

iOS Application Development

Lecture 2: Introduction to Swift & Seminar Topics

Prof. Dr. Jan Borchers Media Computing Group **RWTH Aachen University**

Winter Semester '22/'23

hci.rwth-aachen.de/ios









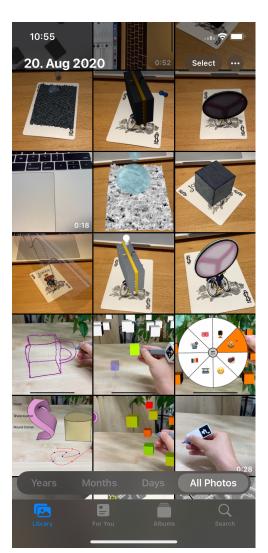
Recap

- Mobile device characteristics
 - Context
 - Screen size
 - One app at a time
- Application Styles
 - Productivity
 - Utility
 - Immersive



VS.









Weather



Super Mario Run









Swift

Prof. Dr. Jan Borchers: iOS Application Development 3





History

- Introduced at WWDC 2014
- Influenced by C and Objective-C
- But designed to be easier to learn and not depend on older languages \bullet
- "Safe, fast, and expressive"
- Open source since 2015
- Replacing Objective-C throughout iOS & macOS







Characteristics

- Clean syntax
- Type safety
- Type inference
- Automatic Reference Counting (ARC)
- Optionals





Characteristics

- Tuples and multiple return values
- Generics
- Fast and concise iteration over collections
- Structs that support methods, extensions, and protocols
- Map, filter, reduce, and other functional programming patterns
- Powerful error handling





olivernowak@i10–33 ~ % swift repl Type :help for assistance. 1> print("Hello, world!") Hello, world! 2> :quit Process 95342 exited with status = 9 (0x00000009) olivernowak@i10–33 ~ %

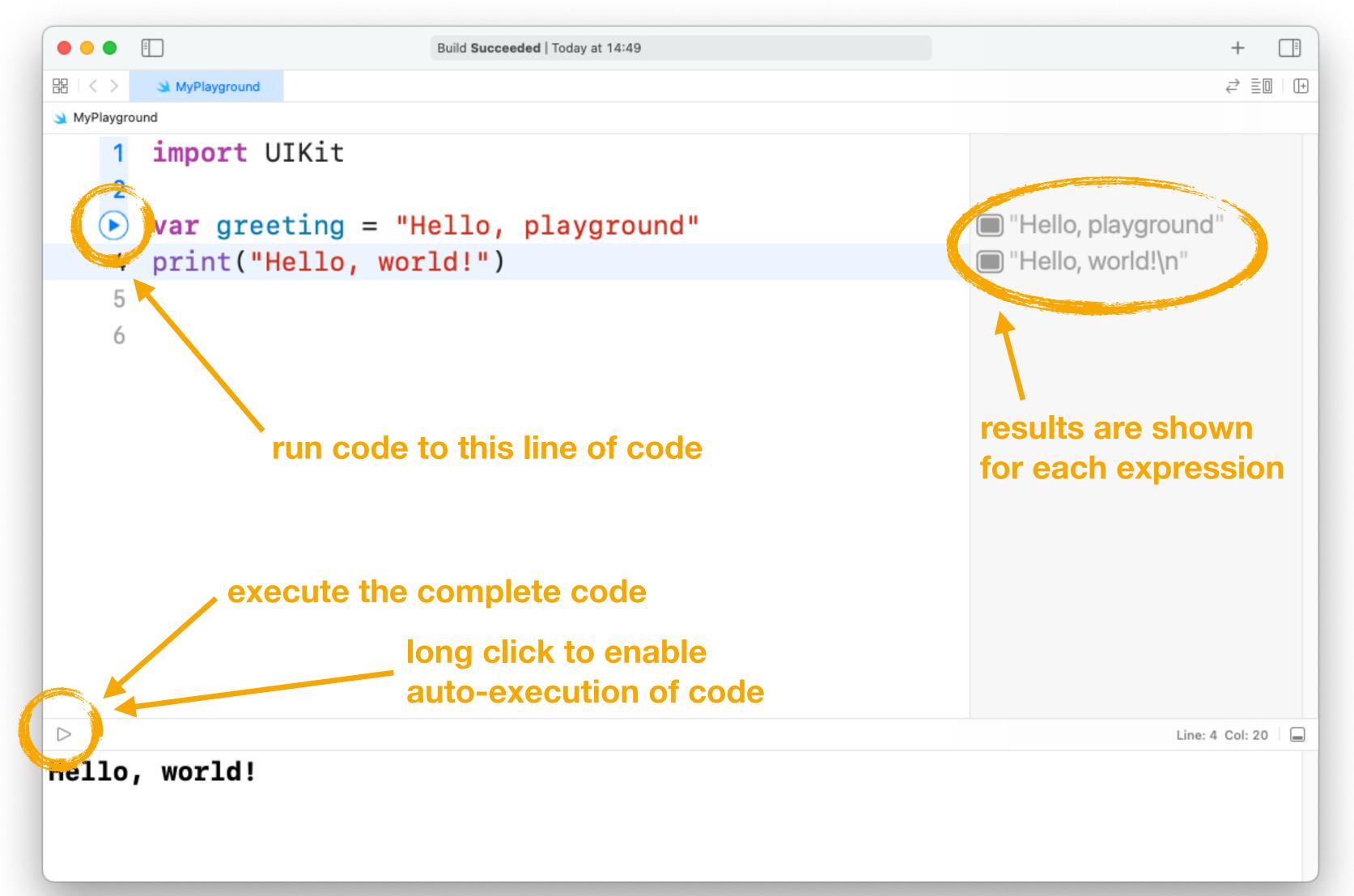
Welcome to Apple Swift version 5.7 (swiftlang-5.7.0.127.4 clang-1400.0.29.50).

terminal command





Playground





Variables and Constants

Variables are declared with var

var x = 100

Constants are declared with let

let pi = 3.14

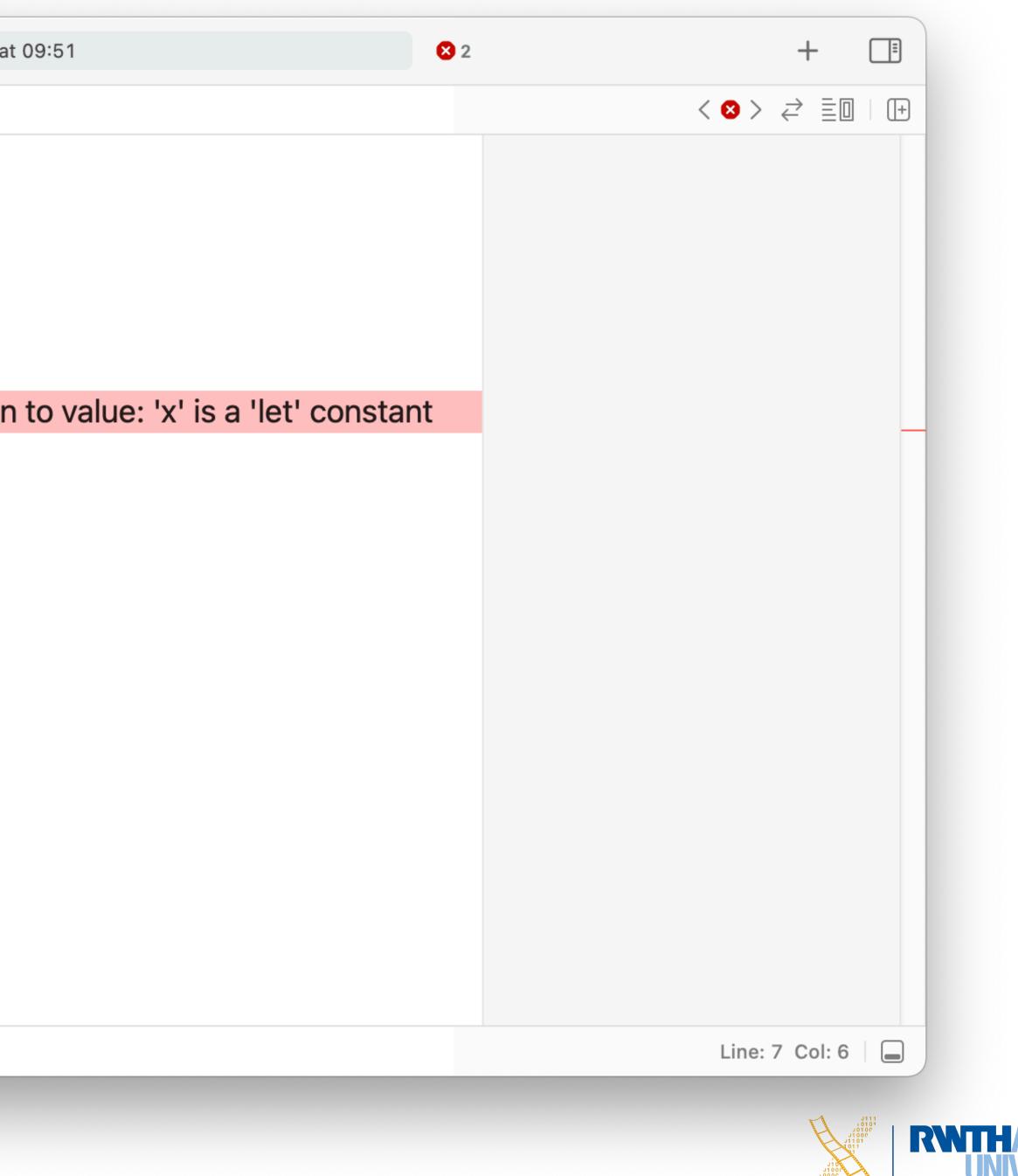








	Build	d Succeeded Today a
器 < > 🔌 MyPlayground		
1 import Cocoa		
2		
3 let x = 1		
4 var y = 2		
5		
6 y = 3		_
$7 \times = 4$	0	Cannot assigr
8		





Type Inference

var x = 100x = 99.5 // Error! x is of type Integer var x = 99.5x = 100 // Correct (x is 100.0, type: Double)

You can also explicitly specify the type

var aString : String

Swift automatically chooses the adequate data type for a variable/constant





Data Types & Type Inference

var x1 = 100// x1 is Int var x2 = 0.5// x2 is Double var x3 = x1 + x2 // Error! Can't add Int and Double var x3 = Double(x1) + x2 // Works! explicit type casting to Double var x4 = 0.5 + 100print("x4 = (x4)") // Output: x4 = 100.5 var 😂 = "LOL" // 😂 is String



- // Works! compiler adds before setting the data type





Optionals

- By default, variables and constants cannot be nill
- Optionals are variables that can also be nil

```
var i:Int? = 3
i = nil
```

Normal variables and Optionals are incompatible to each other

```
var number = Int("42")
print(number + 3) // Error! Int? != Int
                        // Error! Int? != Int
var i:Int = number
```



- // Type of number is Int? because Int() returns Int?



Tuples

• Tuples can contain multiple elements of different types

var tuple = (42, 23.0, "hello", true)var (a,b,c,d) = tuple // a = 42, b = 23.0, etc. print(tuple.2) // Prints "hello" (index starts at 0)





Control Flow

if/else

```
var x=3
if x<0 {
    print("x is negative")
} else if x==0 {
    print ("x is zero")
} else {
    print("x is positive")
}</pre>
```

Ternary Operator

```
var largest: Int
let a = 15
let b = 4
if a > b {
    largest = a
}
else {
    largest = b
}
// Can be written as
largest = a > b ? a : b
```



Control Flow

switch

```
let pt = (0.0, 0.0)
switch pt {
case (0,0):
    print("Origin.")
case (_,0):
    print ("On x-axis.")
case (0,_):
    print ("On y-axis.")
default:
    print ("Elsewhere.")
}
```

No fallthrough

```
let distance = 5
switch distance {
case 0...9:
    print("You are close.")
case 10...500:
    print("Take the train.")
default:
    print("Too far away.")
}
```



Demo: Xcode Development Environment

17 Prof. Dr. Jan Borchers: iOS Application Development



Xcode

	▲ iOSXCodeDemo iOo 〉 ▲ iPhone 13 m
	器 I く > 🛃 iOSXCodeDemo 〉 🚞 iOSXCodeDemo 〉
 iOSXCodeDemo iOSXCodeDemo AppDelegate SceneDelegate 	1 // 2 // ViewController.sw 3 // iOSXCodeDemo 4 //
 SceneDelegate ViewController Main Assets 	5 // Created by Philip 6 // 7
X LaunchScreen	8 import UIKit 9
	<pre>10 class ViewController: 11</pre>
	12override func view13super.viewDid14// Do any add15
	<pre>15 16 print("Hey iO 17 print("This i 18 print("To the</pre>
	19print("The To20print("The De
	21 print("To the 22 }
	He TI Te TI TI TI Te
+ 🖘 Filter	Auto 🌣 🛛 💿 🧻 🕞 Filter 🗛

18 Prof. Dr. Jan Borchers: iOS Application Development

nini Running iOSXCodeDemo on iPhone 13 mini	+			
🔪 ViewController 🛛 M viewDidLoad()	≓ ≣D (+		<u>)</u>	
vift			Type ViewController.sw Default - Swift So	
op Wacker on 14.10.21.			Relative to Group ViewController.swi /Users/Phil/Docum Teaching/21 WS iO XcodeDemo/ iOSXCodeDemo/ iOSXCodeDemo/ ViewController.swi	ift m ents/ DS/Demos/
UIViewController {		On Demand I	Resource Tags	
ewDidLoad() { dLoad() ditional setup after loading	the view.	Target Memb	es are taggable pership CodeDemo	
OS class") is the editor") e left is the Navigator") oolbar is on top") ebugging area is below")		Line Endings Indent Using	No Explicit Encod No Explicit Line E Spaces	ndings 🕄
e right is the Utility area")		Widths	Tab	4 🗘
CodeDemo Li	ne: 21 Col: 50		🗸 Wrap lines	
ley iOS class This is the editor To the left is the Navigator The Toolbar is on top The Debugging area is below To the right is the Utility a				
All Output \$	Ū I I I			

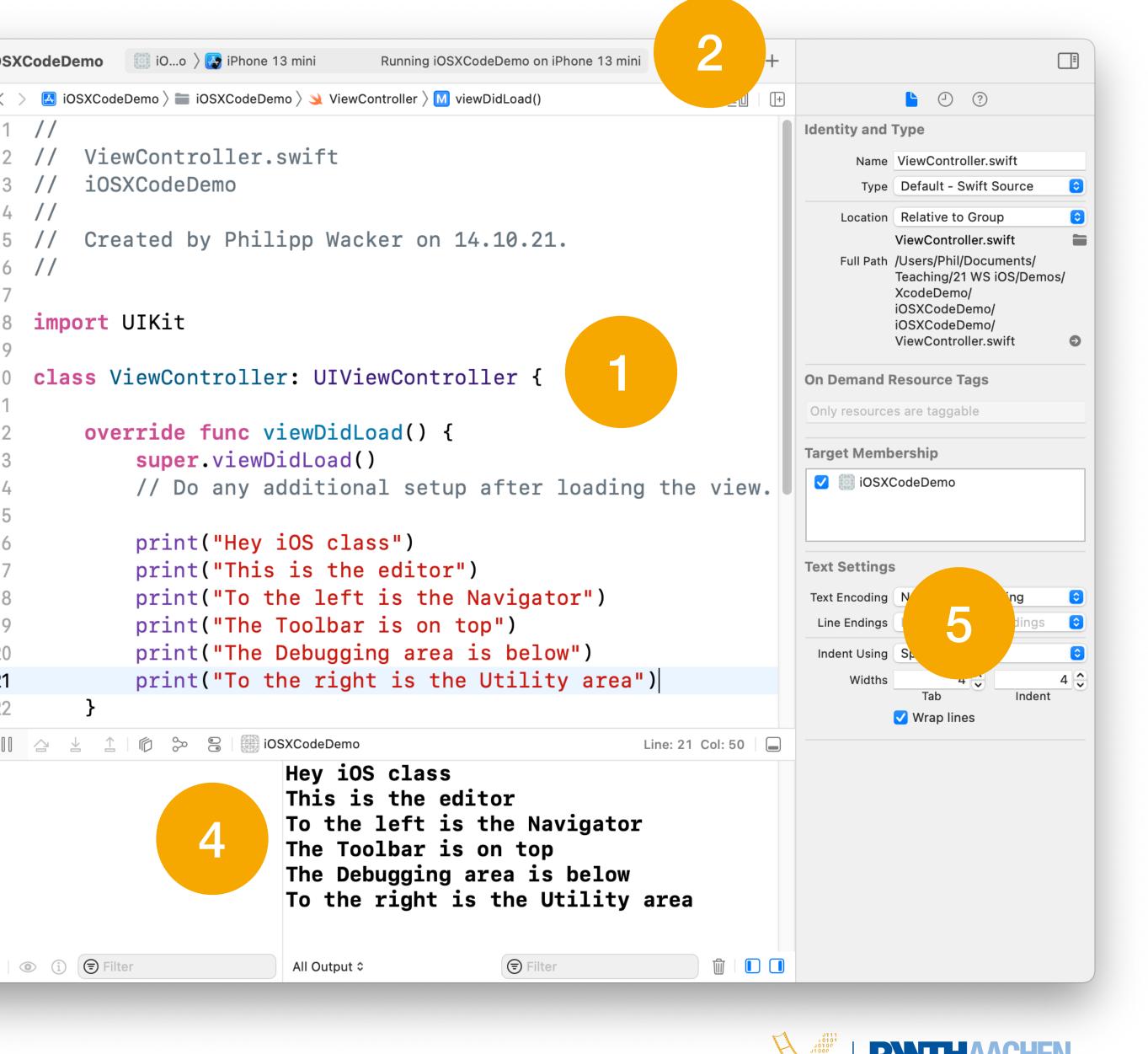




Xcode

- 5 areas
 - 1. Editor
 - 2. Toolbar
 - 3. Navigator
 - 4. Debugging
 - 5. Utility

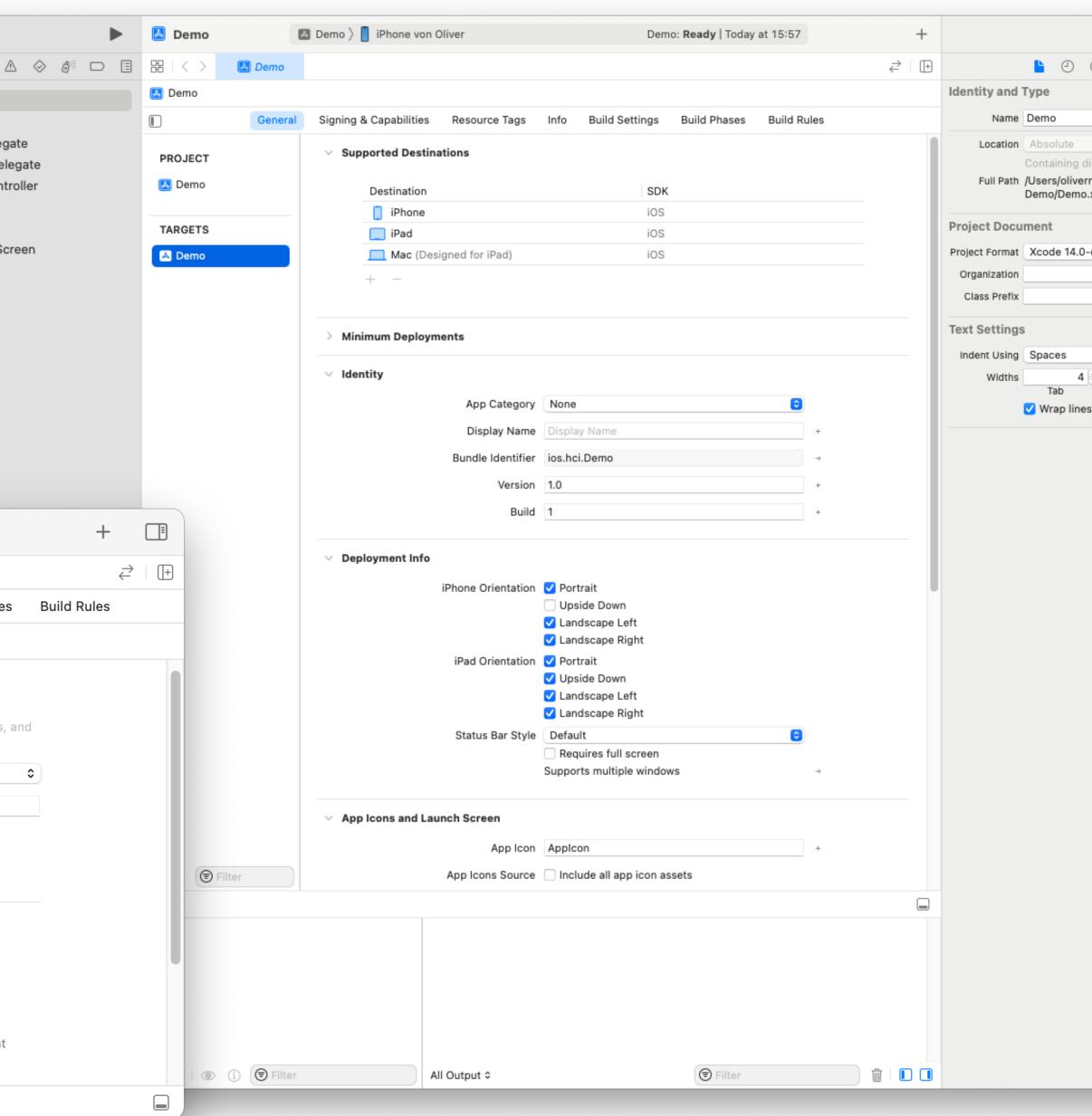
IOSXCodeDemo IOSXCodeDemo AppDelegate SceneDelegate ViewController Main Assets LaunchScreen Info 3			
 iOSXCodeDemo AppDelegate SceneDelegate ViewController Main Assets LaunchScreen Info 3 			🛃 iOS
 IOSXCodeDemo AppDelegate SceneDelegate ViewController Main Assets LaunchScreen Info 3 			器(く
AppDelegate SceneDelegate ViewController Main Assets LaunchScreen Info 3 10 11 12 13 14 15 16 17 18 19 20 21 22 11 12 13 14 15 16 17 18 19 20 21 22 11 12 13 14 15 16 17 18 19 20 21 22 11 12 13 14 15 16 16 17 18 19 20 21 22 10 11 12 13 14 15 16 16 17 18 19 20 21 22 10 10 11 12 13 14 15 16 17 18 19 20 21 22 19 10 10 10 11 12 13 14 15 16 17 18 19 20 21 22 19 20 21 22 10 10 10 10 10 10 10 10 10 10	✓ ▲ iOSXCodeDemo		1
SceneDelegate ViewController Main Assets LaunchScreen Info 3 10 11 12 13 14 15 16 17 18 19 20 21 22 10 11 12 13 14 15 16 17 18 19 20 21 22 10 10 11 12 13 14 15 16 17 18 19 20 21 22 10 10 11 12 13 14 15 16 17 18 19 20 21 22 10 10 11 12 13 14 15 16 17 18 19 20 21 22 10 10 10 10 10 11 12 13 14 15 16 17 18 19 20 21 22 10 10 10 10 11 12 13 14 15 16 17 18 19 20 21 22 10 10 10 10 10 10 10 10 10 10	✓ ■ iOSXCodeDemo		
ViewController X Main Assets LaunchScreen Info 3 10 11 12 13 14 15 16 17 18 19 20 21 22 11 12 13 14 15 16 17 18 19 20 21 20 21 22 11 12 13 14 15 16 16 17 18 19 20 21 22 11 12 13 14 15 16 16 17 18 19 20 21 22 19 10 11 12 13 14 15 16 16 17 18 19 20 21 21 22 19 19 19 19 19 19 19 19 19 19			
 Main Assets LaunchScreen Info 3 			
Assets LaunchScreen Info 3 10 11 12 13 14 15 16 17 18 19 20 21 22 13 14 15 16 17 18 19 20 21 20 21 22			
E Info 3 10 11 12 13 14 15 16 17 18 19 20 21 22 10 10 11 12 13 14 15 16 17 18 19 20 21 20 21 22 10 10 11 12 13 14 15 16 17 18 19 20 21 21 22 19 19 20 21 22 19 10 10 10 10 10 10 10 10 10 10	🖾 Assets		7
3			8
3 11 12 13 14 15 16 17 18 19 20 21 22 1 1 1 1 1 1 1 1 1 1 1 1 1	⊞ Info		9
3 12 13 14 15 16 17 18 19 20 21 20 21 20 21 20			
13 14 15 16 17 18 19 20 21 20 21 22 0			
14 15 16 17 18 19 20 21 22 21 22 0			
16 17 18 19 20 21 22 • 11			14
17 18 19 20 21 22 • 0			15
			16
			17
			22
+ The Auto \$			
+ Tilter Auto \$			
+ Tilter Auto \$			
+ 🖘 Filter 🕘 🕂 Auto 🗘			
+ 🖘 Filter 🕘 +- Auto 🌣			
+ 🖘 Filter 🕘 +- Auto 🌣			
	+ 🖘 Filter	Ð +-	Auto ≎





	e for your project	 Image: Constraint of the second second
	▲ iOSXCodeDemo ♦ iPhone 13 mini Running iOSXCodeD	emo on iPhone 13 mini
	Image: Barrier Signing & Capabilities Resource Tags Info Info	Build Settings Build Phase
 ✓ ▲ iOSXCodeDemo ✓ ■ iOSXCodeDemo 	PROJECT + Capability All Debug Release	
Capabilities		
 Access WiFi Information App Attest 		anage signing and update profiles, app IDs, iversity (Enterprise) ci.iOSXCodeDemo ofile
Ge App Groups	Access WiFi Information	t
App Transport Security Exce	Enabling Access WiFi Information allows your app to obtain information about the currently connected WiFi network	with account @rwth-aachen.de". in.
Associated Domains		de.rwth- XCodeDemo' were found ind any iOS App Development iles matching 'de.rwth- CodeDemo'.
AutoFill Credential Provider		

20 Prof. Dr. Jan Borchers: iOS Application Development



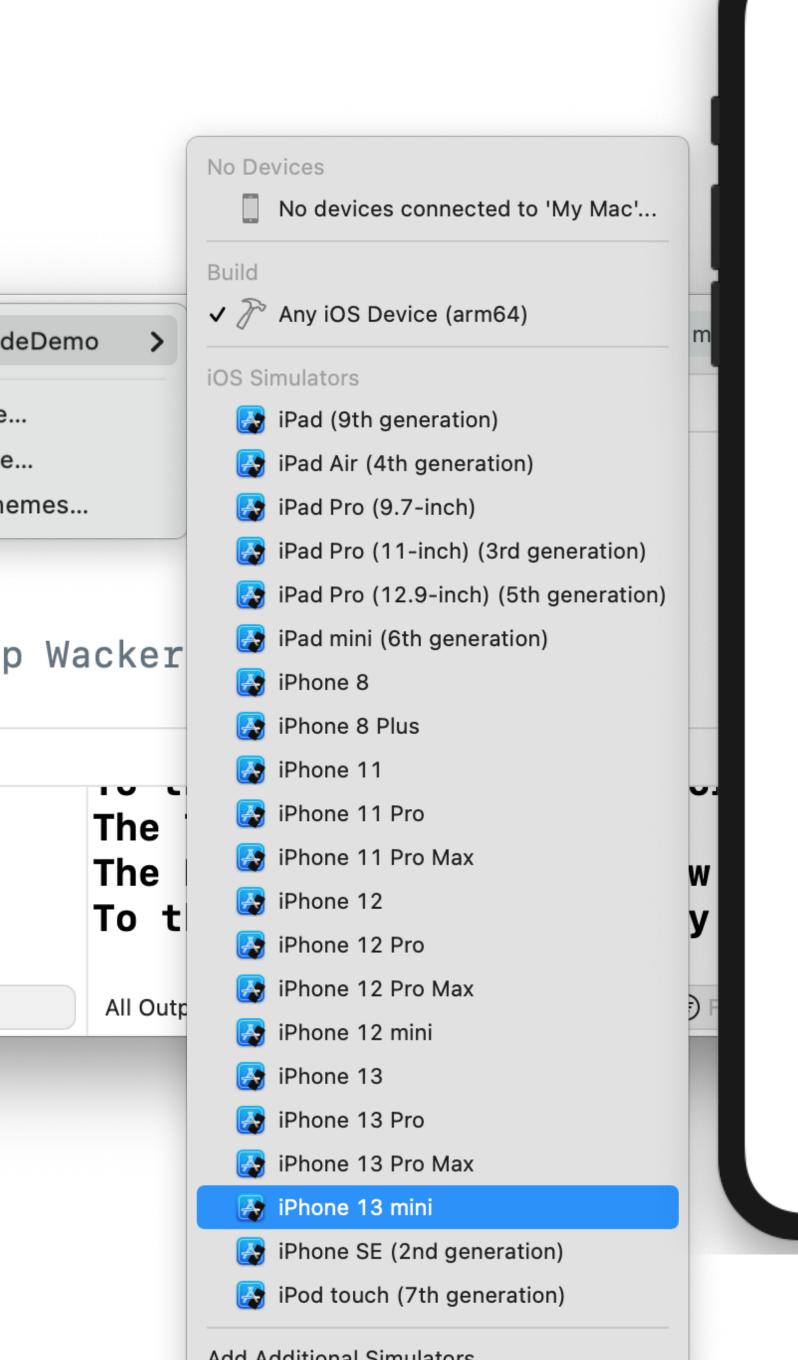


	1
?	
	0
irectory	
nowak/Deskto	p/
.xcodeproj	0
-compatible	٢
	_
	_
	0
\$	4 🗘
Indent	
S	

Building/Running

• Run on selected device

	iOSXCodeDemo
 IOSXCodeDemo IOSXCodeDemo IOSXCodeDemo AppDelegate SceneDelegate ViewController Main Assets LaunchScreen Info 	Image: Stress of the stress of th
+ 🖘 Filter	Auto 🌣 l 💿 🧻 🕞 Filter



What do you want me to say?

Say "Hello!"

Go to next view







Warnings & Errors

- Warnings don't prevent your app from compiling & running
 - Code that never gets executed
 - Variable that does not change
 - Deprecated code



- **Errors** prevent your app from building
 - Invalid code (typo, variable declaration, function calling)
 - Xcode often provides suggestions & fixes

	iOSXCodeDemo	iOSXCodeDemo 🔉 🛃 iPhone 13 mini	Build Succeeded Today at 14:59 🔒 2
$\boxtimes \Pi \bigcirc \triangle \oslash \varnothing \square \blacksquare$	⊞	🚞 iOSXCodeDemo 👌 🄌 ViewController 👌 M	viewDidLoad()
Buildtime (2) Runtime	12 overrid	e func viewDidLoad() ·	{
🔋 iOSXCodeDemo 2 issues 🛛 🔒	13 sup	er.viewDidLoad()	
🗸 🔔 Swift Compiler Warning	14 //	Do any additional setu	up after loading the v
Initialization of immutable	15		
value 'x' was never used; consider replacing with ass	16 let	$\underline{x} = "I'll never use t$	this." 2 🛆 Initialization o
ViewController.swift	17		
👌 Initialization of immutable	18		

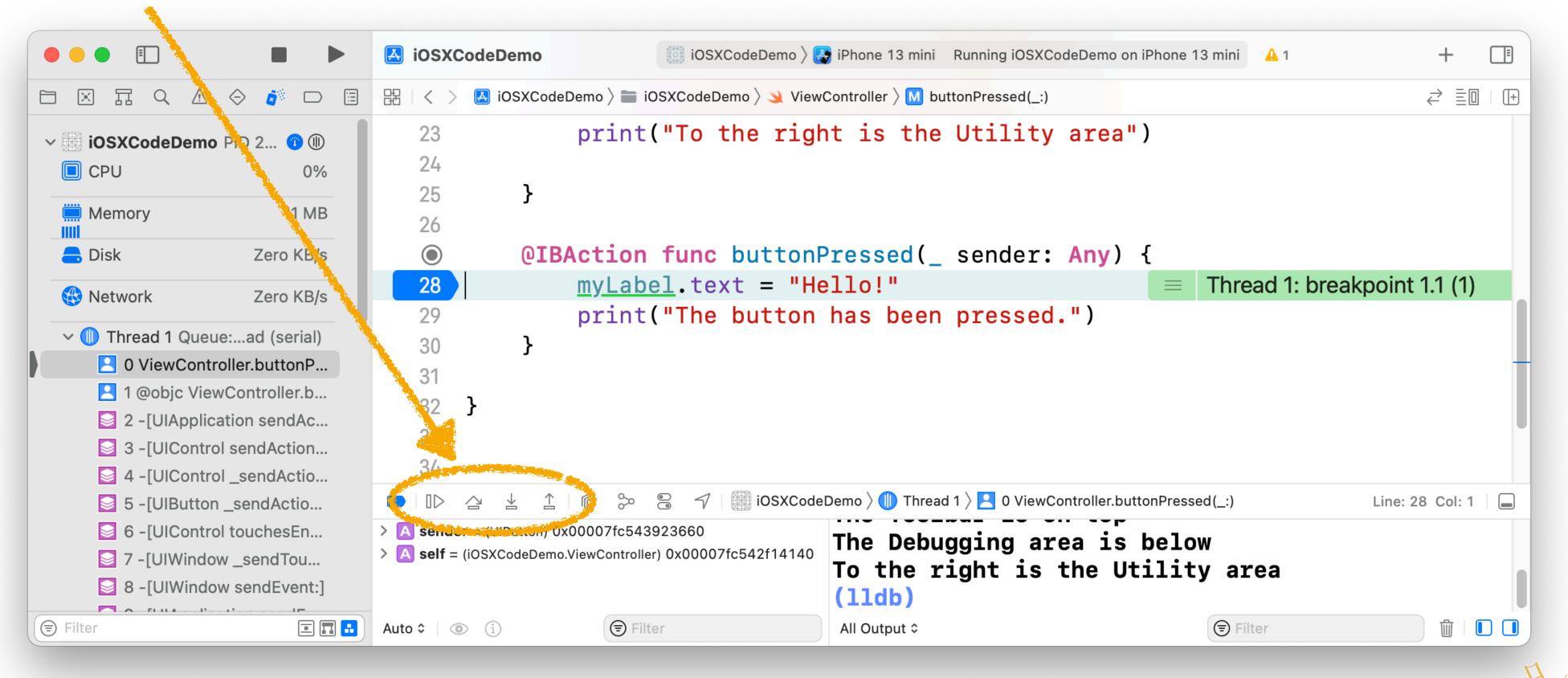
	iOSXCodeDemo	iOSXCodeDemo > 🛃 iPhone	e 13 mini	Build Failed Today at 15:03
	\mathbb{H} < > \mathbb{A} iOSXCodeDemo >	🖿 iOSXCodeDemo $ angle$ 🔌 ViewCor	ntroller \rangle (M viewDidLoad()
Buildtime (2) Runtime	12 overrid	<mark>e func</mark> viewDidLo	ad()	{
✓ ③ iOSXCodeDemo 2 issues	13 sup	<pre>er.viewDidLoad()</pre>		
✓ Swift Compiler Error	14 //	Do any additiona	l set	up after loading
Cannot assign to value: 'x' is a 'let' constant	15			
ViewController.swift	16 let	<pre>x = "I'll never</pre>	use	this."
Change 'let' to 'var' to	17 <u>x</u> =	"Or will I?"	2 🖸	Cannot assign to value:
make it mutable	18			





Debugging

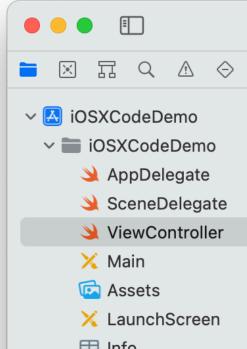
- Set breakpoints for execution on simulator and device
- Continue, Step over, Step into, Step out



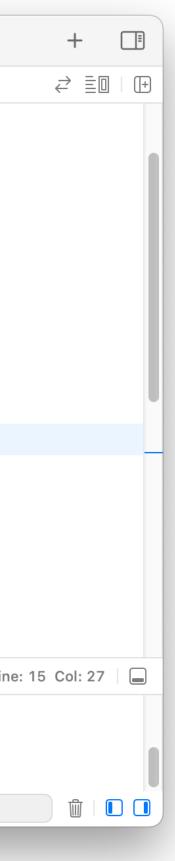
Prof. Dr. Jan Borchers: iOS Application Development 23







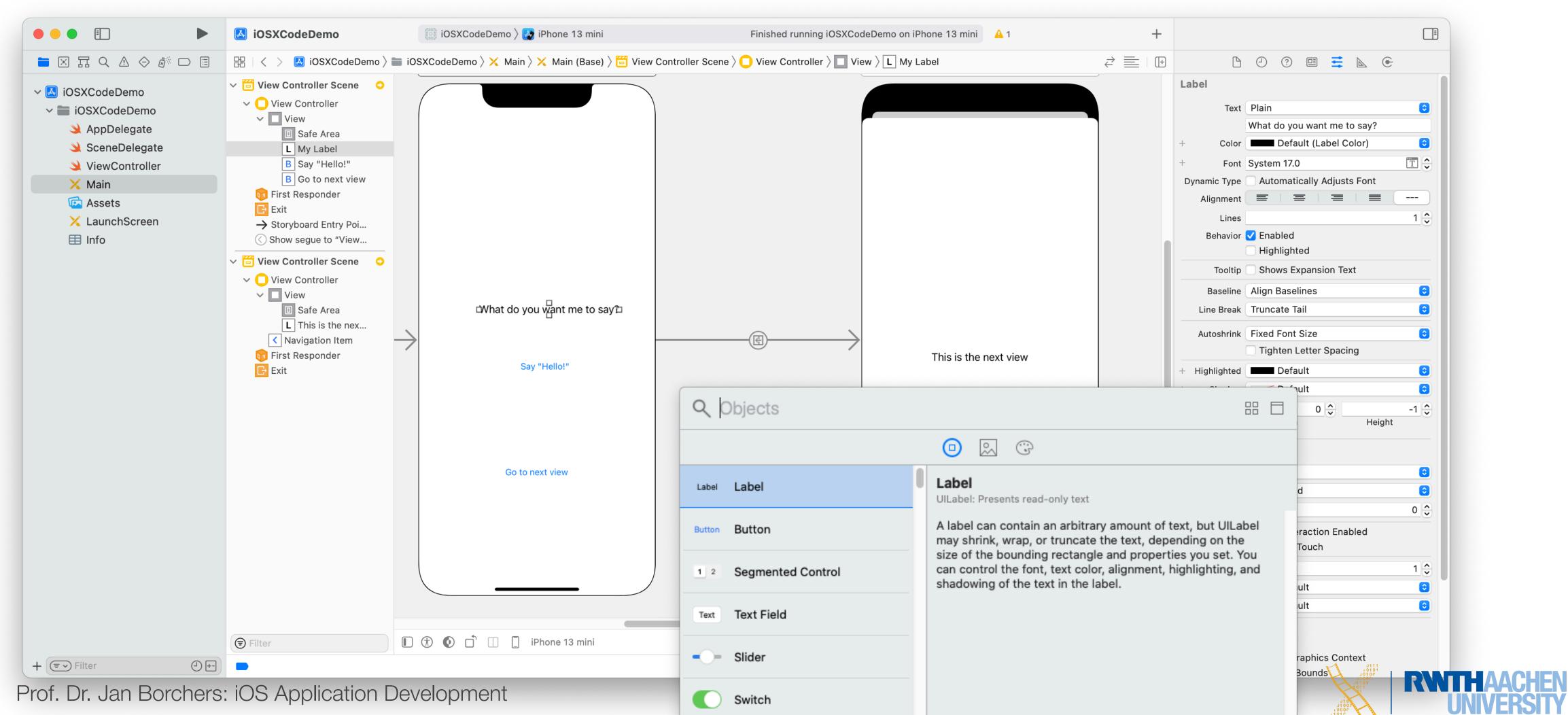
				iOSXCodeD	emo iOSemo > 🚱 iPhone 13 mini Finished running iOS	(CodeDemo on iPhone 13 mini 🔒	1 +
Doc	umentatio	+ IU3AC	CodeDemo	Summary	ter the controller's view is loaded into memory.		₹ Ξ[
 Quic 	k Help (Option+Clic	Sce View X Main K	ets InchScreen	Discussion This method memory. from a nil	nod is called after the view controller has loaded its view hierarchy into This method is called regardless of whether the view hierarchy was loaded o file or created programmatically in the loadView() method. You usually this method to perform additional initialization on views that were loaded from		
 Docu 	mentation Browser			14 15 16 17		oading the view.	
 Prog 	ramming Guides			18 19 20 21	<pre>print("Hey iOS class") print("This is the editor") print("To the left is the Navigato: print("The Toolbar is on top")</pre>		
• • I Swift	< > Q Search documentation ■ UIKit >	Ĉ			The Debugging an To the right is	rea is below the Utility area	Line: 15 Col: 27
 SwittUI TVML TVMLKit TVUIKit UIKit 	Class UIView	r		Sample Code		Developer Q	r
Essentials	An object that manages the content for a rectangular	Language			Location and Maps Programming Guide		
 About App Development Protecting the User's Pri 	area on the screen.	Swift Objective-C	Table of Conte	ents			
App Structure > ⋮ App and Environment > ⋮ Documents, Data, and P > ⋮ Resource Management	Declaration	Availability iOS 2.0+	Introduction Getting the Use Region Monitoria 	ing and iBeacon	About Location Services and M	laps	
 App Extensions Interprocess Communic Mac Catalyst User Interactions 	<pre>@MainActor class UIView : UIResponder</pre>	Mac Catalyst 13.0+ tvOS 9.0+	 Getting the Hea Course of a Dev Geocoding Loca Displaying Map 	rice tion Data	Using location-based information in your app is a great way to ke connected to the surrounding world. Whether you use this inform practical purposes (such as navigation) or for entertainment, loca information can enhance the overall user experience.	nation for	
 > Image: Touches, Presses, and G > Image: Drag and Drop > Image: Touches, Presses, and G 	Overview	Framework UIKit	 Annotating Map Providing Direct 		mormation can enhance the overall user experience.		
 E Pointer Interactions E Pencil Interactions 	Overview		Enabling Search		Carrier 🗢 11:26 AM 7		
 > Image: Focus-Based Navigation > Image	Views are the fundamental building blocks of your app's user interface, and the UIView class defines the behaviors that are common to all views. A view object renders content within its bounds rectangle, and handles any interactions with that content. The UIView class is a	On This Page Declaration ⊙ Overview ⊙ Topics ⊙	Revision Histor	y	1 market st san francisco		
24 Prof. Dr.	Jan Borchers: iOS Application Developr	nent					NTHAACHEN UNIVERSITY





Interface Builder

• Visually define your UI

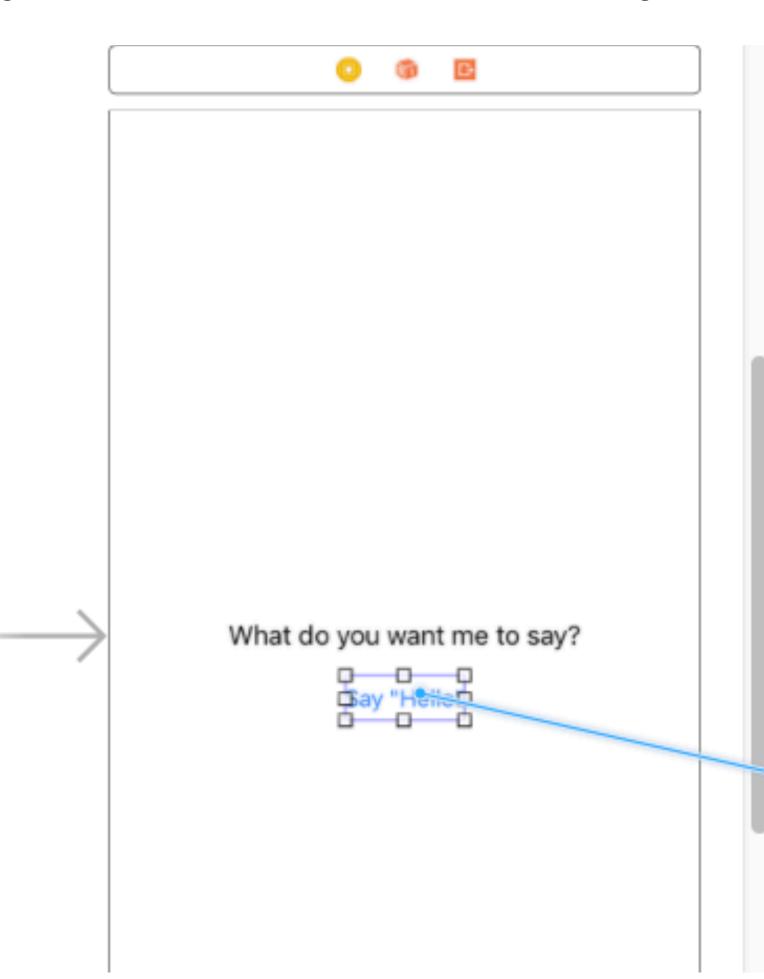


25



Outlets & Actions

Connect your UI elements with your code: Right-click + Drag



	the view, typically from
18	print("Hey iOS class")
19	<pre>print("This is the Editor")</pre>
20	<pre>print("To the left is the Na</pre>
21	<pre>print("The Toolbar is on top</pre>
22	<pre>print("The Debugging area is</pre>
23	<pre>print("To the right is the l</pre>
24	
25	}
26	
27	override func didReceiveMemoryWa
28	<pre>super.didReceiveMemoryWarnin</pre>
29	<pre>// Dispose of any resources</pre>
	recreated.
30	}
31	
32 °}	Insert Outlet, Action, or Outlet Collection
33	
34	



Outlets & Actions

- IBOutlet
 - Access the UI element from code

@IBOutlet weak var textLabel: UILabel!

- **IBAction**
 - Receive UI events

@IBAction func buttonPressed(_ sender: Any) {}

Connection	Outlet
Object	View Controller
Name	textLabel
Туре	UILabel
Storage	Weak
Cancel	Conn

Connection	Action
Object	View Controller
Name	buttonPressed
Туре	Any
Event	Touch Up Inside
Arguments	Sender
Cancel	Conr









Seminar

- 2 presentations per session
- Attendance is **mandatory**
 - Missing >1 time will lead to a 5.0 for the seminar
- 15 min presentation, ~10 min discussion
- 3 people per group
- Dates:
 - 22.11., 28.11., 29.11., 5.12., 6.12., 12.12., 13.12.
- Order is not fixed yet

- Finished version due one week before your presentation
- 15 min slide and content discussions one week before your presentation



Seminar

- Framework overview, conceptional structure
- Demo (small Playground app for Moodle)
 - Show how a problem can be solved elegantly using the framework
 - For most topics, your demo can be in either UIKit or SwiftUI
- Not a list of APIs; instead problem-solution oriented

- Structure:
 - Brief introduction & motivation
 - Basic steps to use the framework
 - Explain one or two advanced features, and show how to use them
 - Code demo •
- Deliverables: slides and demo code





1. Core Animation

- Drawing and animating what's on the screen
- What to look at: lacksquareLayers, paths, shapes, clipping, rasterization, keyframe animations, CADisplayLink





2. Haptics and Sound

- **Enriching interaction with sound and haptic feedback** \bullet
- What to look at: AVAudioPlayer, AVAudioSession, MPNowPlayingInfoCenter, UIFeedbackGenerator







3. Core Image + CI Filters

- Fast image processing and analysis
- What to look at: Automatic Enhancements, CIDetector

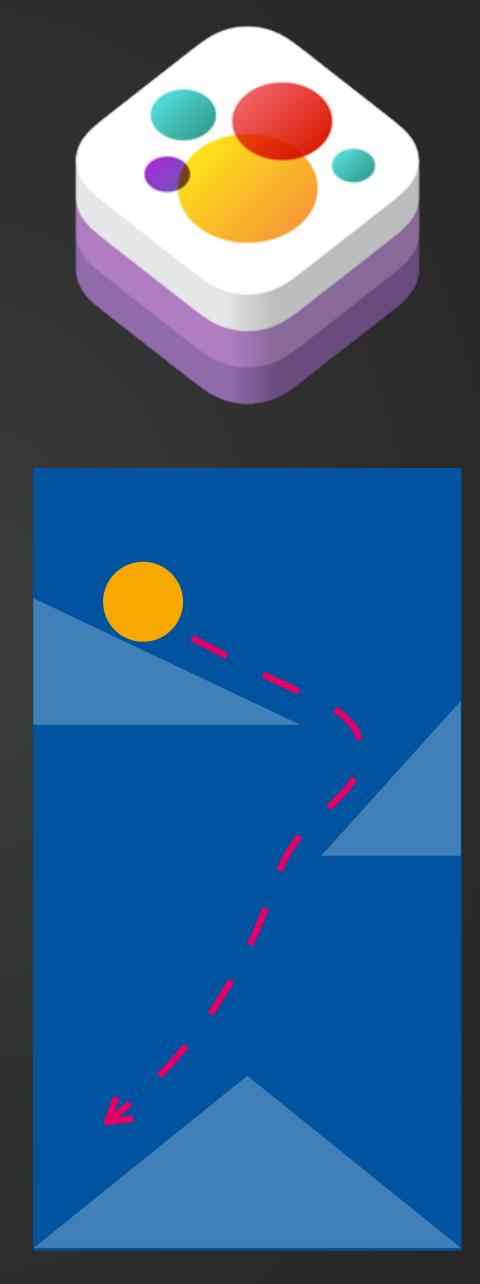






4. SpriteKit

- 2D games
- What to look at: Nodes, scenes, actions, constraints, physics







5. Working with Files

- How to save data to a file and find it in the Files app
- What to look at: FileManager, FileHandle, DocumentBrowser, Files app integration







6. Combine

- Declarative event processing
- What to look at: Publishers & Subscribers, how can the Cancellables of Combine be used for declarative UIs with UIKit?





7. Debugging in Xcode

- Using the debugger and Instruments
- What to look at: print out, View Debugger, exception breakpoints, memory leaks, ...









8. Displaying Rich Articles

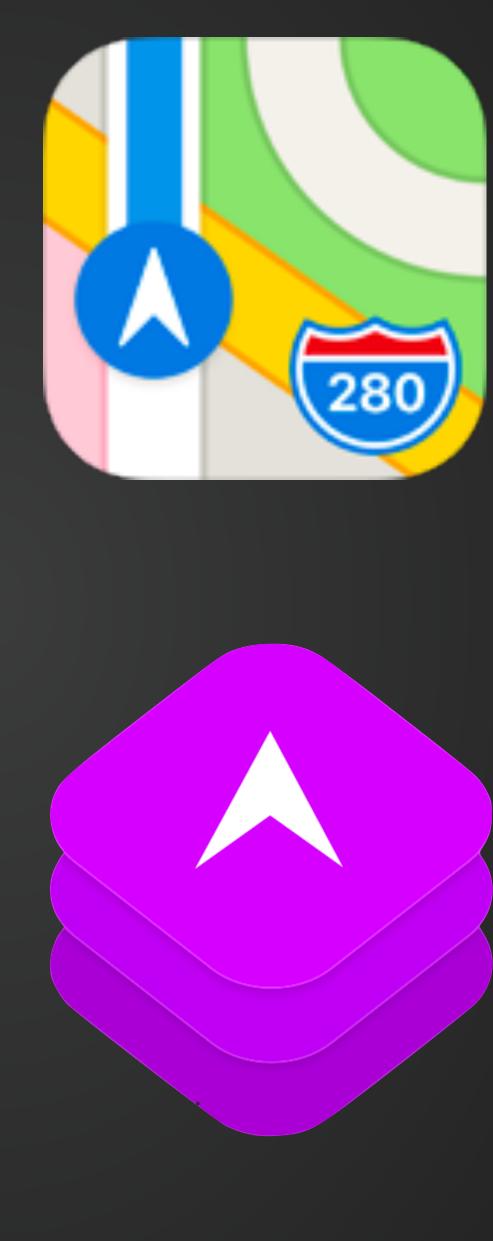
- **Displaying HTML contents in your app** \bullet
- What to look at: WKWebView, UITextView, NSAttributedString, NSParagraphStyle





9. MapKit

- Interactive maps and directions
- What to look at: lacksquareMapKit, CLLocationManager, map styles, overlays, callouts, paths, ...







10. UlPresentationController

- Create a custom view controller presentation style
- What to look at: UIPresentationController, UIViewControllerAnimatedTransitioning, UIViewControllerTransitioningDelegate

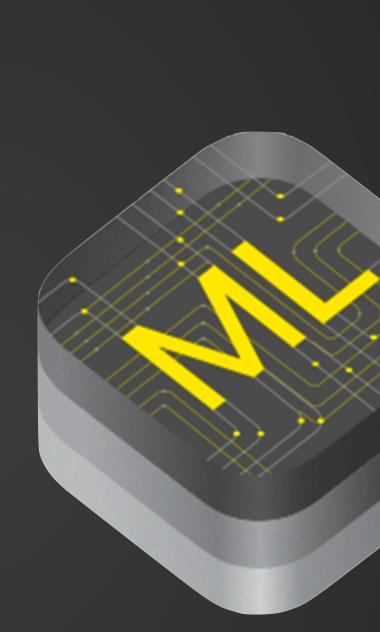




11. Core ML + Create ML

- Machine Learning in iOS
- What to look at: Framework overview in general, but focus on image classification









12. Core Data

- Persistent database
- What to look at: Managed objects, view context, fetch requests, predicates, entity relationship diagram in graphical model editor







13. watch0S

- **Designing native apps for the ÉWatch**
- What to look at: Limitations of the UI toolkit? Communication between phone and watch? Layout in watch apps Special widgets
- Can use WatchKit or SwiftUI

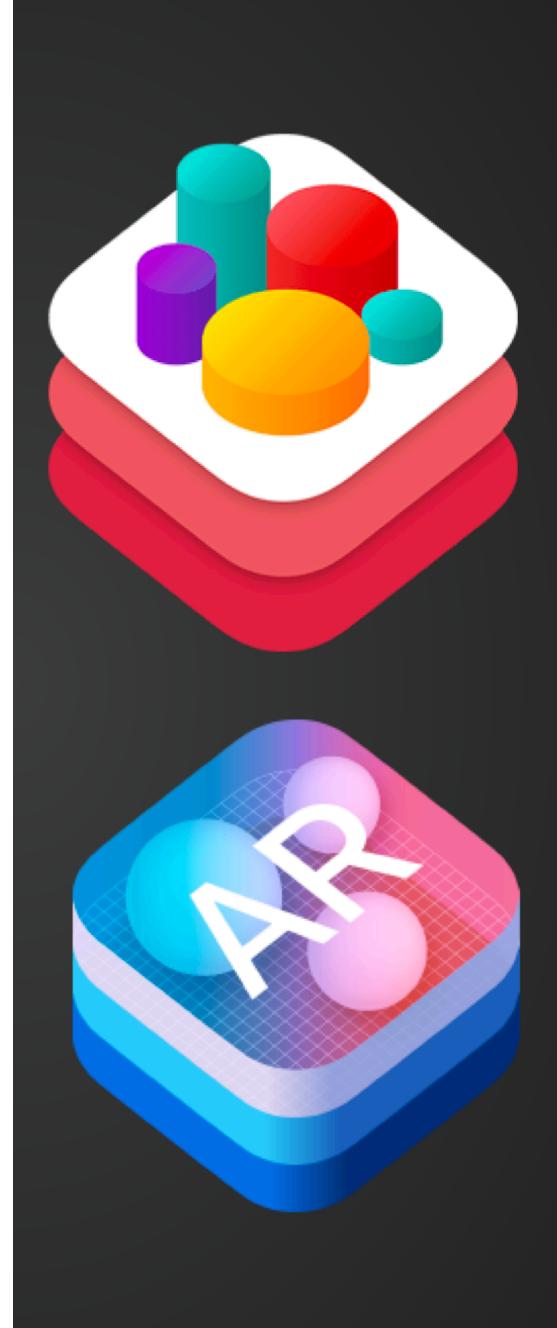






14. ARKit

- Showing AR content in a 3D graphics engine
- What to look at: Session and Configuration, AR anchors, AR onboarding, plane detection, hit testing

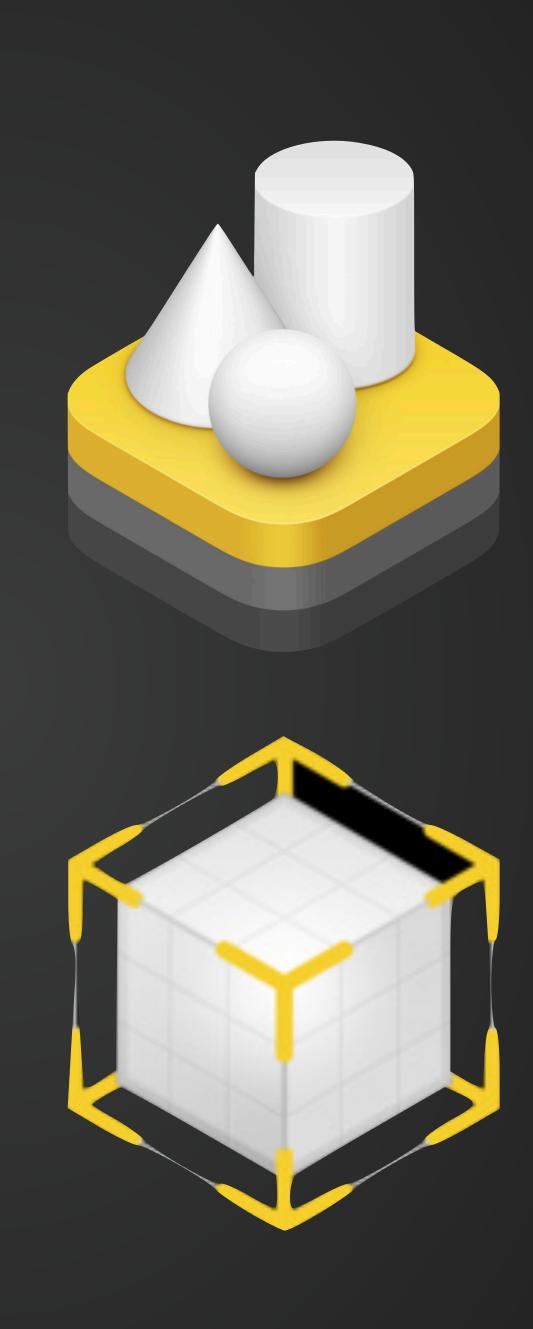






15. RealityKit & Reality Composer

- Simulate and render 3D content in AR
- What do look at: Prototype AR scenes and apps, interaction with the environment







16. Advanced SwiftUl Layout

- Using GeometryReader, priorities, fixed dimensions, alignment guides and more to create great Uls.
- What to look at: How can you express relationships (e.g. resizing a view based on the contents of some other view) in SwiftUI? How can we achieve complex layering and scrolling (e.g. stretchy headers)?







Summary

- Swift: fast, safe, expressive
- Data types, control flow, tuples
- Development Environment
 - Xcode
- Next: Strings, classes, and structs

• Seminar Topics





What's Next?

Vote for your topic

The topics can be ranked in RWTHmoodle

Only one group member should do the ranking!

Deadline: Wednesday, 19.10., at **13:00** Results will be published on Thursday

Prof. Dr. Jan Borchers: iOS Application Development 47



