BEGINNING

CORE DATA

Hands-On Challenges

Beginning Core Data

Luke Parham

Notice of Rights

Notice of Liability

Trademarks

3 Challenge #3: Fixing the Sort

By Luke Parham

As you might have noticed, the names are sorted, but it seems lowercase and uppercase names are weirdly split into two sections.

As you can see, the lowercase "michael" is put at the end. How strange...

Changing the Sort Method

It turns out that by default, `NSSortDescriptor` uses the `compare(_:)` method to compare the values of the attribute you're sorting by. In this case, the result is what you see.

What you actually want to see is people ordered by the first letter of their first name, regardless of the casing.

To accomplish this, go to `reloadData(_:)` and replace

```
let sort = NSSortDescriptor(
    key: #keyPath(Person.name), ascending: true)
```

with

```
let sort = NSSortDescriptor(
    key: #keyPath(Person.name),
    ascending: true,
    selector:#selector(
        NSString.caseInsensitiveCompare(_:)))
```

Build and Run to see people sorted the way nature intended.

Where to Go From Here

There are a ton of options when it comes to sorting strings. For instance, for you app, even `caseInsensitiveCompare(_:)` might not be specific enough. If you need

to worry about localized strings or numbers being involved in the string you could use `localizedStandardCompare(_:)` or even something very specific via the `compare(options:range:locale:)` method.

To learn more about String sorting, check out the String Programming Guide.