

.....
.....
**iOS DESIGN
PATTERNS**
.....
.....



HANDS-ON CHALLENGES

iOS Design Patterns

Joshua Greene

Copyright ©2017 Razeware LLC.

Notice of Rights

All rights reserved. No part of this book or corresponding materials (such as text, images, or source code) may be reproduced or distributed by any means without prior written permission of the copyright owner.

Notice of Liability

This challenge and all corresponding materials (such as source code) are provided on an "as is" basis, without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose, and noninfringement. In no event shall the authors or copyright holders be liable for any claim, damages or other liability, whether in action of contract, tort or otherwise, arising from, out of or in connection with the software or the use of other dealing in the software.

Trademarks

All trademarks and registered trademarks appearing in this book are the property of their own respective owners.

Table of Contents: Overview

Visitor - Challenge	5
---------------------------	---

Table of Contents: Extended

Visitor - Challenge	5
Challenge.....	5
Challenge Solution	6
Über challenge	6

10 Visitor - Challenge

By Joshua Greene

In this challenge, you'll learn about the flexibility that the visitor pattern has brought to the "Home Info" scenes.

Challenge

Try re-arranging the **HomeInfoBuilder.storyboard** scenes.

You should be able to change the order of most of the scenes before the "Review" scene *without* changing any code. Nice! ;]

Try experimenting with this storyboard and answering these questions:

What's easy to change?

What's harder to change?

Challenge Solution

See the **HomeInfoBuilder.storyboard** within the completed challenge in the resources for this video.

What's easy to change?

You can easily reorder the scenes and even the "Review" static table view cells directly in the storyboard, and you don't have to write any additional code either!

What's harder to change?

Adding new functionality, or changing existing functionality, still requires code changes.

Über challenge

For completeness' sake, add the two remaining networking methods for `getHomeInfo` and `sendHomeInfo` to **NetworkClient.swift**.

Then, update `ReviewHomeInfoViewController` to call `sendHomeInfo` when `doneButtonPressed` is called, and then call the `delegate.homeInfoBuilderCompleted(_:)` to inform its delegate.

Lastly, update `StartHomeInfoViewController` to call `getHomeInfo` within `viewDidLoad` if the user is currently authenticated. If successful, show the `ReviewHomeInfoViewController` scene immediately. Otherwise, show the normal **Home Info Builder** scenes.

Implementation Notes:

- You'll need to make `HomeInfo` conform to `Encodable`. Here are the Keys:

```
internal struct Keys {  
    static let id = "id"  
    static let bathroomCount = "bathroom_count"  
    static let bedroomCount = "bedroom_count"  
    static let kitchenSize = "kitchen_size"  
    static let otherRoomsCount = "other_rooms_count"  
    static let squareFootage = "square_footage"  
}
```

- You'll need to add a new method for `encodeJSON` to `DictionaryEncodable`. Use `JSONSerialization` for this.
- The endpoint to GET and POST `HomeInfo` is `users/homeInfo`. Both require authentication.

This is a massive Über challenge!

Feel free to take a look at the completed challenge if you get stuck.