

Xcode4로 시작하는 아이폰 프로그래밍(로드북, 2012)

Xcode4.2에 대한 추가 정보(2012/1/23)

이 책은 Xcode4.0과 Snow Leopard를 기반으로 설명하고 있습니다만. 최신 버전인 Xcode4.2에서도 거의 그대로 사용할 수 있습니다. 하지만 약간의 차이가 있으므로 이 차이점에 대해 다음과 같이 3가지 항목으로 나누어 설명한 문서를 배포하오니 참고 바랍니다.

- A) Xcode 4.2에서 바뀐 곳은 대해
- B) Xcode4.0의 샘플 파일을 Xcode 4.2에서 로드할 때의 주의사항
- C)이 책에서 설명한 내용을 Xcode 4.2에서 실행할 때의 주의사항

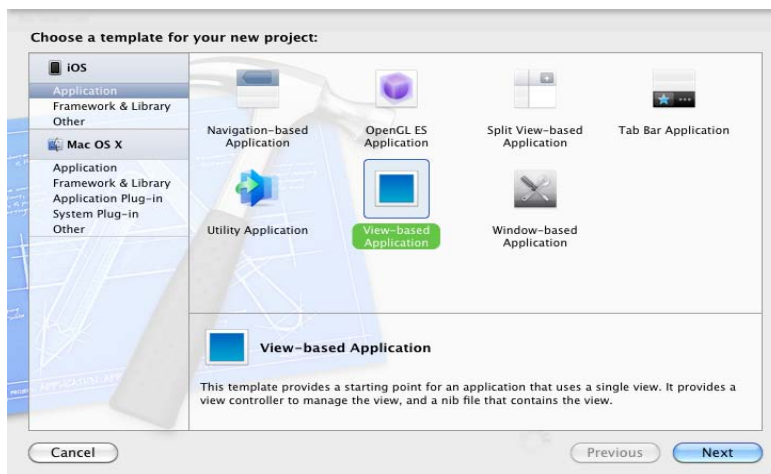
A) Xcode4.2 에서 바뀐 곳은 대해

우선 Xcode4.2 에서 크게 바뀐 부분에 대해 설명합니다.

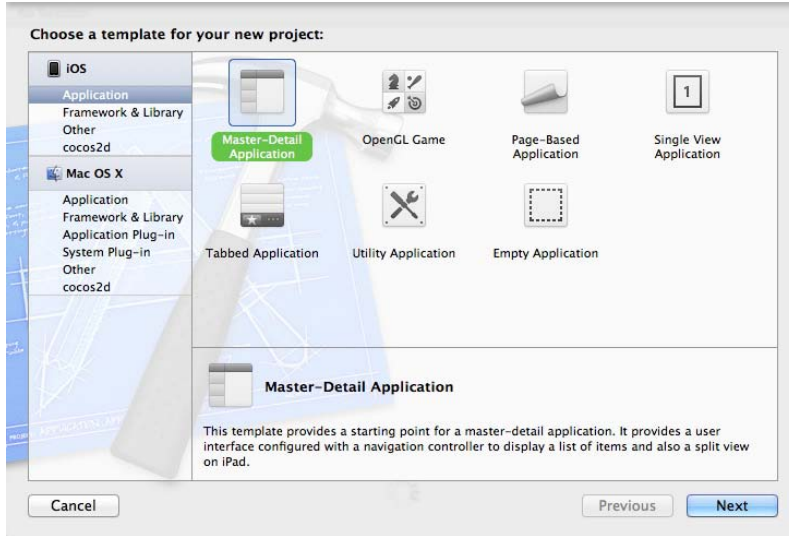
1) 새 프로젝트 작성용 템플릿의 명칭이 바뀌었습니다.

아이콘과 명칭은 바뀌었지만 기본은 같습니다.

- "View-based Application"은 "Single View Application"으로 바뀌었습니다.
- "Utility Application"은 그대로입니다.
- "Tab Bar Application"은 "Tabbed Application"으로 바뀌었습니다.
- "Navigation-based Application"은 "Master-Detail Application"으로 바뀌었습니다 (지금까지의 Navigation-based 와 Split View-based 가 하나로 통합된 것입니다).
- "OpenGL ES Application"은 "OpenGL Game"으로 바뀌었습니다.
- "Window-based Application"은 "Empty Application"으로 바뀌었습니다.
- "Page - Based Application"이 새롭게 추가되었습니다.



(Xcode4.0 의 Choose a template for your new project 화면)



(Xcode4.2 의 Choose a template for your new project 화면)

2) 라이브러리 패널이 조금 바뀌었습니다.

Stepper, Gesture Recognizer 가 추가되었습니다.

외형이 조금 바뀌었지만, 책의 내용은 그대로 사용할 수 있으므로 문제 없습니다.



3) Automatic Reference Counter(ARC) 기능이 추가되었습니다.

지금까지 다소 번잡했던 메모리 관리를 간편하게 만들어 주는 새로운 기능입니다.

Xcode 가 소스 파일을 보고 필요한 곳에 자동으로 release 를 넣어 컴파일을 하게끔 변경

되었기 때문에 프로그래머는 release 를 넣어줄 필요가 없어 훨씬 편해졌습니다. release 를 직접 넣으면 이중 입력이 되기 때문에 주의해야 됩니다.

하지만, 새 프로젝트 대화 상자에서 **ARC 의 체크를 OFF 로 하고** 프로젝트를 시작하면 Xcode4.0 에서처럼(책에서와 같은 방법으로) 실행할 수 있습니다. 또한 ARC 를 ON 으로 하고 시작한 프로젝트도 **"Build Setting" 탭에서 "Apple LLVM compiler 3.0"> "Objective - C Automatic Reference Counting"을 "No"로 설정**하면, 역시 책에서와 같은 방법으로 실행할 수 있습니다.

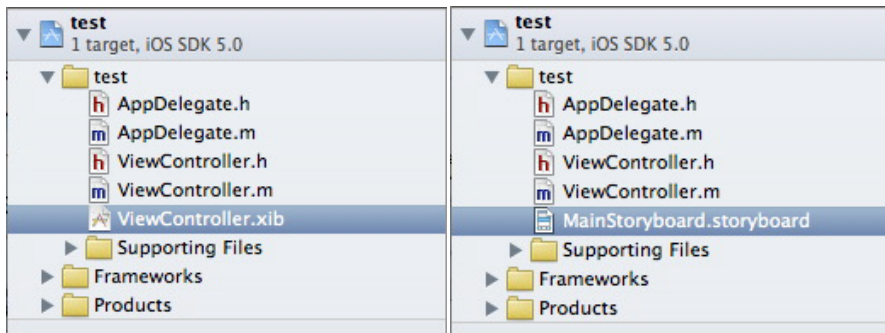
4) StoryBoard 기능이 추가되었습니다.

XIB 가 진화한 것으로, 한 번에 여러 화면을 만들 수 있게 되었습니다.

또한, 화면 전환 기능을 마우스의 드래그만으로 프로그래밍 없이 만들 수 있습니다.

이 기능을 사용할 때는 xib 파일 대신에 storyboard 파일이 사용됩니다.

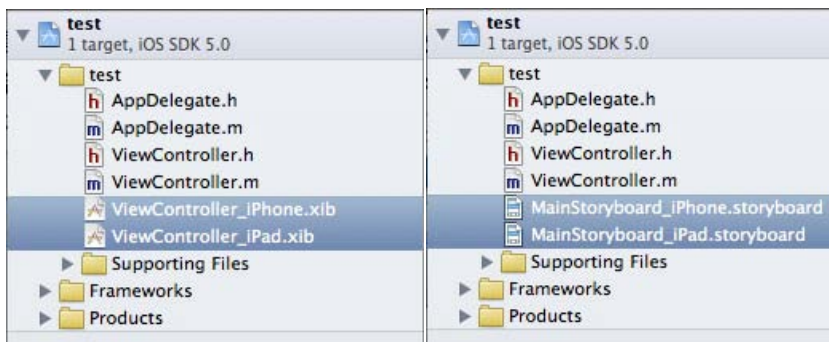
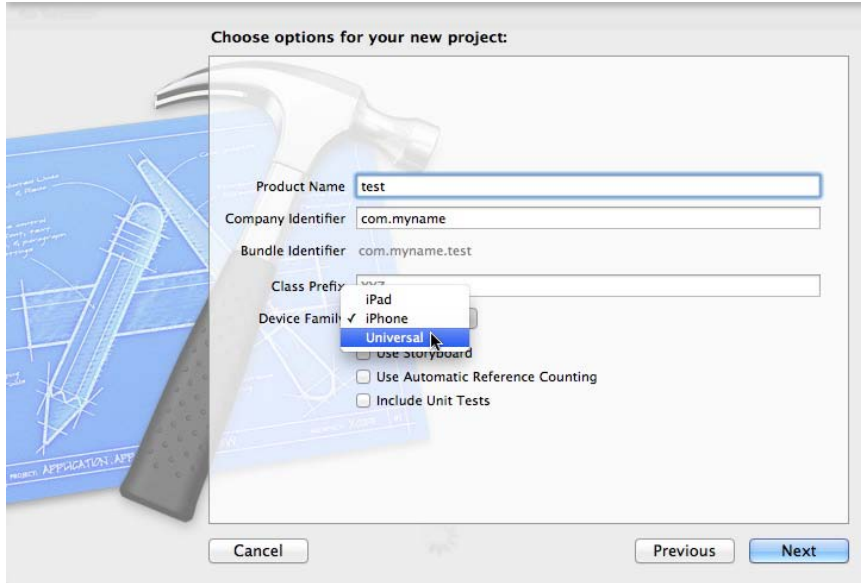
참고로 새 프로젝트 대화 상자에서 **"Use Storyboard"의 체크를 OFF 하고** 프로젝트를 시작하면 책에서와 같은 방법으로 실행할 수 있습니다.



5) iPhone 과 iPad 화면을 동시에 설계할 수 있게 되었습니다.

새 프로젝트 대화 상자의 "Device Family"에서 "Universal"을 선택하고 시작하면 iPhone 과 iPad 모두의 xib 파일(또는 storyboard 파일)이 만들어진 상태에서 시작합니다.

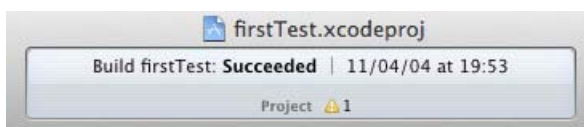
iPhone 전용 앱(또는 iPad 전용)을 개발할 때는 iPhone(또는 iPad)을 선택하면 xib 파일(또는 storyboard 파일)이 하나만 생성됩니다(책에서 쓰는 방법).



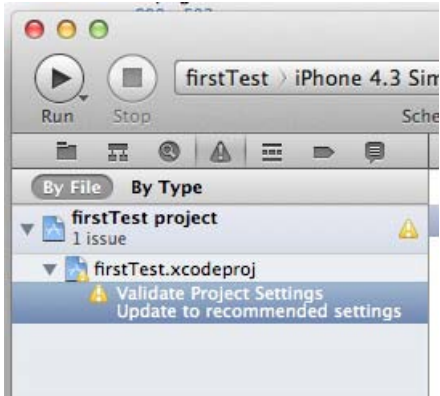
B) Xcode4.0 의 샘플 파일을 Xcode4.2 에서 로드할 때의 주의사항

※ 이 책을 따라 하면서 Xcode4.0 으로 만든 샘플 파일을 Xcode4.2 에서 가져오면 경고가 나타나지만 거의 그대로 사용할 수 있습니다. 또한 설정을 바꿔주면 경고를 없앨 수도 있습니다.

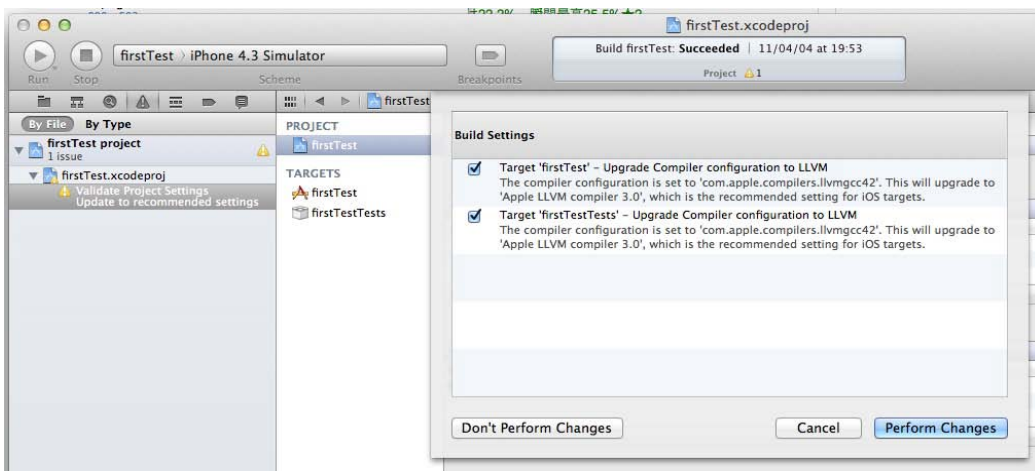
1) 샘플 파일을 로드할 때 다음과 같이 **"Project! 1"**이라는 경고가 뜹니다. 이것은 "컴파일러 설정이 오래되었기 때문에, 업그레이드를 해두는 편이 좋습니다."라는 의미의 경고입니다.



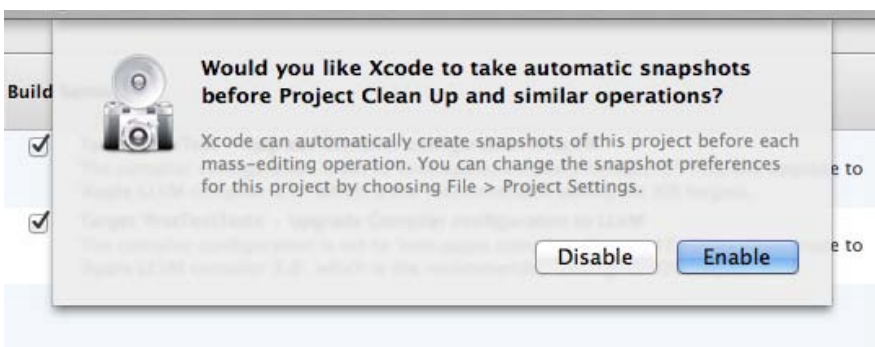
2) 이 상태에서도 별 문제없이 앱을 만들 수 있지만 바로 설정을 수정할 수도 있습니다. 위의 그림에서 **"Project! 1"** 을 클릭하면 아래와 같이 **"Validate Project Settings Update to recommended settings"** 가 뜨는데, 이것을 클릭합니다.



3) 그러면 다음과 같이 "컴파일러 설정을 권장 설정으로 업그레이드 하시겠습니까?"라는 메시지가 나타나고, "**Perform Changes(변경 사항 실행)**"를 클릭합니다.



4) 아래와 같은 대화 상자(변경사항을 저장 하시겠습니까?)가 나타나고 "**Enable**"을 클릭하면 변경이 실행되고 샘플 파일에서 경고가 표시되지 않습니다("File> Restore snapshot"으로 실행을 취소할 수도 있습니다).



※ CHAPTER 10-1 [Table View]를 Xcode4.2 에서 실행하면 경고 메시지가 한가지 더 표시됩니다. 이것은 "**initWithFrame : reuseIdentifier** :라는 명령은 오래 되었기 때문에 다른 방법으로 만들어주세요" 라는 경고입니다.

이것은 다음과 같이 수정하면 경고가 없어집니다.

- (UITableViewCell *) tableView : (UITableView *) tableView cellForRowAtIndexPath : (NSIndexPath *) indexPath

의

```
if (cell == nil) {  
    cell = [[[UITableViewCell alloc] initWithFrame : CGRectZero reuseIdentifier : CellIdentifier]  
    autorelease];  
    을
```

을

```
if (cell == nil) {  
    cell = [UITableViewCell alloc] initWithStyle : UITableViewCellStyleDefault reuseIdentifier :  
    CellIdentifier];  
    으로 변경하면 됩니다.
```

으로 변경하면 됩니다.

C) 이 책에서 설명한 내용을 Xcode 4.2 에서 실행할 때의 주의사항

※ 기본적으로 다음 두 가지에 대해 주의하면 이 책에서 제시하는 설명대로 앱을 만들 수 있습니다.

- Xcode4.2 에서 바뀐 명칭의 템플릿을 적용해야 합니다(A.설명 참조)
- 새 프로젝트를 만들 때 몇 가지 체크 상자에 OFF 를 해주어야 합니다(B.설명 참조).

CHAPTER 4-3, 5-1,5-2, 5-3, 5-4, 5-5, 5-6, 5-7, 5-8, 5-9, 6-1, 6-2, 6 -3, 6-4, 7-1, 7-2, 7-3, 7-4, 8-1, 8-2, 8-3, 8-4

PRACTICE [1]

[View - based Application 을 선택합니다.

↓ **[Single View Application]**을 선택합니다.

PRACTICE [1]

"Next" 버튼을 클릭하여 저장합니다.

↓ **"Device Family"에서 "iPhone"을 선택하고 "Use Storyboard"와 "Use Automatic Reference Counting" 확인란을 OFF 하고 "Next" 버튼을 클릭하여 저장합니다.**

※ARC 에 체크를 ON 으로 하고 시작한 프로젝트에서 ARC 를 OFF 하려면, Build Setting 에서 설정을 변경합니다.

- 1) 프로젝트를 엽니다.
- 2) 맨 왼쪽 탐색기 영역에서 상단에 있는 파란색 프로젝트 파일을 선택합니다.
- 3) 오른쪽에 나타나는 화면에서 "Build Setting" 탭을 선택하고 "Apple LLVM compiler 3.0" "Objective-C Automatic Reference Counting"의 "YES"를 "No"로 전환합니다(항목이 많이 있기 때문에 Reference 에서 검색하면 빨리 찾을 수 있습니다).

이렇게 하면, 책에서처럼 release 를 쓰는 방법으로 만들 수 있습니다.

CHAPTER 9-1 [Utility Application]

※ xib 파일 이름이 바뀌었습니다.

PRACTICE [1]

"Next"버튼을 클릭하여 저장합니다.

↓ "Device Family"에서 "iPhone"을 선택하고 "Use Storyboard"와 "Use Automatic Reference Counting"확인란을 OFF 하고 "Next" 버튼을 눌러 저장합니다.

PRACTICE [2]

"MainView.xib"를 클릭하여

↓ "MainViewController.xib"를 클릭하여

PRACTICE [6]

"FlipsideView.xib"를 클릭하여

↓ "FlipsideViewController.xib"를 클릭하여

PRACTICE [11]

[controller release];

↓ 이 구문은 제거합니다.

CHAPTER 9-2 [Tab Bar Application]

※ xib 파일 이름과 AppDelegate 의 파일 이름이 바뀌었습니다.

PRACTICE [1]

"Tab Bar Application"을 선택합니다.

↓ "Tabbed Application" 을 선택합니다.

PRACTICE [1]

"Next"를 클릭하여 저장합니다.

↓

"Device Family"에서 "iPhone"을 선택하고 "Use Storyboard"와 "Use Automatic Reference Counting" 체크 상자를 OFF 로 하고, "Next"를 클릭하여 저장합니다.

PRACTICE [2]

"FirstView.xib"를 클릭해서

↓ "FirstViewController.xib"를 클릭해서

PRACTICE [5]

"SecondView.xib"를 클릭해서

↓ "SecondViewController.xib"를 클릭해서

PRACTICE [10] [11]

"tabBarAppDelegate.h"를 임포트합니다.

↓ **"AppDelegate.h"**를 임포트합니다.

```
# import "tabBarAppDelegate.h"
```

↓ **# import "AppDelegate.h"**

```
tabBarAppDelegate * appDelegate = (tabBarAppDelegate *) [UIApplication sharedApplication]
delegate];
```

↓ **AppDelegate * appDelegate = (AppDelegate *) [UIApplication sharedApplication]
delegate];**

P343

PRACTICE [5] [6] 단계는 모두 삭제하고 다음과 같이 변경합니다.

PRACTICE [5] AppDelegate.m의 탭 표시줄에 새로 만든 화면을 추가하는 프로그램을 작성합니다. "AppDelegate.m" 파일을 선택합니다.

1) # import "SecondViewController.h" 다음 줄에

import "myViewController.h"

를 추가합니다.

2) -(BOOL) application : (UIApplication *) application didFinishLaunchingWithOptions :
에서,

```
UIViewController * viewController2 = [[[SecondViewController alloc] initWithNibName : @
"SecondViewController"bundle : nil] autorelease];
```

다음 줄에

**UIViewController * viewController3 = [[[myViewController alloc] initWithNibName : @
"myViewController"bundle : nil] autorelease];**

를 추가합니다.

```
3) self.tabBarController.viewControllers = [NSArray arrayWithObjects : viewController1,
viewController2, nil];
```

을

```
self.tabBarController.viewControllers = [NSArray arrayWithObjects : viewController1,
viewController2, viewController3 , nil];
```

로 수정합니다.

4) PRACTICE [6] myViewController.m에 탭 표시줄 버튼에 문자를 표시하는 프로그램을 작성합니다. "myViewController.m"파일을 선택합니다.


```
- (id) initWithNibName : (NSString *) nibNameOrNil bundle : (NSBundle *) nibBundleOrNil
중
if (self) {
}
부분을
if (self) { self.title = NSLocalizedString (@ "Third"@ "Third"); }
로 변경합니다.
```

CHAPTER 10-1 [Table View]

※ cell 을 만드는 방법이 바뀌었습니다.

PRACTICE [1]

[View-based Application] 을 선택합니다.

↓ **[Single View Application]** 을 선택합니다.

PRACTICE [1]

"Next" 를 클릭하여 저장합니다.

↓ **"Device Family" 에서 "iPhone" 을 선택하고 "Use Storyboard" 와 "Use Automatic Reference Counting" 확인란에 OFF 하고 "Next" 를 클릭하여 저장합니다.**

PRACTICE [6]

```
if (cell == nil) {
cell = [[[UITableViewCell alloc] initWithFrame : CGRectZero reuseIdentifier : CellIdentifier]
autorelease];
↓
```

↓

```
IF (cell == nil) { cell = [UITableViewCell alloc] initWithStyle : UITableViewCellStyleDefault
reuseIdentifier : CellIdentifier];
```

P362

※ 기본 파일 이름이 **MasterViewController** 로 바뀌었습니다.

PRACTICE [1]

[Navigation-based Application] 을 선택합니다.

↓ **[Master-Detail Application]** 을 선택합니다.

PRACTICE [1]

"Next" 를 클릭하여 저장합니다.

↓ **"Device Family" 에서 "iPhone" 을 선택하고 "Use Storyboard" 와 "Use Automatic Reference Counting" 확인란에 OFF 하고 "Next" 를 클릭하여 저장합니다.**

PRACTICE [2]

.h 파일(RootViewController.h) 을 선택합니다.

↓

.h 파일(**MasterViewController.h**)을 선택합니다.

@ interface RootViewController : UITableViewController

↓

@ interface **MasterViewController** : UITableViewController

PRACTICE [3]

.m 파일(RootViewController.m)을 선택하고

↓

.m 파일(**MasterViewController.h**)을 선택하고

CHAPTER 10-2 [Navigation-Based Application]

※ cell 을 만드는 방법이 바뀌어, 하위계층 화면이 처음부터 제공되게 되었습니다.

PRACTICE [1]

[Navigation-based Application]을 선택합니다.

↓ **[Master - Detail Application]**을 선택합니다.

PRACTICE [1]

[Next]를 클릭하여 저장합니다.

↓ **"Device Family"에서 "iPhone"을 선택하고 "Use Storyboard"와 "Use Automatic Reference Counting"확인란을 OFF 하고** [Next]를 클릭하여 저장합니다.

PRACTICE [2] 단계는 삭제.

Master-Detail Application 은 처음부터 하위계층 화면이 제공되기 때문에 직접 만들 필요가 없습니다. 하위계층 화면의 이름은 "DetailViewController"입니다.

PRACTICE [3]

"myViewContoller.xib"를 클릭해서

↓ **"DetailViewController.xib"**를 클릭해서

PRACTICE [4]

.h 파일(myViewContoller.h)을 선택합니다.

↓

.h 파일(**DetailViewController.h**)을 선택합니다.

PRACTICE [6]

.m 파일(myViewContoller.m)을 선택합니다.

↓

.m 파일(**DetailViewController.m**)을 선택합니다.

PRACTICE [7]

.h 파일(RootViewController.h)을 선택합니다.

↓

.h 파일(**MasterViewController.h**)을 선택합니다.

PRACTICE [8] 단계는 삭제(이미 임포트 되어 있습니다)

PRACTICE [9]

.m 파일(RootViewController.m)을 선택합니다.

↓

.m 파일(**MasterViewController.m**)을 선택합니다.

PRACTICE [11]

```
myViewController * detailViewController = [myViewController alloc] initWithNibName : @"myViewController" bundle : nil];
```

```
detailViewController.dispStr = myData objectAtIndex : indexPath.row];
```

↓ ↓

```
self.detailViewController = [[[ DetailViewController alloc] initWithNibName : @"DetailViewController" bundle : nil] autorelease]; self.detailViewController . dispStr = myData objectAtIndex : indexPath.row];
```

CHAPTER 10-3 [Cell 지정]

※ cell 을 만드는 방법이 바뀌어, 하위계층 화면이 처음부터 제공되게 되었습니다.

PRACTICE [1]

[Navigation-based Application]을 선택합니다.

↓

[**Master-Detail Application**]을 선택합니다.

PRACTICE [1]

[Next]를 클릭하여 저장합니다.

↓ "**Device Family**"에서 "**iPhone**"을 선택하고 "**Use Storyboard**"와 "**Use Automatic Reference Counting**" 확인란에 **OFF** 하고 [Next]를 클릭하여 저장합니다.

PRACTICE [5]

그리고 나서 [File 's Owner]를 오른쪽 버튼으로 클릭합니다. [View] 오른쪽에 있는 [○]를 드래그해서 선을 잡아 늘려 [My Cell View Controller]로 연결하면 직접 작성한 셀과 직접 작성한 셀의 클래스가 연결됩니다.

↓

제거합니다. (4.2 에는 view 가 나오지 않아 셀을 직접 지정하는 방식을 사용합니다)

PRACTICE [7]

루트 화면(RootViewController.h)에서

↓

루트 화면(**MasterViewController.h**)에서

PRACTICE [7]

```
if (cell == nil) {  
    UIViewController * vc;  
    vc = [UIViewController alloc] initWithNibName : @ "myCell"bundle : nil];  
    cell = (myCellViewController *) vc.view;  
}
```

↓

```
IF (cell == nil) { UINib * nib = UINib nibWithNibName : @ "myCell"bundle :  
nil]; NSArray * array = nib instantiateWithOwner : nil options : nil]; cell =  
array objectAtIndex : 0]; }
```