

아이폰 키보드를 감추는 iOS 5 코드 만들기

아이폰 앱에서 데이터를 입력하려고 하면(보통 문자열 입력 부분을 터치할 때) 자동으로 화면에 키보드가 표시된다. 그러나 앞 장에서 언급했듯이 유저의 입력이 끝났을 때 자동으로 키보드가 사라지지 않는다.

다른 아이폰 앱들을 사용해 보면 키보드의 리턴키를 누르거나 유저 인터페이스의 다른 부분을 건드리면 보통 키보드가 화면에서 사라진다. 사실 이러한 것은 해당 애플리케이션의 개발자가 이러한 기능을 구현해 놓았기 때문에 가능한 것이다. 이번 장에서는 여러분의 iOS 5 앱에 이러한 기능을 추가하는 방법에 대해 알아보겠다.

12.1 예제 앱 만들기

이 책을 앞부분부터 읽어왔다면 앞 장의 예제 프로그램을 이용하여 이번 장의 예제를 수행할 수 있을 것이다. 그렇지만 순서대로 읽지 않고 이 장을 참조하는 경우라면 예제 애플리케이션을 만들어야 할 것이다.

Xcode를 실행한 후 Single View based iPhone iOS application을 만들고 hideKeyboard를 클래스 접두사 및 product 이름으로 한다. Xcode 화면에서 hideKeyboardViewController.xib 파일을 선택하여 유저 인터페이스를 수정한다. 인터페이스 빌더 화면에

서 Object Library의(View → Utilities → Show Object Library) Text Filed 오브젝트를 View 영역으로 드래그한다.

유저 인터페이스 디자인이 끝났으면 뷰 컨트롤러의 코드에서 텍스트 필드를 참조할 수 있도록 아웃렛을 만든다. Xcode 화면에서 hideKeyboardViewController.h 파일을 선택한 후 다음과 같이 수정한다.

```
#import <UIKit/UIKit.h>

@interface hideKeyboardViewController : UIViewController {
    UITextField *textField;
}
@property (strong, nonatomic) IBOutlet UITextField *textField;
@end
```

다음은 hideKeyboardViewController.m 파일에 새로운 아웃렛을 위한 통합된 접근자를 만든다.

```
#import "hideKeyboardViewController.h"
@implementation hideKeyboardViewController
@synthesize textField;
.
.
@end
```

다시 hideKeyboardViewController.xib를 선택하여 텍스트 필드를 뷰 컨트롤러의 아웃렛과 연결시킨다. Ctrl 키를 누른 상태에서 File's Owner 아이콘을 클릭한 후 View의 텍스트 필드 오브젝트로 드래그한다. 결과 메뉴에서 textField 아웃렛을 선택한다.

12.2 사용자가 리턴키를 터치할 때 키보드 감추기

다음 단계는 유저가 키보드의 리턴키를 터치하였을 때 키보드를 사라지게 하는 코드를 추가하는 것이다. 이를 위해 리턴키가 눌렸을 때 텍스트 필드의 resign the first

responder를 수행하는 메서드를 만들어야 한다.

hideKeyboardViewController.h 파일을 수정하여 textFieldReturn이라는 이름의 메서드를 선언한다.

```
#import <UIKit/UIKit.h>

@interface hideKeyboardViewController : UIViewController {
    UITextField *textField;
}
@property (strong, nonatomic) IBOutlet UITextField *textField;
-(IBAction)textFieldReturn:(id)sender;
@end
```

메서드 선언을 하였으니 이제 hideKeyboardViewController.m 파일에서 메서드를 구현해야 한다.

```
#import "hideKeyboardViewController.h"

@implementation hideKeyboardViewController
@synthesize textField;

-(IBAction)textFieldReturn:(id)sender
{
    [sender resignFirstResponder];
}
.
.
.
@end
```

위에서 메서드는 이벤트가 호출된 오브젝트의 resignFirstResponder 메서드를 호출한다. first responder는 유저가 현재 상호작용을 하고 있는 오브젝트를 의미한다(예를 들면, 아이폰 화면에 표시된 가상 키보드).

코드를 작성하였으면 이제 적절한 시점에 메서드가 호출될 수 있도록 유저 인터페이스

와 코드를 연결해야 한다. 이는 인터페이스 빌더에서 수행되므로 `hideKeyboardViewController.xib`를 다시 한 번 선택한다.

텍스트 필드를 선택한 후 오른쪽 화면에 Connections Inspector를 표시한다(View → Utilities → Connections Inspector). Did End on Exit 이벤트의 오른쪽의 원을 클릭한 후 File's Owner 아이콘까지 드래그한 후 메서드 목록에서 `textFieldReturn`을 선택한다.

Xcode 툴바의 Run 버튼을 클릭한다. iOS 시뮬레이터에서 애플리케이션이 실행되면 텍스트 필드를 선택하여 키보드가 보이는 것을 확인하고 리턴키를 터치한다. 그럼 뷰에서 키보드가 사라질 것이다.

12.3 사용자가 백그라운드를 터치했을 때 키보드 감추기

다음은 사용자가 화면의 뷰를 터치했을 때 키보드가 사라지게 하는 방법에 대해 알아보자. 먼저 이 작업을 수행할 액션 메서드를 만들어야 한다. Xcode에서 `hideKeyboardViewController.h` 파일을 선택한 후 `backgroundTouched`라는 액션 메서드를 선언한다.

```
#import <UIKit/UIKit.h>

@interface hideKeyboardViewController : UIViewController {
    UITextField *textField;
}
@property (nonatomic, strong) IBOutlet UITextField *textField;
- (IBAction)textFieldReturn:(id)sender;
- (IBAction)backgroundTouched:(id)sender;
@end
```

`hideKeyboardViewController.m` 파일을 선택하고 `textField` 오브젝트의 `resignFirstResponder` 메서드를 호출하는 부분을 구현한다.

```
#import "hideKeyboardViewController.h"
@implementation hideKeyboardViewController
@synthesize textField;
```

```

-(IBAction)textFieldReturn:(id)sender
{
    [sender resignFirstResponder];
}
-(IBAction)backgroundTouched:(id)sender
{
    [textField resignFirstResponder];
}
.
.
@end

```

액션 메서드의 코드를 작성하였으면 이제 사용자가 뷰를 터치하였을 때 이 메서드가 호출 되도록 설정해야 한다. 이는 인터페이스 빌더를 통해 수행해야 하므로 `hideKeyboardViewController.xib` 파일을 선택하도록 한다.

키보드가 사라지도록 하기 위해서는 사용자가 뷰를 터치하였을 때 액션 메서드가 호출 되도록 설정해야 한다. 기본적으로 인터페이스 빌더는 인터페이스의 배경으로 `UIView` 클래스의 인스턴스를 만든다. 그러나 불행하게도 `UIView` 인스턴스는 이벤트에 응답할 수 없으므로 액션 메서드를 호출하도록 설정할 수 없다. 그러므로 뷰의 클래스를 `UIControl` 클래스의 인스턴스로 변경하여야 한다. 가운데 화면의 `File's Owner`와 `FirstResponder` 아이콘 아래에 `View`가 있다. 이는 인터페이스의 뷰를 의미한다. 이 아이콘을 선택한 후 오른쪽 화면에서 `Identity Inspector`를 표시한 후 `Inspector` 화면에서 `Class` 설정을 `UIView`에서 `UIControl`로 변경한다.

이제 배경 뷰의 클래스가 `UIControl`로 바뀌었다(`UIControl`은 `UIView`의 서브클래스다). 따라서 `backgroundTouched` 액션 메서드로 연결을 설정할 수 있다. 유저 인터페이스의 배경을 선택한 후 `Connections Inspector` 화면을 표시한다(`View` → `Utilities` → `Show Connections Inspector`). `Touch Down` 이벤트의 오른쪽 원을 클릭하여 `File's Owner` 아이콘으로 드래그한다. 마우스 버튼을 놓은 후 메뉴에서 `backgroundTouched` 메서드를 선택한다. 디자인을 저장한 후 빌드를 실행하고 애플리케이션을 실행한다. iOS 시뮬레이터가 시작되고 텍스트 필드를 선택하면 키보드가 나타날 것이다. 배경화면을 터치하면 키보드가 사라져야 한다.

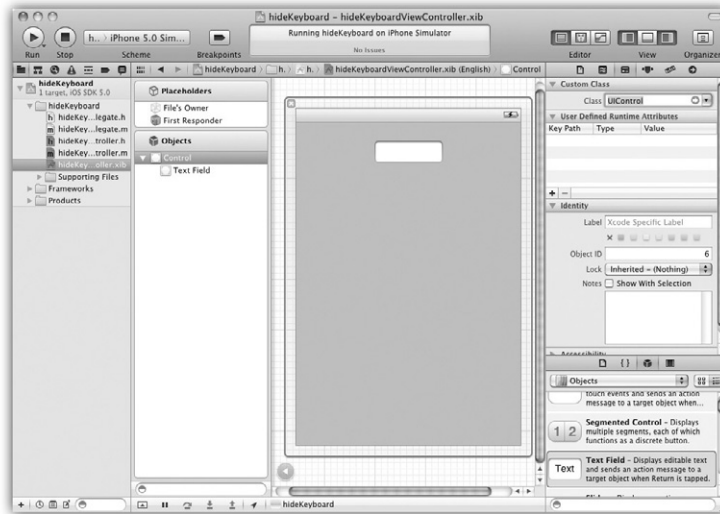


그림 12-1

12.4 요약

사용자가 입력을 할 경우 아이폰의 화면 키보드가 나타나는 것은 자동이지만, 키보드가 사라지는 것은 애플리케이션의 처리에 의존한다. 일반적으로 사용자가 키보드의 리턴키를 누르거나 배경화면의 다른 부분을 누르면 키보드는 사라진다. 이번 장에서는 이 두 가지 기능의 구현에 대해 알아보았다.