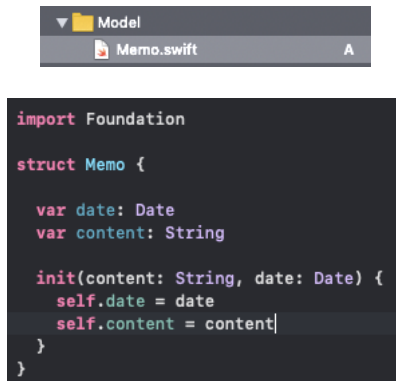


8Day_Summary

Date @Jan 21, 2020

Memo 모델만들기

13. 모델은 Struct로 생성한다! (Class vs Struct)



14. MemoComposeVC에서 save구현

1. save를 클릭했을때 MemoListViewController로 Memo모델(content, date)를 MemoListViewController로 넘겨야 한다.
2. 일단 contentview.text가 빈값일때 메모를 입력해달라고 알림창을 띄우자

```
@IBAction func save(_ sender: UIBarButtonItem) {
    guard let content = contentView.text,
          content.count > 0 else {
        let controller = UIAlertController(title: "알림", message: "메모를 입력해주세요", preferredStyle: .alert)
        let okAction = UIAlertAction(title: "확인", style: .default, handler: nil)
        controller.addAction(okAction)
        present(controller, animated: true, completion: nil)
        return
    }
}
```

| 💡 중요!

<ViewController1 ⇒ ViewController2 로 데이터로 넘길때>

storyboard?.instantiateViewController를 이용해서 인스턴스를 생성해 데이터를 넘겼죠?

반대로, <ViewController2 ⇒ ViewController1로 데이터를 넘길때>

dismiss, pop 메서드를 이용해서 쉽게 이동 가능한데

굳이 storyboard?.instantiateViewController를 이용해서 **ViewController1**의 인스턴스를 생성할 필요가 없다. (강하게 Coupling이 걸려있는 문제)

```
/*
    dismiss, pop등 돌아가는 메서드가 존재할때,
    전에 화면으로 돌아가갈때는 storyboard?.instantiateViewController로 인스턴스를 생성해서 돌아가도되긴 하는데,
    전 화면, 현재 화면이 강하게 Coupling이 걸려있어 좋지 않은 방법.
*/
```

```
let listVC = storyboard?.instantiateViewController(withIdentifier: "MemoListViewController") as? MemoListViewController {
    listVC.memos.append(memo)
    dismiss(animated: true, completion: nil)
}
```

쉽게, 되돌아갈때 데이터를 넘길때는 클로저를 이용하자! - 중요

3. 클로저를 이용해 MemoListViewController에 값을 넘겨보기

```
import UIKit

class MemoComposeViewController: UIViewController {

    @IBOutlet weak var contentTextView: UITextView!

    var addHandler: ((Memo) -> Void)?

    override func viewDidLoad() {
        super.viewDidLoad()
    }

    @IBAction func dismiss(_ sender: UIBarButtonItem) {
        dismiss(animated: true, completion: nil)
    }

    @IBAction func save(_ sender: UIBarButtonItem) {

        guard let content = contentTextView.text,
              content.count > 0 else {
            let controller = UIAlertController(title: "알림", message: "메모를 입력해주세요", preferredStyle: .alert)
            let okAction = UIAlertAction(title: "확인", style: .default, handler: nil)
            controller.addAction(okAction)
            present(controller, animated: true, completion: nil)
            return
        }

        let memo = Memo(content: content, date: Date())
        addHandler?(memo)

        dismiss(animated: true, completion: nil)
    }
}
```

Memo 모델을 addHandler라는 클로저를 통해서 전달한다

저장 후 MemoComposeViewController는 돌아야겠죠?

15. MemoListVC에 Memo 모델 데이터가 잘 들어 오는지 확인

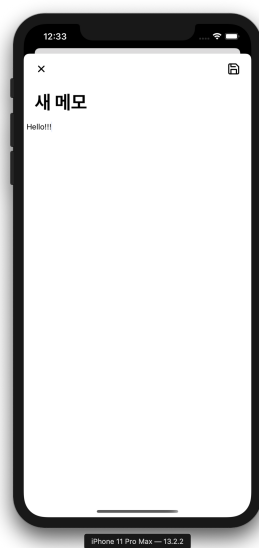
```
@IBAction func addMemo(_ sender: UIBarButtonItem) {
    if let composeVC = storyboard?.instantiateViewController(withIdentifier: String(describing: "MemoComposeViewController")) as? UINavigationController {
        present(composeVC, animated: true, completion: nil)
    }
}
```

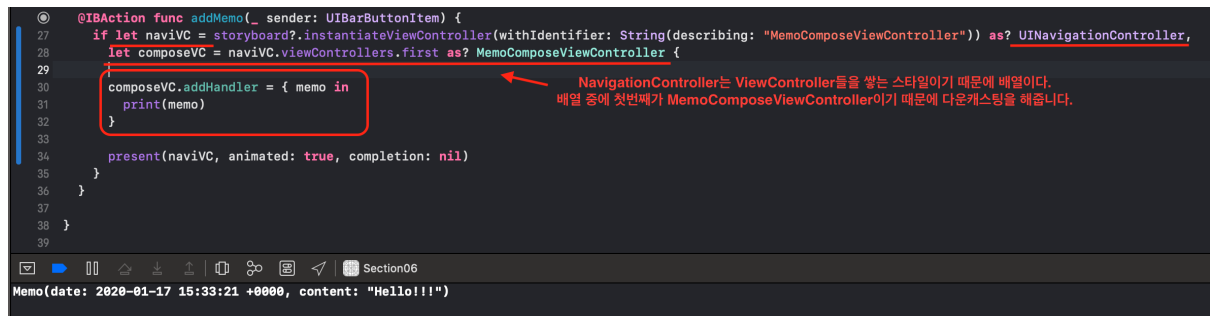
잘 보면 UINavigationController으로 다운 캐스팅을 했다.

이유는 아래 그림처럼 UINavigationController를 가지고 있기때문!



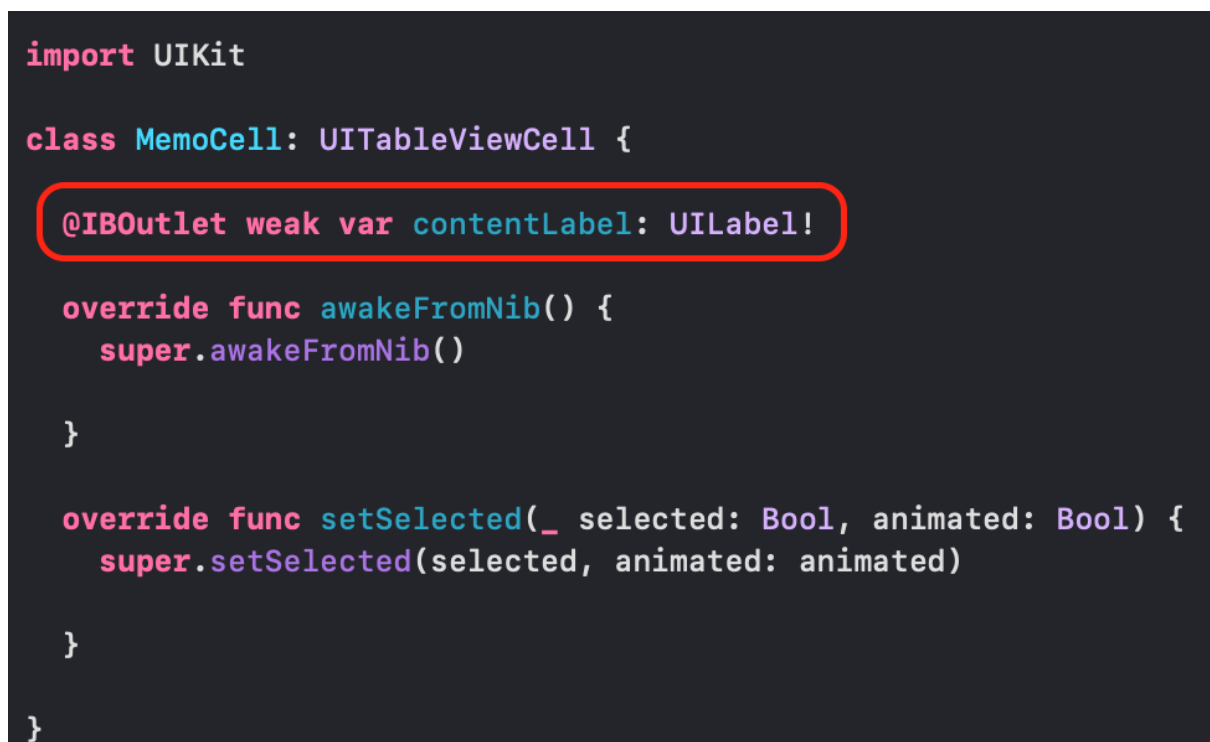
여기서 ComposeVC에서 addHandler에 접근해야되기 때문에 약간 수정하도록 하겠습니다.





NavigationController에서 MemoComposeViewController를 찾았고, composeVC.addHandler를 통해서 memo가 잘 들어온것을 볼 수 있다.

16. MemoCell에 contentLabel Outlet 추가



17. ComposeVC에서 넘어온 Memo 데이터 테이블 뷰에 넘기기

1. Memo 배열 만들기

```
var memos: [Memo] = []
```

2. addHandler에 넘어온 memo데이터 배열에 추가하기

새로 추가한 메모를 제일 첫번째에 보이기 위해서 0번째에 insert

```
composeVC.addHandler = { memo in
    self.memos.insert(memo, at: 0)
}
```

3. 배열에 넣었으니 테이블 뷰에 Cell에 바인딩 작업

```

extension MemoListViewController: UITableViewDataSource {

    func tableView(_ tableView: UITableView, numberOfRowsInSectionSection section: Int) -> Int {
        return memos.count
    }

    func tableView(_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell {
        guard let cell = tableView.dequeueReusableCell(withIdentifier: String(describing: MemoCell.self), for: indexPath) as? MemoCell else {
            return UITableViewCell()
        }

        cell.textLabel.text = memos[indexPath.row].content

        return cell
    }
}

```

여기까지 MemoListViewController 전체 소스

```

9  import UIKit
10
11  class MemoListViewController: UIViewController {
12
13  @IBOutlet weak var tableView: UITableView!
14
15  var memos: [Memo] = []
16
17  override func viewDidLoad() {
18      super.viewDidLoad()
19
20      setUpUI()
21  }
22
23  private func setUpUI() {
24      tableView.delegate = self
25      tableView.dataSource = self
26  }
27
28  @IBAction func addMemo(_ sender: UIBarButtonItem) {
29      if let navVC = storyboard?.instantiateViewController(withIdentifier: String(describing: "MemoComposeViewController")) as? UINavigationController,
30          let composeVC = navVC.viewControllers.first as? MemoComposeViewController {
31
32          composeVC.addHandler = { memo in
33              self.memos.insert(memo, at: 0)
34          }
35
36          present(navVC, animated: true, completion: nil)
37      }
38  }
39
40  }
41
42  extension MemoListViewController: UITableViewDataSource {
43
44      func tableView(_ tableView: UITableView, numberOfRowsInSectionSection section: Int) -> Int {
45          return memos.count
46      }
47
48      func tableView(_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell {
49          guard let cell = tableView.dequeueReusableCell(withIdentifier: String(describing: MemoCell.self), for: indexPath) as? MemoCell else {
50              return UITableViewCell()
51          }
52
53          cell.textLabel.text = memos[indexPath.row].content
54
55          return cell
56      }
57  }
58
59  extension MemoListViewController: UITableViewDelegate {
60
61  }
62
63

```

실행해 보면 데이터는 넘어 오고 에러도 나지 않는데, 테이블뷰에 Memo가 쌓이지 않는다!

테이블뷰에서 중요한 개념중에 데이터에 변화가 있을때 항상 **reload**를 시켜줘야된다!

```
tableView.reloadData()
```

테이블뷰에 넣어줄 데이터가 변화하는 곳이 어딜까?

ComposeVC에서 memo 데이터를 받아와 배열에 저장하는 곳이다!

```
composeVC.addHandler = { memo in
    self.memos.insert(memo, at: 0)
    self.tableView.reloadData()
}
```

다시 실행해보면 메모가 쌓인걸 확인 할 수 있다!

