



Piscine iOS SwiftUI - Project 04

Shopping

Summary: This document contains the subject for the Project04 for the iOS SwiftUI piscine of

Contents

I	Instructions	2
II	Foreword	3
III	Exercise 00: One More	4
IV	Exercise 01: We need to go deeper	5
V	Exercise 02: Pass me the sugar	6
VI	Exercise 03: Grocery Store	7
VII	Exercise 04: Shopping Mall	9
VIII	Exercise 05: We are watching you	10

Chapter I

Introduction

책이란 것이 나오기 전, 사람들은 두루마리를 사용했습니다. 두루마리는 들고 다니거나 전달하기엔 편했을 순 있지만, 방대한 양의 정보를 전달하기엔 무리가 있었습니다. 하지만 책이 나오면서 많은 지식을 쌓을 수 있게 되었고, 그 지식들은 전승되기 시작했습니다.

우리가 보는 책은 여러 페이지로 되어있습니다. 페이지 단위로 정보를 저장하고, 그 페이지를 다시 한 번 묶어 많은 양의 정보를 건네줄 수 있습니다.

그렇다면 프로그램도 그렇다면 어떨까요? 책 페이지를 하나의 View 로 본다면 어떨까요? View 들을 묶어서 전달한다면 방대한 양의 정보를 전달할 수 있을 것입니다.

오늘은 데이터들을 종류에 맞게 나누어 보여줄 수 있는 View 에 대해 알아볼 것입니다.

Chapter II

Foreword

Machine Class Professor

: Why are you back?

Rancho

: Sir, I forgot to take something...

Machine Class Professor

: What?

Rancho

: Instruments that record analyse summarize organize debate and explain information which are illustrative non-illustrative hardbound paperback jacketed non-jacketed with forward introduction, table of contents, index that are indented for the enlightenment, understanding enrichment enhancement and education of the human brain through sensory root of vision... sometimes leisure

Machine Class Professor


: [*confused*] What are you trying to say?

Rancho

: Books sir! I forgot to take my books.

Chapter III

Exercise 00: One More


	Exercise 00
Project04_00	
Files to turn in: Swift Standard Library, SwiftUI	
Authorized functions: n/a	
Notes: n/a	

기존 뷰에서 이동할 수 있는 새로운 뷰를 만들어 보세요.

- 새로운 뷰에서도 당연히 기존 뷰로 넘어올 수 있어야합니다.

Chapter IV

Exercise 01: We need to go deeper


	Exercise 01
Project04_01	
Files to turn in: Swift Standard Library, SwiftUI	
Authorized functions: n/a	
Notes: n/a	

기존 View 를 제외하고, 추가로 View 를 3 개 더 만들어보세요.

- 4 개의 뷰를 각각 A, B, C, D 라고 한다면 $D \rightarrow A$ 가 가능해야합니다.
- 당연히 $A \rightarrow B \rightarrow C \rightarrow D$ 와 $D \rightarrow C \rightarrow B \rightarrow A$ 도 가능해야합니다.

Chapter V

Exercise 02: Pass me the Sugar

	Exercise 02
Project04_02	
Files to turn in: Swift Standard Library, SwiftUI	
Authorized functions: n/a	
Notes: n/a	


설탕을 주고 받는 두 개의 View 를 만들어보세요.

- 첫번째 뷰에서 설탕을 누르면 두번째 뷰의 설탕의 양이 10 증가해야합니다.
- 두번째 뷰에서 설탕을 누르면 첫번째 뷰의 설탕의 양이 10 증가해야합니다.
- 설탕 양이 바뀌었음을 볼 수 있어야합니다.
- 당연히 두번째 뷰에서 첫번째 뷰로 돌아올 수 있어야 합니다.



Chapter VI

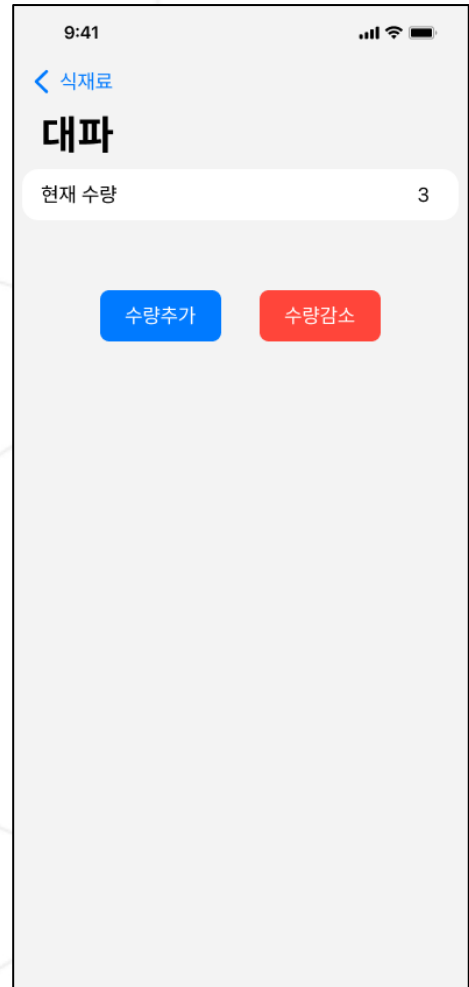
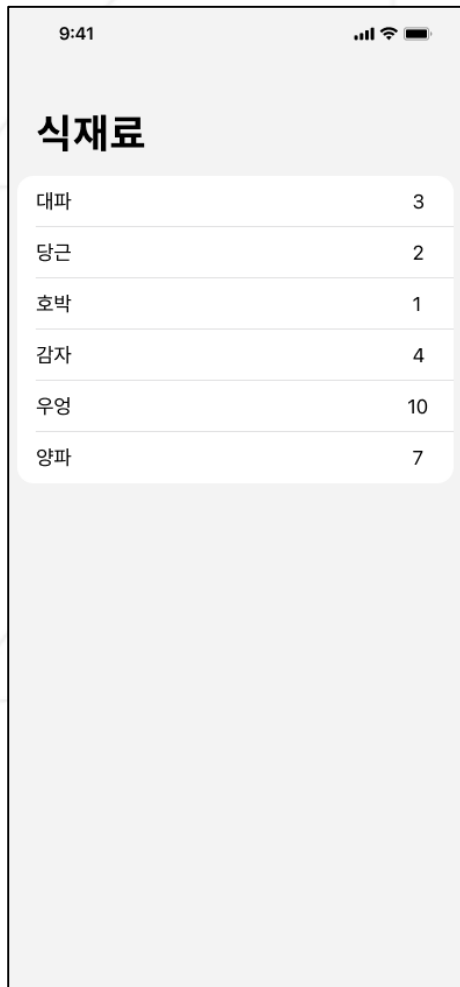
Exercise 03: Grocery Store

	Exercise 03
Project04_03	
Files to turn in: Swift Standard Library, SwiftUI	
Authorized functions: n/a	
Notes: n/a	

식재료의 양과 세부정보를 볼 수 있는 식재료 리스트를 만들어보세요.


- 식재료의 정보는 뷰의 Struct 에서 관리하면 안됩니다.
- 첫번째 뷰는 식재료의 이름과 수량을 보여주는 리스트를 보여줘야합니다.
- 첫번째 뷰에서 식재료의 버튼을 누르면 상세 페이지에 해당하는 두번째 뷰로 이동합니다.
- 두번째 뷰에는 버튼이 있어 식재료를 넣고, 뺄 수 있습니다.
- 두번째 뷰에서 갱신된 식재료의 정보를 A 에서 보여줘야합니다.
- 당연히 B 뷰에서 A 뷰로 돌아올 수 있어야합니다.
- 아래의 Struct 를 이용해야합니다.

```
struct FoodIngredient {  
    var name: String  
    var count: Int  
}
```

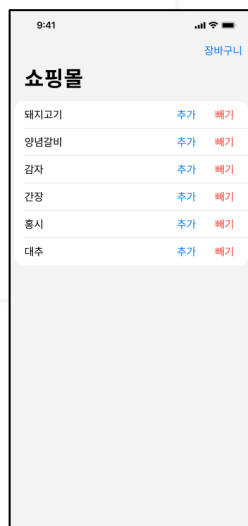
Chapter VII

Exercise 04: Shopping Mall

	Exercise 04
Project04_04	
Files to turn in: Swift Standard Library, SwiftUI	
Authorized functions: n/a	
Notes: n/a	


간단한 장바구니 앱을 만들어보세요.

- 첫번째 뷰는 쇼핑물 역할을 하고, 두번째 뷰는 장바구니 역할을 합니다.
- 첫번째 뷰에서 식재료 추가 버튼, 또는 빼기 버튼을 누르면 그 정보가 Class 에 전달되어야합니다.
- 첫번째 뷰에서 정보를 수정하고 오면, 두번째 뷰에 그 정보가 담겨있어야합니다.
- 두번째 뷰에서도 추가된 식재료들의 정보를 다룰 수 있어야하고 볼 수 있어야합니다.
- 당연히 음수는 취급하지 않습니다.



Chapter VII

Exercise 05: We are watching you

	Exercise 05
Project04_05	
Files to turn in: Swift Standard Library, SwiftUI	
Authorized functions: n/a	
Notes: n/a	

Exercise 04 의 내용을 EnvironmentObject 를 이용하여 구현해보세요.