Document — Edited

Helvetica | Regular | 12 | 1.0

Here's to the crazy ones. The misfits. The rebels. The trouble-makers. The round pegs in the square holes. The ones who see things differently. They're not fond of rules, and they have no respect for the status-quo. You can quote them, disagree with them, glorify, or vilify them. But the only thing you can't do is ignore them. Because they change things. They push the human race forward. And while some may see them as the crazy ones, we see genius. Because the people who are crazy enough to think they can change the world, are the ones who do.

Etiam mollis dolor sed cursus dapibus. Phasellus venenatis leo lectus, a eleifend tortor lobortis dapibus. Mauris vulputate volutpat euismod. Fusce suscipit, urna ut consectetur volutpat, mi purus bibendum diam, nec eleifend risus risus quis nisl. Suspendisse leo ligula, pulvinar eget magna a, malesuada rhoncus risus. Suspendisse in diam ornare, elementum felis eget, consequat felis. Duis posuere quam sit amet bibendum hendrerit. Nulla ultricies nisi a enim vestibulum, vehicula consequat est placerat. Aenean posuere nisi et ipsum vestibulum porta. Mauris viverra tincidunt felis, a tempor sem. Praesent rutrum ullamcorper lectus sed aliquet. Fusce hendrerit enim elit, sit amet euismod sapien dictum id. Etiam varius, tortor non auctor rutrum, lorem leo mattis neque, sit amet facilisis nisi dui quis tellus. Integer non odio sagittis, consectetur enim eu, efficitur tortor.

Vivamus non efficitur neque. Sed tristique pretium auctor. Ut imperdiet sem id lorem

Document — Edited

Helvetica | Regular | 12 | 1.0

vel molestie eu, dignissim nec erat. Morbi finibus luctus feugiat.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut auctor, massa eget hendrerit iaculis, mauris tortor rutrum lorem, in vehicula odio libero quis felis. Etiam nisl elit, tincidunt nec mollis vel, luctus ut diam. Sed convallis lacus sit amet venenatis lacinia. Aliquam quam lectus, convallis quis neque in, lobortis suscipit tortor. Mauris condimentum laoreet gravida. Duis accumsan orci vel luctus dapibus. Curabitur ullamcorper sapien quis quam mattis facilisis. Vestibulum in tellus dolor. Suspendisse ac venenatis lacus. Morbi id metus odio. Maecenas a viverra ante. Mauris a urna eget lacus tincidunt blandit. Maecenas vitae fringilla sem, at pretium quam. Cras sit amet

suscipit augue, et semper mauris.

Pellentesque id quam turpis. Fusce et massa nec massa ultricies euismod. Donec id est suscipit, semper metus sed, pulvinar magna. Morbi blandit ex vel enim sodales, in hendrerit justo consequat. Aenean efficitur eros in vulputate placerat. Fusce eu nibh mi. In mollis ante a libero luctus pellentesque. Etiam nunc neque, sagittis a nisl sed, tristique pharetra elit.

Document — Edited
Helvetica
Regular
12
1.0
Text Storage
Layout Manager
Text Container
Text Container
weak
weak
Text View
Text View

TextEdit

Sample Code

Choosing the right customization approach.

Our Journal App on macOS

OurJournal
My Mac
OurJournal: Ready | Today at 9:26 AM
OurJournal
OurJournal
Main.storyboard
Main.storyboard (Base)
No Selection
Application Scene
Window Controller Scene
View Controller Scene
View Controller
View
Wednesday, June 6, 2018
First Responder
Wednesday, June 6, 2018
No Selection
Filter

OurJournal 〉 My Mac
OurJournal: Ready | Today at 12:02 AM
OurJournal 〉 OurJournal 〉 Main....yboard 〉 Main....(Base) 〉 View...r Scene 〉 View...ntroller 〉 View 〉 Wednesday, June 6, 2018
Application Scene
Window Controller Scene
Window Controller
Window
First Responder
Storyboard Entry Point
Relationship "window content" to "...
View Controller Scene
View Controller
View
Wednesday, June 6, 2018
Wednesday, June 6, 2018
First Responder
Filter
Wednesday, June 6, 2018
Text Field
Title Wednesday, June 6, 2018
Placeholder Placeholder String
Alignment
Border
Display Draws Background
Text Color Label Color (Default)
Background Text Background C...
Font System Regular
Layout Truncates
Uses Single Line Mode
Action Send On Enter Only
Behavior None
Allows Rich Text Undo
Only Roman Characters
Touch Bar Auto Text Completion
Allows Character Picker
Control
Line Break Clip
Truncates Last Visible Line
State Enabled
Continuous
Refuses First Responder
Tooltips Allows Expansion Tooltips
Text Direction Natural
Layout Left To Right
Mirror Automatically
View
Tag 0
Focus Ring Default
Drawing Hidden
Can Draw Concurrently
Autoresizing Autoresizes Subviews

OurJournal
My Mac
OurJournal: Ready | Today at 12:02 AM
OurJournal
OurJournal
Main....yboard
Main....(Base)
View...r Scene
View...ntroller
View
Wednesday, June 6, 2018
Application Scene
Window Controller Scene
Window Controller
Window
First Responder
Storyboard Entry Point
Relationship "window content" to "...
View Controller Scene
View Controller
View
Wednesday, June 6, 2018
Wednesday, June 6, 2018
First Responder
Wednesday, June 6, 2018
Text Field
Title
Wednesday, June 6, 2018
Placeholder
Placeholder String
Alignment
Border
Display
Draws Background
Text Color
Label Color (Default)
Background
Text Background C...
Font
System Regular
Layout
Truncates
Uses Single Line Mode
Action
Send On Enter Only
Behavior
None
Allows
Rich Text
Undo
Only Roman Characters
Touch Bar
Auto Text Completion
Allows Character Picker
Control
Line Break
Clip
Truncates Last Visible Line
State
Enabled
Continuous
Refuses First Responder
Tooltips
Allows Expansion Tooltips
Text Direction
Natural
Layout
Left To Right
Mirror
Automatically
View
Tag
0
Focus Ring
Default
Drawing
Hidden
Can Draw Concurrently
Autoresizing
Autoresizes Subviews
Filter

OurJournal
My Mac
OurJournal: Ready | Today at 12:03 AM
OurJournal
OurJournal
Main....yboard
Main....(Base)
View...r Scene
View...ntroller
View
Wednesday, June 6, 2018
Application Scene
Window Controller Scene
Window Controller
Window
First Responder
Storyboard Entry Point
Relationship "window content" to "...
View Controller Scene
View Controller
View
Wednesday, June 6, 2018
Wednesday, June 6, 2018
First Responder
Wednesday, June 6, 2018
Text Field
Title
Wednesday, June 6, 2018
Placeholder
Placeholder String
Alignment
Border
Display
Draws Background
Text Color
Label Color (Default)
Background
Text Background C...
Font
System Regular
Layout
Truncates
Selectable
Editable
None
Allows
Rich Text
Undo
Only Roman Characters
Touch Bar
Auto Text Completion
Allows Character Picker
Control
Line Break
Clip
Truncates Last Visible Line
State
Enabled
Continuous
Refuses First Responder
Tooltips
Allows Expansion Tooltips
Text Direction
Natural
Layout
Left To Right
Mirror
Automatically
View
Tag
0
Focus Ring
Default
Drawing
Hidden
Can Draw Concurrently
Autoresizing
Autoresizes Subviews
Filter

OurJournal 〉 My Mac
OurJournal: Ready | Today at 12:03 AM
OurJournal 〉 OurJournal 〉 Main....yboard 〉 Main....(Base) 〉 View...r Scene 〉 View...ntroller 〉 View 〉 Wednesday, June 6, 2018
Application Scene
Window Controller Scene
Window Controller
Window
First Responder
Storyboard Entry Point
Relationship "window content" to "...
View Controller Scene
View Controller
View
Wednesday, June 6, 2018
Wednesday, June 6, 2018
First Responder
Filter
Wednesday, June 6, 2018
Text Field
Title Wednesday, June 6, 2018
Placeholder Placeholder String
Alignment
Border
Display Draws Background
Text Color Label Color (Default)
Background Text Background C...
Font System Regular
Layout Truncates
Selectable
Editable
None
Only Roman Characters
Touch Bar Auto Text Completion
Allows Character Picker
Control
Line Break Clip
Truncates Last Visible Line
State Enabled
Continuous
Refuses First Responder
Tooltips Allows Expansion Tooltips
Text Direction Natural
Layout Left To Right
Mirror Automatically
View
Tag 0
Focus Ring Default
Drawing Hidden
Can Draw Concurrently
Autoresizing Autoresizes Subviews

OurJournal 〉 My Mac
OurJournal: Ready | Today at 9:26 AM
OurJournal 〉 OurJournal 〉 Main.storyboard 〉 Main.storyboard (Base) 〉 No Selection
Application Scene
Window Controller Scene
View Controller Scene
View Controller
View
Wednesday, June 6, 2018
First Responder
textview
Scrollable Text View - Displays and manipulates text laid out in an area defined by an NSTextContainer and adds many features to those defined by its superclass, NSText.
Rich Document Content Text View - Displays and manipulates text laid out in an area defined by an NSTextContainer and adds many features to those defined by its superclass, NSText.
Plain Document Content Text View - Displays and manipulates text laid out in an area defined by an NSTextContainer and adds many features to those defined by its superclass, NSText.
NSTouchBar Character Picker - Character picker for use in a bar.
Wednesd
No Selection
Filter

OurJournal 〉 My Mac
OurJournal | Build OurJournal: Succeeded | 5/30/18 at 12:27 PM
OurJournal 〉 Our...urnal 〉 Mai...oard 〉 Mai...ase) 〉 Vie...cene 〉 Vie...troller 〉 View 〉 Scr...View 〉 Clip View 〉 Text View
Application Scene
Window Controller Scene
View Controller Scene
View Controller
View
Wednesday, June 6, 2018
Scroll View - Text View
Clip View
Text View
Scroller
Constraints
First Responder
Filter
Wednesday, June 6, 2018
Text View
Text
No Font
Text
Text Direction Natural
Orientation Horizontal
Selection Character
Behavior Editable Selectable
Field Editor
Allows Rich Text Undo
Graphics
Image Editing
Non-contiguous Layout
Background Color Change
Only Roman Characters
Uses Font Panel Ruler
Inspector Bar
Find None
Incremental Searching
Display Draws Background
Text Color Text Color (Default)
Background Custom
Insertion Control Text Color
Linguistics Continuous Spell Checking
Automatic Spelling Correcti...
Grammar Checking
Substitution Smart Links
Smart Quotes
Smart Dashes
Data Detectors
Text Replacement
Editing Smart Insert and Delete
Touch Bar Auto Text Completion
Allows Character Picker
View

OurJournal
My Mac
OurJournal | Build OurJournal: Succeeded | 5/30/18 at 12:27 PM
Application Scene
Window Controller Scene
View Controller Scene
View Controller
View
Wednesday, June 6, 2018
Scroll View - Text View
Clip View
Text View
Scroller
Constraints
First Responder
Wednesday, June 6, 2018
Text View
Text
Text Direction
Natural
Orientation
Horizontal
Behavior
Editable
Selectable
Allows
Rich Text
Undo
Image Editing
Non-contiguous Layout
Background Color Change
Uses
Font Panel
Ruler
Find
None
Incremental Searching
Display
Draws Background
Text Color
Text Color (Default)
Background
Custom
Insertion
Control Text Color
Linguistics
Continuous Spell Checking
Automatic Spelling Correcti...
Grammar Checking
Substitution
Smart Links
Smart Quotes
Smart Dashes
Data Detectors
Text Replacement
Editing
Smart Insert and Delete
Touch Bar
Auto Text Completion
Allows Character Picker
View

OurJournal 〉 My Mac
OurJournal | Build OurJournal: Succeeded | 5/30/18 at 12:27 PM
OurJournal 〉 OurJournal 〉 Main.storyboard 〉 Main.sto...rd (Base) 〉 View Co...ler Scene 〉 View Controller 〉 View 〉 Word Count:
Application Scene
Window Controller Scene
View Controller Scene
View Controller
View
Wednesday, June 6, 2018
Scroll View - Text View
Clip View
Text View
Scroller
Word Count:
0
Constraints
First Responder
Filter
Wednesday, June 6, 2018
Word Count: 0
Text Field
Title Word Count:
Placeholder Placeholder String
Alignment
Border
Display Draws Background
Text Color Label Color (Default)
Background Text Background C...
Font System Regular
Layout Truncates
Uses Single Line Mode
Action Send On Enter Only
Behavior None
Allows Rich Text Undo
Only Roman Characters
Touch Bar Auto Text Completion
Allows Character Picker
Control
Line Break Clip
Truncates Last Visible Line
State Enabled
Continuous
Refuses First Responder
Tooltips Allows Expansion Tooltips
Text Direction Natural
Layout Left To Right
Mirror Automatically
View
Tag 0
Focus Ring Default
Drawing Hidden
Can Draw Concurrently
Autoresizing Autoresizes Subviews

OurJournal 〉 My Mac
OurJournal | Build OurJournal: Succeeded | 5/30/18 at 12:27 PM
OurJournal 〉 OurJournal 〉 Main.storyboard 〉 Main.sto...rd (Base) 〉 View Co...ler Scene 〉 View Controller 〉 View 〉 Word Count:
Application Scene
Window Controller Scene
View Controller Scene
View Controller
View
Wednesday, June 6, 2018
Scroll View - Text View
Clip View
Text View
Scroller
Word Count:
0
Constraints
First Responder
Filter
Wednesday, June 6, 2018
Word Count: 0
Text Field
Title Word Count:
Placeholder Placeholder String
Alignment
Border
Display Draws Background
Text Color Label Color (Default)
Background Text Background C...
Font System Regular
Layout Truncates
Uses Single Line Mode
Action Send On Enter Only
Behavior None
Allows Rich Text Undo
Only Roman Characters
Touch Bar Auto Text Completion
Allows Character Picker
Control
Line Break Clip
Truncates Last Visible Line
State Enabled
Continuous
Refuses First Responder
Tooltips Allows Expansion Tooltips
Text Direction Natural
Layout Left To Right
Mirror Automatically
View
Tag 0
Focus Ring Default
Drawing Hidden
Can Draw Concurrently
Autoresizing Autoresizes Subviews

Our Journal
Wednesday, June 6, 2018
Word Count: 0

Our Journal
Wednesday, June 6, 2018
Today
Word Count: 0

Delegation

NSTextStorageDelegate

```
func textStorage(_:didProcessEditing:range:changeInLength:)
```

Notification

NSTextStorage

```
didProcessEditingNotification
```

Subclassing

NSTextView

```
override func textDidChange()
```

Delegation

NSTextStorageDelegate

```
func textStorage(_:didProcessEditing:range:changeInLength:)
```

Notification

NSTextStorage

```
didProcessEditingNotification
```

Subclassing

NSTextView

```
override func textDidChange()
```

AppKit

```
func registerObservers() {
    NotificationCenter.default.addObserver(self,
                                           selector: #selector(updateWordCount),
                                           name: NSTextStorage.didProcessEditingNotification,
                                           object: nil)
}

@objc func updateWordCount() {
    guard let textStorage = textView.textStorage else { fatalError() }

    let wordCount = textStorage.words.count
    wordCountLabel.stringValue = String(wordCount)
}
```

AppKit

```
func registerObservers() {
    NotificationCenter.default.addObserver(self,
                                           selector: #selector(updateWordCount),
                                           name: NSTextStorage.didProcessEditingNotification,
                                           object: nil)
}

@objc func updateWordCount() {
    guard let textStorage = textView.textStorage else { fatalError() }

    let wordCount = textStorage.words.count
    wordCountLabel.stringValue = String(wordCount)
}
```

AppKit

```
func registerObservers() {
    NotificationCenter.default.addObserver(self,
                                           selector: #selector(updateWordCount),
                                           name: NSTextStorage.didProcessEditingNotification,
                                           object: nil)
}

@objc func updateWordCount() {
    guard let textStorage = textView.textStorage else { fatalError() }

    let wordCount = textStorage.words.count
    wordCountLabel.stringValue = String(wordCount)
}
```

Our Journal
Wednesday, June 6, 2018
Word Count: 0

Our Journal

Wednesday, June 6, 2018

Today I celebrated the crazy ones. The misfits. The rebels. The troublemakers.

Word Count: 12

Our Journal
Wednesday, June 6, 2018
Today I celebrated the crazy ones. The misfits. The rebels. The troublemakers.
Word Count: 12

Our Journal

Wednesday, June 6, 2018

Today I celebrated the crazy ones. The misfits. **The rebels.** The troublemakers.

Word Count: 12

Our Journal

Wednesday, June 6, 2018

Today I celebrated the crazy ones. The misfits. The rebels. The troublemakers.

Word Count: 12

Our Journal

Wednesday, June 6, 2018

Today I celebrated the crazy ones. The misfits. **The rebels.* The troublemakers.

Word Count: 12

Delegation

NSTextStorageDelegate

```
func textStorage(_:didProcessEditing:range:changeInLength:)
```

Notification

NSTextStorage

```
didProcessEditingNotification
```

Subclassing

NSTextStorage

```
override func replaceCharacters(in:with:)
```

Delegation

NSTextStorageDelegate

```
func textStorage(_:didProcessEditing:range:changeInLength:)
```

Notification

NSTextStorage

```
didProcessEditingNotification
```

Subclassing

NSTextStorage

```
override func replaceCharacters(in:with:)
```

AppKit

```
func textStorage(_ textStorage: NSTextStorage,
                 didProcessEditing editedMask: NSTextStorageEditActions,
                 range editedRange: NSRange,
                 changeInLength delta: Int) {

    let boldRange = // Range including start and end **
    let start = boldRange.location

    guard let font = textStorage.attribute(.font, at: start, effectiveRange: nil) as? NSFont
        else { fatalError() }
    let boldFont = NSFontManager.shared.convert(font, toHaveTrait: .boldFontMask)

    textStorage.addAttribute(.font, value: boldFont, range: boldRange)
}
```

AppKit

```
func textStorage(_ textStorage: NSTextStorage,
                 didProcessEditing editedMask: NSTextStorageEditActions,
                 range editedRange: NSRange,
                 changeInLength delta: Int) {

    let boldRange = // Range including start and end **
    let start = boldRange.location

    guard let font = textStorage.attribute(.font, at: start, effectiveRange: nil) as? NSFont
        else { fatalError() }
    let boldFont = NSFontManager.shared.convert(font, toHaveTrait: .boldFontMask)

    textStorage.addAttribute(.font, value: boldFont, range: boldRange)
}
```

AppKit

```
func textStorage(_ textStorage: NSTextStorage,
                 didProcessEditing editedMask: NSTextStorageEditActions,
                 range editedRange: NSRange,
                 changeInLength delta: Int) {

    let boldRange = // Range including start and end **
    let start = boldRange.location

    guard let font = textStorage.attribute(.font, at: start, effectiveRange: nil) as? NSFont
        else { fatalError() }
    let boldFont = NSFontManager.shared.convert(font, toHaveTrait: .boldFontMask)

    textStorage.addAttribute(.font, value: boldFont, range: boldRange)
}
```

Our Journal

Wednesday, June 6, 2018

Today I celebrated the crazy ones. The misfits. **The rebels.* The troublemakers.

Word Count: 12

Our Journal

Wednesday, June 6, 2018

Today I celebrated the crazy ones. The misfits. **The rebels.** The troublemakers.

Word Count: 12

Our Journal

Wednesday, June 6, 2018

Today I celebrated the crazy ones. The misfits. **The rebels.** The troublemakers.

Word Count: 12

Our Journal

Wednesday, June 6, 2018

Today I celebrated the crazy ones. The misfits. **The rebels.** The troublemakers.

```swift
let string = “Hello World!”
```

Word Count: 17

Our Journal

Wednesday, June 6, 2018

Today I celebrated the crazy ones. The misfits. **The rebels.** The troublemakers.

Swift Code

```swift
let string = “Hello World!”
```

Word Count: 17

Delegation

NSTextStorageDelegate

```
func textStorage(_:didProcessEditing:range:changeInLength:)
```

Notification

NSTextStorage

```
didProcessEditingNotification
```

Subclassing

NSTextStorage

```
override func replaceCharacters(in:with:)
```

Delegation

NSTextStorageDelegate

```
func textStorage(_:didProcessEditing:range:changeInLength:)
```

Notification

NSTextStorage

```
didProcessEditingNotification
```

Subclassing

NSTextStorage

```
override func replaceCharacters(in:with:)
```

AppKit

```
class CustomTextStorage: NSTextStorage {
    override var string: String {
        // Return string
    }
    override func attributes(at: Int,
                effectiveRange: NSRangePointer?) -> [NSAttributedString.Key : Any] {
        // Return attributes
    }
    override func replaceCharacters(in range: NSRange,
                                    with str: String) {
        // Replace characters
    }
    override func setAttributes(_: [NSAttributedString.Key : Any]?,
                                range: NSRange) {
        // Set attributes
    }
}
```

AppKit

```
class CustomTextStorage: NSTextStorage {

    override func replaceCharacters(in range: NSRange, with str: String) {
        // Replace characters

        let codeBlockRange = // Range including start and end ```
        let paragraphStyle = // Get existing or new paragraph style

        let mutableStyle = NSMutableParagraphStyle()
        mutableStyle.setParagraphStyle(paragraphStyle)
        mutableStyle.textBlocks = [NSTextBlock()]

        addAttribute(.paragraphStyle, value: mutableStyle, range: codeBlockRange)
    }
}
```

AppKit

```
class CustomTextStorage: NSTextStorage {

    override func replaceCharacters(in range: NSRange, with str: String) {
        // Replace characters

        let codeBlockRange = // Range including start and end ```
        let paragraphStyle = // Get existing or new paragraph style

        let mutableStyle = NSMutableParagraphStyle()
        mutableStyle.setParagraphStyle(paragraphStyle)
        mutableStyle.textBlocks = [NSTextBlock()]

        addAttribute(.paragraphStyle, value: mutableStyle, range: codeBlockRange)
    }
}
```

AppKit

```
class CustomTextStorage: NSTextStorage {

    override func replaceCharacters(in range: NSRange, with str: String) {
        // Replace characters

        let codeBlockRange = // Range including start and end ```
        let paragraphStyle = // Get existing or new paragraph style

        let mutableStyle = NSMutableParagraphStyle()
        mutableStyle.setParagraphStyle(paragraphStyle)
        mutableStyle.textBlocks = [NSTextBlock()]

        addAttribute(.paragraphStyle, value: mutableStyle, range: codeBlockRange)
    }
}
```

AppKit

```
class CodeBlock: NSTextBlock {

    override init() {
        super.init()

        setWidth(15.0, type: .absoluteValueType, for: .padding)
        setWidth(45.0, type: .absoluteValueType, for: .padding, edge: .minY)

        backgroundColor = NSColor(white: 0.95, alpha: 1.0)
    }
}
```

AppKit

```
class CodeBlock: NSTextBlock {

    override init() {
        super.init()

        setWidth(15.0, type: .absoluteValueType, for: .padding)
        setWidth(45.0, type: .absoluteValueType, for: .padding, edge: .minY)

        backgroundColor = NSColor(white: 0.95, alpha: 1.0)
    }
}
```

AppKit

```swift
class CodeBlock: NSTextBlock {

    override func drawBackground(withFrame frameRect: NSRect, in controlView: NSView,
characterRange charRange: NSRange, layoutManager: NSLayoutManager) {

        let adjustedFrame: NSRect = // Padded rect inside frameRect
        super.drawBackground(withFrame: adjustedFrame, in: controlView, characterRange:
charRange, layoutManager: layoutManager)

        let drawPoint: NSPoint = // Point inset from adjustedFrame origin
        let drawString = NSString(string: "Swift Code")
        let attributes = [NSAttributedString.Key.font: font, .foregroundColor: .blue]
        drawString.draw(at: drawPoint, withAttributes: attributes)
    }
}
```

AppKit

```
class CodeBlock: NSTextBlock {

    override func drawBackground(withFrame frameRect: NSRect, in controlView: NSView,
characterRange charRange: NSRange, layoutManager: NSLayoutManager) {

        let adjustedFrame: NSRect = // Padded rect inside frameRect
        super.drawBackground(withFrame: adjustedFrame, in: controlView, characterRange:
charRange, layoutManager: layoutManager)

        let drawPoint: NSPoint = // Point inset from adjustedFrame origin
        let drawString = NSString(string: "Swift Code")
        let attributes = [NSAttributedString.Key.font: font, .foregroundColor: .blue]
        drawString.draw(at: drawPoint, withAttributes: attributes)
    }
}
```

AppKit

```
class CustomTextStorage: NSTextStorage {

    override func replaceCharacters(in range: NSRange, with str: String) {
        // Replace characters

        let codeBlockRange = // Range including start and end ```
        let paragraphStyle = // Get existing or new paragraph style

        let mutableStyle = NSMutableParagraphStyle()
        mutableStyle.setParagraphStyle(paragraphStyle)
        mutableStyle.textBlocks = [CodeBlock()]

        addAttribute(.paragraphStyle, value: mutableStyle, range: codeBlockRange)
    }
}
```

AppKit

```
class CustomTextStorage: NSTextStorage {

    override func replaceCharacters(in range: NSRange, with str: String) {
        // Replace characters

        let codeBlockRange = // Range including start and end ```
        let paragraphStyle = // Get existing or new paragraph style

        let mutableStyle = NSMutableParagraphStyle()
        mutableStyle.setParagraphStyle(paragraphStyle)
        mutableStyle.textBlocks = [CodeBlock()]

        addAttribute(.paragraphStyle, value: mutableStyle, range: codeBlockRange)
    }
}
```

AppKit

```
class ViewController: NSViewController {

    @IBOutlet var textView: NSTextView!

    override func viewDidLoad() {
        super.viewDidLoad()

        guard let layoutManager = textView.layoutManager else { fatalError() }

        let customTextStorage = CustomTextStorage()
        layoutManager.replaceTextStorage(customTextStorage)
    }
}
```

AppKit

```
class ViewController: NSViewController {

    @IBOutlet var textView: NSTextView!

    override func viewDidLoad() {
        super.viewDidLoad()

        guard let layoutManager = textView.layoutManager else { fatalError() }

        let customTextStorage = CustomTextStorage()
        layoutManager.replaceTextStorage(customTextStorage)
    }
}
```

Our Journal

Wednesday, June 6, 2018

Today I celebrated the crazy ones. The misfits. **The rebels.** The troublemakers.

Swift Code

```swift
let string = “Hello World!”
```

Word Count: 17

OurJournal
My Mac
Finished running OurJournal : OurJournal
Application Scene
Window Controller Scene
View Controller Scene
View Controller
View
Wednesday, June 6, 2018
Word Count:
Word Count Label
Scroll View - Text View
Clip View
Text View
Scroller
Scroll View - Text View
Clip View
Text View
Scroller
Constraints
First Responder
Wednesday, June 6, 2018
Word Count: 0
Text View
Text
No Font
Text
Text Direction
Natural
Orientation
Horizontal
Selection
Character
Behavior
Editable
Selectable
Field Editor
Allows
Rich Text
Undo
Graphics
Image Editing
Non-contiguous Layout
Background Color Change
Only Roman Characters
Uses
Font Panel
Ruler
Inspector Bar
Find
None
Incremental Searching
Display
Draws Background
Text Color
Text Color (Default)
Background
Custom
Insertion
Control Text Color
Linguistics
Continuous Spell Checking
Automatic Spelling Correcti...
Grammar Checking
Substitution
Smart Links
Smart Quotes
Smart Dashes
Data Detectors
Text Replacement
Editing
Smart Insert and Delete
Touch Bar
Auto Text Completion
Allows Character Picker
View

OurJournal
My Mac
Finished running OurJournal : OurJournal
Application Scene
Window Controller Scene
View Controller Scene
View Controller
View
Wednesday, June 6, 2018
Word Count:
Word Count Label
Scroll View - Text View
Clip View
Text View
Scroller
Scroll View - Text View
Clip View
Text View
Scroller
Constraints
First Responder
Wednesday, June 6, 2018
Word Count: 0
Text View
Text
No Font
Text
Text Direction
Natural
Orientation
Horizontal
Selection
Character
Behavior
Editable
Selectable
Field Editor
Allows
Rich Text
Undo
Graphics
Image Editing
Non-contiguous Layout
Background Color Change
Only Roman Characters
Uses
Font Panel
Ruler
Inspector Bar
Find
None
Incremental Searching
Display
Draws Background
Text Color
Text Color (Default)
Background
Custom
Insertion
Control Text Color
Linguistics
Continuous Spell Checking
Automatic Spelling Correcti...
Grammar Checking
Substitution
Smart Links
Smart Quotes
Smart Dashes
Data Detectors
Text Replacement
Editing
Smart Insert and Delete
Touch Bar
Auto Text Completion
Allows Character Picker
View

OurJournal
My Mac
Finished running OurJournal : OurJournal
Application Scene
Window Controller Scene
View Controller Scene
View Controller
View
Wednesday, June 6, 2018
Word Count:
Word Count Label
Scroll View - Text View
Clip View
Text View
Scroller
Scroll View - Text View
Clip View
Text View
Scroller
Constraints
First Responder
Wednesday, June 6, 2018
Word Count: 0
Text View
Text
No Font
Text
Text Direction
Natural
Orientation
Horizontal
Selection
Character
Behavior
Editable
Selectable
Field Editor
Allows
Rich Text
Undo
Graphics
Image Editing
Non-contiguous Layout
Background Color Change
Only Roman Characters
Uses
Font Panel
Ruler
Inspector Bar
Find
None
Incremental Searching
Display
Draws Background
Text Color
Text Color (Default)
Background
Custom
Insertion
Control Text Color
Linguistics
Continuous Spell Checking
Automatic Spelling Correcti...
Grammar Checking
Substitution
Smart Links
Smart Quotes
Smart Dashes
Data Detectors
Text Replacement
Editing
Smart Insert and Delete
Touch Bar
Auto Text Completion
Allows Character Picker
View

OurJournal
My Mac
Finished running OurJournal : OurJournal
Application Scene
Window Controller Scene
View Controller Scene
View Controller
View
Wednesday, June 6, 2018
Word Count:
Word Count Label
Scroll View - Text View
Clip View
Text View
Scroller
Scroll View - Text View
Clip View
Text View
Scroller
Constraints
First Responder
Wednesday, June 6, 2018
Word Count: 0
Text View
Text
No Font
Text
Text Direction
Natural
Orientation
Horizontal
Behavior
Editable
Allows
Rich Text
Undo
Graphics
Image Editing
Non-contiguous Layout
Background Color Change
Only Roman Characters
Uses
Font Panel
Ruler
Inspector Bar
Find
None
Incremental Searching
Display
Draws Background
Text Color
Text Color (Default)
Background
Custom
Insertion
Control Text Color
Linguistics
Continuous Spell Checking
Automatic Spelling Correcti...
Grammar Checking
Substitution
Smart Links
Smart Quotes
Smart Dashes
Data Detectors
Text Replacement
Editing
Smart Insert and Delete
Touch Bar
Auto Text Completion
Allows Character Picker
View

Text Storage
Layout Manager
Layout Manager
Text Container
Text Container
weak
weak
Text View
Text View

AppKit

```
class ViewController: NSViewController {

    @IBOutlet var leftTextView: NSTextView!
    @IBOutlet var rightTextView: NSTextView!

    override func viewDidLoad() {
        super.viewDidLoad()

        guard let leftTextStorage = leftTextView.textStorage,
            let rightLayoutManager = rightTextView.layoutManager else { fatalError() }

        rightLayoutManager.replaceTextStorage(leftTextStorage)
    }
}
```

AppKit

```
class ViewController: NSViewController {

    @IBOutlet var leftTextView: NSTextView!
    @IBOutlet var rightTextView: NSTextView!

    override func viewDidLoad() {
        super.viewDidLoad()

        guard let leftTextStorage = leftTextView.textStorage,
            let rightLayoutManager = rightTextView.layoutManager else { fatalError() }

        rightLayoutManager.replaceTextStorage(leftTextStorage)
    }
}
```

Our Journal

Wednesday, June 6, 2018

Today I celebrated the crazy ones. The misfits. **The rebels.** The troublemakers.

Swift Code

```swift
let string = “Hello World!”
```

Today I celebrated the crazy ones. The misfits. **The rebels.** The troublemakers.

Swift Code

```swift
let string = “Hello World!”
```

Word Count: 17

Delegation

Notification

Subclassing

NSLayoutManagerDelegate

```
func layoutManager(_:shouldGenerateGlyphs:properties:
                   characterIndexes:font:forGlyphRange:)
```

AppKit

```
func layoutManager(_ layoutManager: NSLayoutManager,
                   shouldGenerateGlyphs glyphs: UnsafePointer<CGGlyph>,
                   properties props: UnsafePointer<NSLayoutManager.GlyphProperty>,
                   characterIndexes charIndexes: UnsafePointer<Int>,
                   font aFont: NSFont, forGlyphRange glyphRange: NSRange) -> Int {

}
```

AppKit

```
func layoutManager(_ layoutManager: NSLayoutManager,
                   shouldGenerateGlyphs glyphs: UnsafePointer<CGGlyph>,
                   properties props: UnsafePointer<NSLayoutManager.GlyphProperty>,
                   characterIndexes charIndexes: UnsafePointer<Int>,
                   font aFont: NSFont, forGlyphRange glyphRange: NSRange) -> Int {
    var controlCharProps: UnsafeMutablePointer<NSLayoutManager.GlyphProperty>? = nil
    for index in 0..<glyphRange.length {
        if charIsMarkdownCharacter { // Determine if charIndexes[index] is markdown character
            controlCharProps?[index] = .null
        }
    }





}
```

AppKit

```
func layoutManager(_ layoutManager: NSLayoutManager,
                   shouldGenerateGlyphs glyphs: UnsafePointer<CGGlyph>,
                   properties props: UnsafePointer<NSLayoutManager.GlyphProperty>,
                   characterIndexes charIndexes: UnsafePointer<Int>,
                   font aFont: NSFont, forGlyphRange glyphRange: NSRange) -> Int {
    var controlCharProps: UnsafeMutablePointer<NSLayoutManager.GlyphProperty>? = nil
    for index in 0..<glyphRange.length {
        if charIsMarkdownCharacter { // Determine if charIndexes[index] is markdown character
            controlCharProps?[index] = .null
        }
    }

}
```

AppKit

```
func layoutManager(_ layoutManager: NSLayoutManager,
                   shouldGenerateGlyphs glyphs: UnsafePointer<CGGlyph>,
                   properties props: UnsafePointer<NSLayoutManager.GlyphProperty>,
                   characterIndexes charIndexes: UnsafePointer<Int>,
                   font aFont: NSFont, forGlyphRange glyphRange: NSRange) -> Int {
    var controlCharProps: UnsafeMutablePointer<NSLayoutManager.GlyphProperty>? = nil
    for index in 0..<glyphRange.length {
        if charIsMarkdownCharacter { // Determine if charIndexes[index] is markdown character
            controlCharProps?[index] = .null
        }
    }
    if let newProps = controlCharProps {
        layoutManager.setGlyphs(glyphs, properties: newProps, characterIndexes: charIndexes,
                                font: aFont, forGlyphRange: glyphRange)
        return glyphRange.length
    } else { return 0 }
}
```

AppKit

```
func layoutManager(_ layoutManager: NSLayoutManager,
                   shouldGenerateGlyphs glyphs: UnsafePointer<CGGlyph>,
                   properties props: UnsafePointer<NSLayoutManager.GlyphProperty>,
                   characterIndexes charIndexes: UnsafePointer<Int>,
                   font aFont: NSFont, forGlyphRange glyphRange: NSRange) -> Int {
    var controlCharProps: UnsafeMutablePointer<NSLayoutManager.GlyphProperty>? = nil
    for index in 0..<glyphRange.length {
        if charIsMarkdownCharacter { // Determine if charIndexes[index] is markdown character
            controlCharProps?[index] = .null
        }
    }
    if let newProps = controlCharProps {
        layoutManager.setGlyphs(glyphs, properties: newProps, characterIndexes: charIndexes,
                                font: aFont, forGlyphRange: glyphRange)
        return glyphRange.length
    } else { return 0 }
}
```

Our Journal

Wednesday, June 6, 2018

Today I celebrated the crazy ones. The misfits.
The rebels. The troublemakers.

Swift Code

```swift
let string = “Hello World!”
```

Today I celebrated the crazy ones. The misfits.
The rebels. The troublemakers.

Swift Code

let string = “Hello World!”

Word Count: 17

Our Journal

Wednesday, June 6, 2018

Today I celebrated the crazy ones. The misfits.
The rebels. The troublemakers.

Swift Code

```swift
let string = “Hello World!”
```

Today I celebrated the crazy ones. The misfits.
The rebels. The troublemakers.

Swift Code

let string = “Hello World!”

Word Count: 17

Documentation

Text Programming Guide for iOS

Cocoa Text Architecture Guide

Introduction to Text Layout Programming Guide

Key concepts

Examples

Best practices

Correctness:

Beware of default attributes

Don't hate.

Don't hate. → **Don't** hate.

apply bold font

```
// Correctness: Beware of Default Attributes

guard let originalFont = UIFont(name: "Comic Sans MS" size: 24.0) else { return }

let boldFontDescriptor = originalFont.fontDescriptor.withSymbolicTraits(.traitBold)
let boldFont = UIFont(descriptor: boldFontDescriptor!, size: (originalFont.pointSize))

let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.font, value: boldFont, range: NSRange(location: 0, length: 5))
myTextView.attributedText = newText
```

```swift
// Correctness: Beware of Default Attributes

guard let originalFont = UIFont(name: "Comic Sans MS" size: 24.0) else { return }

let boldFontDescriptor = originalFont.fontDescriptor.withSymbolicTraits(.traitBold)
let boldFont = UIFont(descriptor: boldFontDescriptor!, size: (originalFont.pointSize))

let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.font, value: boldFont, range: NSRange(location: 0, length: 5))
myTextView.attributedText = newText
```

```swift
// Correctness: Beware of Default Attributes

guard let originalFont = UIFont(name: "Comic Sans MS" size: 24.0) else { return }

let boldFontDescriptor = originalFont.fontDescriptor.withSymbolicTraits(.traitBold)
let boldFont = UIFont(descriptor: boldFontDescriptor!, size: (originalFont.pointSize))

let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.font, value: boldFont, range: NSRange(location: 0, length: 5))
myTextView.attributedText = newText
```

```
// Correctness: Beware of Default Attributes

guard let originalFont = UIFont(name: "Comic Sans MS" size: 24.0) else { return }

let boldFontDescriptor = originalFont.fontDescriptor.withSymbolicTraits(.traitBold)
let boldFont = UIFont(descriptor: boldFontDescriptor!, size: (originalFont.pointSize))

let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.font, value: boldFont, range: NSRange(location: 0, length: 5))
myTextView.attributedText = newText
```

```
// Correctness: Beware of Default Attributes

guard let originalFont = UIFont(name: "Comic Sans MS" size: 24.0) else { return }

let boldFontDescriptor = originalFont.fontDescriptor.withSymbolicTraits(.traitBold)
let boldFont = UIFont(descriptor: boldFontDescriptor!, size: (originalFont.pointSize))

let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.font, value: boldFont, range: NSRange(location: 0, length: 5))
myTextView.attributedText = newText
```

Don't hate.

```
// Correctness: Beware of Default Attributes

guard let originalFont = UIFont(name: "Comic Sans MS" size: 24.0) else { return }

let boldFontDescriptor = originalFont.fontDescriptor.withSymbolicTraits(.traitBold)
let boldFont = UIFont(descriptor: boldFontDescriptor!, size: (originalFont.pointSize))

let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.font, value: boldFont, range: NSRange(location: 0, length: 5))
myTextView.attributedText = newText
```

Don't hate.

```
// Correctness: Beware of Default Attributes

guard let originalFont = UIFont(name: "Comic Sans MS" size: 24.0) else { return }

let boldFontDescriptor = originalFont.fontDescriptor.withSymbolicTraits(.traitBold)
let boldFont = UIFont(descriptor: boldFontDescriptor!, size: (originalFont.pointSize))

let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.font, value: boldFont, range: NSRange(location: 0, length: 5))
myTextView.attributedText = newText
```

Don't hate.

```
// Correctness: Beware of Default Attributes

guard let originalFont = UIFont(name: "Comic Sans MS" size: 24.0) else { return }

let boldFontDescriptor = originalFont.fontDescriptor.withSymbolicTraits(.traitBold)
let boldFont = UIFont(descriptor: boldFontDescriptor!, size: (originalFont.pointSize))

let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.font, value: boldFont, range: NSRange(location: 0, length: 5))
myTextView.attributedText = newText
```

```
// Correctness: Beware of Default Attributes

guard let originalFont = UIFont(name: "Comic Sans MS" size: 24.0) else { return }

let boldFontDescriptor = originalFont.fontDescriptor.withSymbolicTraits(.traitBold)
let boldFont = UIFont(descriptor: boldFontDescriptor!, size: (originalFont.pointSize))

let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.font, value: boldFont, range: NSRange(location: 0, length: 5))
myTextView.attributedText = newText
```

```swift
// Correctness: Beware of Default Attributes

guard let originalFont = UIFont(name: "Comic Sans MS" size: 24.0) else { return }

let boldFontDescriptor = originalFont.fontDescriptor.withSymbolicTraits(.traitBold)
let boldFont = UIFont(descriptor: boldFontDescriptor!, size: (originalFont.pointSize))

let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.font, value: boldFont, range: NSRange(location: 0, length: 5))
myTextView.attributedText = newText
```

Don't hate.

Comic Sans 24

Don't hate.

Comic Sans 24

Don't hate.

Comic Sans 24

Don’t hate.

Comic Sans 24

Don’t hate.

Helvetica 12

Don’t hate.

Comic Sans 24

Don’t hate.

apply bold font

Helvetica 12

Don’t hate.

Comic Sans 24

Don’t hate.

apply bold font

Helvetica 12

Don’t hate.

Comic Sans 24

Don’t hate.

apply bold font

Helvetica 12

Don't hate.

Comic Sans 24

Don't hate.

apply bold font

Helvetica 12

Don't hate.

Don't hate.

Comic Sans 24

Don't hate.

apply bold font

Helvetica 12

Don't hate.

Comic Sans Bold 24

Don't hate.

Comic Sans 24

Don't hate.

apply bold font

Helvetica 12

Don't hate.

Comic Sans Bold 24

Helvetica 12

```
// Correctness
// Approach #1: Avoid mixing plain and attributed text

guard let originalFont = UIFont(name: "Comic Sans MS” size: 24.0) else { return }

let boldFontDescriptor = originalFont.fontDescriptor.withSymbolicTraits(.traitBold)
let boldFont = UIFont(descriptor: boldFontDescriptor!, size: (originalFont.pointSize))

let newText = NSMutableAttributedString(attributedString: myTextView.attributedText)
newText.addAttribute(.font, value: boldFont, range: NSRange(location: 0, length: 5))
myTextView.attributedText = newText
```

```
// Correctness
// Approach #1: Avoid mixing plain and attributed text

guard let originalFont = UIFont(name: "Comic Sans MS" size: 24.0) else { return }

let boldFontDescriptor = originalFont.fontDescriptor.withSymbolicTraits(.traitBold)
let boldFont = UIFont(descriptor: boldFontDescriptor!, size: (originalFont.pointSize))

let newText = NSMutableAttributedString(attributedString: myTextView.attributedText)
newText.addAttribute(.font, value: boldFont, range: NSRange(location: 0, length: 5))
myTextView.attributedText = newText
```

```
// Correctness
// Approach #2: Specify attributes on creation

guard let originalFont = UIFont(name: "Comic Sans MS" size: 24.0) else { return }

let boldFontDescriptor = originalFont.fontDescriptor.withSymbolicTraits(.traitBold)
let boldFont = UIFont(descriptor: boldFontDescriptor!, size: (originalFont.pointSize))

let newText = NSMutableAttributedString(string: myTextView.text,
                                        attributes: [.font : originalFont])
newText.addAttribute(.font, value: boldFont, range: NSRange(location: 0, length: 5))
myTextView.attributedText = newText
```

```
// Correctness
// Approach #2: Specify attributes on creation

guard let originalFont = UIFont(name: "Comic Sans MS" size: 24.0) else { return }

let boldFontDescriptor = originalFont.fontDescriptor.withSymbolicTraits(.traitBold)
let boldFont = UIFont(descriptor: boldFontDescriptor!, size: (originalFont.pointSize))

let newText = NSMutableAttributedString(string: myTextView.text,
                                        attributes: [.font : originalFont])
newText.addAttribute(.font, value: boldFont, range: NSRange(location: 0, length: 5))
myTextView.attributedText = newText
```

Correctness

Attributes with default values

Attachment	None (no attachment)
Background color	None (no background)
Baseline offset	0.0
Font	Helvetica 12-point
Foreground color	Black
Kern	None (use font’s default kerning)
Ligature	None (use font’s standard ligatures)
Link	None (no link)
Paragraph style	Return value of NSParagraphStyle’s defaultParagraphStyle method
Superscript	0
Underline style	None (no underline)

Correctness

Attributes with default values

Attachment	None (no attachment)
Background color	None (no background)
Baseline offset	0.0
Font	Helvetica 12-point
Foreground color	Black
Kern	None (use font’s default kerning)
Ligature	None (use font’s standard ligatures)
Link	None (no link)
Paragraph style	Return value of NSParagraphStyle’s defaultParagraphStyle method
Superscript	0
Underline style	None (no underline)

```
// Correctness : Beware of Default Attributes
// Modifying paragraph style

// Use tail truncation instead of the default word wrapping
let myParagraphStyle = NSMutableParagraphStyle()
myParagraphStyle.lineBreakMode = .byTruncatingTail

// Apply paragraph style to last word
let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.paragraphStyle, value: myParagraphStyle,
                     range: NSRange(location: 6, length: 5))
myTextView.attributedText = newText
```

```
// Correctness : Beware of Default Attributes
// Modifying paragraph style

// Use tail truncation instead of the default word wrapping
let myParagraphStyle = NSMutableParagraphStyle()
myParagraphStyle.lineBreakMode = .byTruncatingTail

// Apply paragraph style to last word
let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.paragraphStyle, value: myParagraphStyle,
                     range: NSRange(location: 6, length: 5))
myTextView.attributedText = newText
```

```
// Correctness : Beware of Default Attributes
// Modifying paragraph style

// Use tail truncation instead of the default word wrapping
let myParagraphStyle = NSMutableParagraphStyle()
myParagraphStyle.lineBreakMode = .byTruncatingTail

// Apply paragraph style to last word
let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.paragraphStyle, value: myParagraphStyle,
                     range: NSRange(location: 6, length: 5))
myTextView.attributedText = newText
```

Don't...

```
// Correctness : Beware of Default Attributes
// Modifying paragraph style

// Use tail truncation instead of the default word wrapping
let myParagraphStyle = NSMutableParagraphStyle()
myParagraphStyle.lineBreakMode = .byTruncatingTail

// Apply paragraph style to last word
let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.paragraphStyle, value: myParagraphStyle,
                     range: NSRange(location: 6, length: 5))
myTextView.attributedText = newText
```

Don't...

Don't
hate.

```
// Correctness : Beware of Default Attributes
// Modifying paragraph style

// Use tail truncation instead of the default word wrapping
let myParagraphStyle = NSMutableParagraphStyle()
myParagraphStyle.lineBreakMode = .byTruncatingTail

// Apply paragraph style to last word
let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.paragraphStyle, value: myParagraphStyle,
                     range: NSRange(location: 6, length: 5))
myTextView.attributedText = newText
```

Don't...

Don't
hate.

```
// Correctness : Beware of Default Attributes
// Modifying paragraph style

// Use tail truncation instead of the default word wrapping
let myParagraphStyle = NSMutableParagraphStyle()
myParagraphStyle.lineBreakMode = .byTruncatingTail

// Apply paragraph style to last word
let newText = NSMutableAttributedString(string: myTextView.text)
newText.addAttribute(.paragraphStyle, value: myParagraphStyle,
                     range: NSRange(location: 6, length: 5))
myTextView.attributedText = newText
```

Don't...

Don't hate.

Attribute Fixing
Glyph Generation
Glyph Layout
Display

Don’t hate.

default paragraph style

custom paragraph style

Don’t hate.

default paragraph style

Correctness

Best practice

Be explicit with attributes

- Avoids “resetting” all or part of string to default attributes
- Ensure correct string coloring for Dark Mode AppKit

Performance:

Use noncontiguous layout

Performance

Layout process review

The first requisite in the preparation of good sandwiches is to have perfect bread in suitable condition. Either white, brown, rye, or entire wheat bread may be used, but it should be of close, even texture and at least one day old. Cream the butter with a wooden spoon and spread smoothly on the bread before it is cut; after cutting remove the crust and avoid spreading the butter or filling over the edge. When ready to serve, cut the sandwiches either square, triangular, long, narrow, round, or crescent shaped.

In making rolled bread sandwiches, cut off the crust of a loaf of fresh bread and spread a thin layer of butter on one end of the loaf; cut off this buttered end in as thin a slice as possible and spread with the sandwich filling ; roll up this slice and lay on a napkin draw the napkin firmly around the rolled bread and pin it. Put in a cool place until ready to serve, then remove napkin and tie the sandwiches with baby ribbon or fasten with a tooth pick.

French rolls may be used for picnics and out-of-door luncheons. Remove from the top of each roll a piece of the crust the size of a silver dollar, and with a spoon take out the centre. Fill the space with highly seasoned

viewport

shaped.

In making rolled bread sandwiches, cut off the crust of a loaf of fresh bread and spread a thin layer of butter on one end of the loaf; cut off this buttered end in as thin a slice as possible and spread with the sandwich filling ; roll up this slice and lay on a napkin draw the napkin firmly around the rolled bread and pin it. Put in a cool place until ready to serve, then remove napkin and tie the sandwiches with baby ribbon or fasten with a tooth pick.

French rolls may be used for picnics and out-of-door luncheons. Remove from the top of each roll a piece of the crust the size of a silver dollar, and with a spoon take out the centre. Fill the space with highly seasoned chopped meat, fish, lobster, or crab, replace lid, wrap in tissue paper, and serve with pickles or olives.

For very small, dainty sandwiches to be served at afternoon teas or luncheons, the bread may be baked at home in pound baking powder cans. These should be only half filled, and then allowed to rise before baking. You then have a round slice without crust.

viewport

A garnish such as the following may be used: For meat sandwiches, use pickles, olives, lettuce, water- cress, parsley, and mint. For fish sandwiches, use pickles, olives, cress, parsley, slices of lemon, and hard-boiled egg. For cheese sandwiches, use pickles and olives. For sweet sandwiches, use lettuce, maiden hair fern, smilax, berries, flowers, and candied fruit.

To keep sandwiches fresh, if prepared an hour or two before serving, wring out a napkin in cold water and cover the tray and keep in a cool place or

shaped.

In making rolled bread sandwiches, cut off the crust of a loaf of fresh bread and spread a thin layer of butter on one end of the loaf; cut off this buttered end in as thin a slice as possible and spread with the sandwich filling ; roll up this slice and lay on a napkin draw the napkin firmly around the rolled bread and pin it. Put in a cool place until ready to serve, then remove napkin and tie the sandwiches with baby ribbon or fasten with a tooth pick.

French rolls may be used for picnics and out-of-door luncheons. Remove from the top of each roll a piece of the crust the size of a silver dollar, and with a spoon take out the centre. Fill the space with highly seasoned chopped meat, fish, lobster, or crab, replace lid, wrap in tissue paper, and serve with pickles or olives.

For very small, dainty sandwiches to be served at afternoon teas or luncheons, the bread may be baked at home in pound baking powder cans. These should be only half filled, and then allowed to rise before baking. You then have a round slice without crust.

A garnish such as the following may be used: For meat sandwiches, use pickles, olives, lettuce, water- cress, parsley, and mint. For fish sandwiches, use pickles, olives, cress, parsley, slices of lemon, and hard-boiled egg. For cheese sandwiches, use pickles and olives. For sweet sandwiches, use lettuce, maiden hair fern, smilax, berries, flowers, and candied fruit.

To keep sandwiches fresh, if prepared an hour or two before serving, wring out a napkin in cold water and cover the tray and keep in a cool place or

contiguous glyph generation and layout

viewport

shaped.

In making rolled bread sandwiches, cut off the crust of a loaf of fresh bread and spread a thin layer of butter on one end of the loaf; cut off this buttered end in as thin a slice as possible and spread with the sandwich filling ; roll up this slice and lay on a napkin draw the napkin firmly around the rolled bread and pin it. Put in a cool place until ready to serve, then remove napkin and tie the sandwiches with baby ribbon or fasten with a tooth pick.

French rolls may be used for picnics and out-of-door luncheons. Remove from the top of each roll a piece of the crust the size of a silver dollar, and with a spoon take out the centre. Fill the space with highly seasoned chopped meat, fish, lobster, or c[illegible]e lid, wrap in tissue paper, and serve with pickles or olives.

For very small, dainty sandwiche[illegible]rved at afternoon teas or luncheons, the bread may be baked at home in pound baking powder cans. These should be only half filled, and then allowed to rise before baking. You then have a round slice without crust.

A garnish such as the following may be used: For meat sandwiches, use pickles, olives, lettuce, water- cress, parsley, and mint. For fish sandwiches, use pickles, olives, cress, parsley, slices of lemon, and hard-boiled egg. For cheese sandwiches, use pickles and olives. For sweet sandwiches, use lettuce, maiden hair fern, smilax, berries, flowers, and candied fruit.

To keep sandwiches fresh, if prepared an hour or two before serving, wring out a napkin in cold water and cover the tray and keep in a cool place or

contiguous glyph generation and layout

viewport

shaped.

In making rolled bread sandwiches, cut off the crust of a loaf of fresh bread and spread a thin layer of butter on one end of the loaf; cut off this buttered end in as thin a slice as possible and spread with the sandwich filling ; roll up this slice and lay on a napkin draw the napkin firmly around the rolled bread and pin it. Put in a cool place until ready to serve, then remove napkin and tie the sandwiches with baby ribbon or fasten with a tooth pick.

French rolls may be used for picnics and out-of-door luncheons. Remove from the top of each roll a piece of the crust the size of a silver dollar, and with a spoon take out the centre. Fill the space with highly seasoned chopped meat, fish, lobster, or c[illegible]e lid, wrap in tissue paper, and serve with pickles or olives.

For very small, dainty sandwiche[illegible]rved at afternoon teas or luncheons, the bread may be baked at home in pound baking powder cans. These should be only half filled, and then allowed to rise before baking. You then have a round slice without crust.

A garnish such as the following may be used: For meat sandwiches, use pickles, olives, lettuce, water- cress, parsley, and mint. For fish sandwiches, use pickles, olives, cress, parsley, slices of lemon, and hard-boiled egg. For cheese sandwiches, use pickles and olives. For sweet sandwiches, use lettuce, maiden hair fern, smilax, berries, flowers, and candied fruit.

To keep sandwiches fresh, if prepared an hour or two before serving, wring out a napkin in cold water and cover the tray and keep in a cool place or

contiguous glyph generation and layout

viewport

The first requisite in the preparation of good sandwiches is to have perfect bread in suitable condition. Either white, brown, rye, or entire wheat bread may be used, but it should be of close, even texture and at least one day old. Cream the butter with a wooden spoon and spread smoothly on the bread before it is cut; after cutting remove the crust and avoid spreading the butter or filling over the edge. When ready to serve, cut the sandwiches either square, triangular, long, narrow, round, or crescent shaped.

In making rolled bread sandwiches, cut off the crust of a loaf of fresh bread and spread a thin layer of butter on one end of the loaf; cut off this buttered end in as thin a slice as possible and spread with the sandwich filling ; roll up this slice and lay on a napkin draw the napkin firmly around the rolled bread and pin it. Put in a cool place until ready to serve, then remove napkin and tie the sandwiches with baby ribbon or fasten with a tooth pick.

French rolls may be used for picnics and out-of-door luncheons. Remove from the top of each roll a piece of the crust the size of a silver dollar, and with a spoon take out the centre. Fill the space with highly seasoned

viewport

shaped.

In making rolled bread sandwiches, cut off the crust of a loaf of fresh bread and spread a thin layer of butter on one end of the loaf; cut off this buttered end in as thin a slice as possible and spread with the sandwich filling ; roll up this slice and lay on a napkin draw the napkin firmly around the rolled bread and pin it. Put in a cool place until ready to serve, then remove napkin and tie the sandwiches with baby ribbon or fasten with a tooth pick.

noncontiguous glyph generation and layout

French rolls may be used for picnics and out-of-door luncheons. Remove from the top of each roll a piece of the crust the size of a silver dollar, and with a spoon take out the centre. Fill the space with highly seasoned chopped meat, fish, lobster, or crab, replace lid, wrap in tissue paper, and serve with pickles or olives.

For very small, dainty sandwiches to be served at afternoon teas or luncheons, the bread may be baked at home in pound baking powder cans. These should be only half filled, and then allowed to rise before baking. You then have a round slice without crust.

viewport

A garnish such as the following may be used: For meat sandwiches, use pickles, olives, lettuce, water- cress, parsley, and mint. For fish sandwiches, use pickles, olives, cress, parsley, slices of lemon, and hard-boiled egg. For cheese sandwiches, use pickles and olives. For sweet sandwiches, use lettuce, maiden hair fern, smilax, berries, flowers, and candied fruit.

To keep sandwiches fresh, if prepared an hour or two before serving, wring out a napkin in cold water and cover the tray and keep in a cool place or

Performance

Turning on noncontiguous layout

AppKit

NSTextView

- Set `allowsNonContiguousLayout` property on layout manager

Performance

Turning on noncontiguous layout

UIKit

UITextView

- Turned on by default
- Controlled by layout manager property `allowsNonContiguousLayout`
- Requires scrolling enabled

Performance

Best practices

With noncontiguous layout, avoid requesting layout for the entire text at once

```
// With only one container, this effectively requests layout for all the text
func ensureLayout(for: NSTextContainer)

// Avoid asking for a large range that includes the end of the text
func ensureLayout(forCharacterRange: NSRange)

// Avoid asking for a large range that includes the end of the text
func boundingRect(forGlyphRange: NSRange, in: NSTextContainer)
```

Related Previous Sessions

Efficient Interactions with Frameworks

WWDC 2017

Security:

Defense in depth

Your app
or framework

iOS, macOS
watchOS, tvOS

A stronger defense

Security:

Consider setting limits on text input

All text input is potentially untrusted

Text field for a single line

Text field for a single line

+

⌘V

3,142,582 characters?

Text field for a single line

+

⌘V

3,142,582 characters?

Security

Setting limits on text input

Validate text field input with:

- UITextFieldDelegate UIKit
- NSFormatter AppKit

Security enhancements are coming

Summary

Choose the Right...

Control

Customization approach

Customization point

Best Practices

Correctness

Security

Performance

More Information

https://developer.apple.com/wwdc18/221

Text and Fonts Lab	Technology Lab 11	Thursday 9:00AM
UIKit Layout and Lab	Technology Lab 2	Thursday 11:00AM
Cocoa Lab	Technology Lab 7	Friday 11:00AM

WWDC18