

Favori Kitaplar

Öğrenci kaynakları klasörünüzde "FavoriteBooks" adlı bir başlangıç projesi bulacaksınız. Bakmak için bir dakikanızı ayırın. Bir **ViewController** elemanı, kullanıcının bir kitap hakkında ayrıntı girmesine izin veren bir form içerir. Bu dökümanda, formu statik **TableView** ile değiştirecek, ana kitap listesinden kitap silme özelliğini ekleyecek ve ana listedeki her kitabın ayrıntılarını daha iyi görüntülemek için özel bir **TableView** elemanı oluşturacaksınız.

Adım 1: Statik TableView Oluşturma

- Bir **TableView** elemanını Storyboard'a sürükleyin. **TableView** seçiliyken, içeriği, statik **Cell** elemanlarını gruplanacak şekilde ayarlamak için "Attributes Inspector"ı kullanın.
- Her bir **TextField** elemanının kendi hücreinde ve her hücrenin (cell) kendi bölümünde (section) olduğu mevcut formunuzu modelleyen bir form oluşturacaksınız. Beş bölüm oluşturacaksınız. Film şeridinde bölüm başlığı başlıklarını "Başlık", "Yazar", "Tür" ve "Uzunluk" olarak ayarlayabilirsiniz. Son bölüm Kaydet düğmesi içindir.
- İlk dört hücrenin her birine bir **TextField** ve son hücreye bir buton yerleştirin.
- "BookFormTableViewCell" adlı yeni bir **UITableViewController** elemanı oluşturun.
- **TableView** elemanınızda yazan kodların şu an çoğu gereksizdir. `viewDidLoad()` fonksiyonu dışında tümünü silin.
- Storyboard'da **TableView** elemanının sınıfını ayarlayın ve **TextField** elemanlarınız için çıkışlar ve "Kaydet" düğmeniz için bir aksiyon oluşturun.

Adım 2: BookFormViewController'ı BookFormTableViewCell ile Değiştirin

- Formu storyboard'daki **TableView** elemanını değiştirmeye hazırlanırken **BookFormViewController** işlevlerini **BookFormTableViewCell** elemanına aktarın. `viewDidLoad()` fonksiyonundan çağırdığınız **BookFormTableViewCell** elemanına bir `updateView()` fonksiyonu ekleyerek başlayın. Tam olarak **BookFormViewController** elemanındaki karşılığı gibi davranmalıdır. Daha sonra **saveButtonTapped** fonksiyonunu doldurun; aynı zamanda **BookFormController** elemanındaki karşılığı gibi davranmalıdır. Bu fonksiyonları kopyalayıp doğrudan yeni dosyaya yapıştırabilirsiniz.
- **saveButtonTapped(_:)** fonksiyonunu uygularken segue işlemi yapmanız gerektiğini unutmayın. Segue elemanının çalışması için, storyboard'daki **BookFormTableViewCell** elemanınıza bağlamanız ve segue'ye bir tanımlayıcı (identifier) vermeniz gerekir.
- **Book?** türündeki **BookFormTableViewCell** elemanının üstüne bir **Book** özelliği ekleyin. Storyboard'daki **ViewController** elemanını kaldırın ve **TableViewCell** elemanını ekleyin. Az önce sildiğiniz **ViewController** elemanı ile olan segueleri hatırlıyor musunuz? Tanımlayıcılarını doğru şekilde ayarladığınızdan emin olarak, aynı iki sekmeyi bu

TableViewController elemanında yeniden oluşturun.

- Projeyi çalıştırın. Uygulamanız büyük bir ihtimalle daha önce olduğu gibi çalışıyordu. Uygulamanız aynı şekilde çalışsa da artık farklı bir forma sahip!

Adım 3: Kipları Silin

- Kullanıcı bir kitabı listeden kaldırmak isterse ne olur? **BookTableViewController** gidin ve bir **TableViewController** fonksiyonu olan **tableView(_:commit:forRowAt:)** fonksiyonunu ekleyin.
- Bu fonksiyona, düzenleme stilinin **.delete** olup olmadığını denetleyecek denetim akışı ekleyin; öyleyse, **books** dizisindeki söz konusu **indexPath** değerine karşılık gelen **Book** nesnesini kaldırın, ardından aynı **indexPath** değerindeki **TableViewController** elemanının satırını kaldırın.
- Uygulamayı çalıştırın ve kitapları listeden silip silmeyeceğinizi görün.

Adım 4: Özel Kitap Hücresi Oluşturma

- Belki kitap bilgilerinin gösterimini özelleştirmek ister misiniz? **BookTableViewController** elemanlarınızda, prototip hücresinin stilini "Custom" olarak değiştirin. Örneğin, üstte, kitap başlığı için bir etiket isteyebilirsiniz; ve altında, yazar, tür ve uzunluk için biraz daha küçük ve hafif girintili etiketler isteyebilirsiniz. Veya başlık etiketine ek olarak, üç satır halinde düzenlenmiş altı etiketiniz olabilir; soldaki etiketler "Yazar", "Tür" ve "Uzunluk"; ve sağdaki etiketler yazar, tür ve uzunluğa karşılık gelen gerçek değerleri yazar. Devam edin ve kendi özel prototip hücrenizi tasarlayın. **Book** nesnesinde bulunan tüm özellikleri gösterdiğinizden emin olun.
- Elemanlarınız için çıkışlar oluşturmak için özel bir **UITableViewCell** alt sınıfına ihtiyacınız olacaktır. **UITableViewCell** türünden, "BookTableViewCell" adında yeni bir dosya oluşturun.
- **BookTableViewCell** elemanında, her kitap hücresi için yazıları ayarlayacak olan **update(with book: Book)** adlı bir fonksiyon oluşturun.
- **BookTableViewController** elemanına geri döndüğünüzde, **tableView(_:cellForRowAt:)** fonksiyonunuzda küçük değişiklikler yapacaksınız. İlk olarak, **cell** sabitini **tableView.dequeueReusableCell(withIdentifier:"BookCell") as! BookTableViewCell** olarak yazalım. Daha sonra **cell.textLabel?.text** ve **cell.detailTextLabel?.text** değerlerini ayarlamak yerine, özel hücrenizin **update(with:)** fonksiyonunu çağırabilir ve **Book** nesnesine iletebilirsiniz.
- Uygulamayı çalıştırın ve hücrelerinizin görünümünün nasıl değiştiğini görün.

Tebrikler! "Favori Kitaplar" projesine eklediğiniz **TableView** elemanlarıyla, kullanıcıya çok daha iyi bir deneyim sağlayacaktır.