

Errata-Corrige

Nel libro (nei Capitoli 6, 7 e 8; in modo specifico nelle pagine 263, 321, 342, 343-344, 395), per osservare le notifiche utilizzo il metodo

```
addObserverForName:object:queue:usingBlock:
```

Tipicamente poi provo a rimuovere l'observer utilizzando del codice simile a questo:

```
- (void)viewDidUnload
{
    [super viewDidUnload];

    // Rilascia le subview "retained" della main view.
    // ad es. self.myOutlet = nil;

    [[NSNotificationCenter defaultCenter]
        removeObserver:self];
}
```

Questa tecnica, semplicemente non funzionerà. Il view controller (self) non è l'observer. Rimuoverlo non fa affatto nulla.

Invece, ogni volta che si invoca **addObserverForName:** è necessario intercettare il valore di ritorno. Poi si utilizza questo valore quando si vuole rimuovere l'observer. Probabilmente è più semplice farlo utilizzando una proprietà.

```
@property(strong, nonatomic) id observer;
```

Poi creare l'observer come è mostrato qui:

```
self.observer =
    [[NSNotificationCenter defaultCenter]
        addObserverForName:NSUserDefaultsDidChangeNotification
        object:[NSUserDefaults standardUserDefaults]
```

```
queue:nil

usingBlock:^(NSNotification *note) {
    [graphView setWeightEntries:self.weightHistory
               andUnits:getDefaultUnits()];
}];
```

E poi rimuovere l'observer:

```
- (void)viewDidUnload
{
    [super viewDidUnload];
    // Rilascia le subview "retained" della main view.
    // ad es. self.myOutlet = nil;
    [[NSNotificationCenter defaultCenter]
     removeObserver:self.observer];
}
```

Per ulteriori informazioni sulle segnalazioni di correzioni si prega di consultare il sito dell'autore <http://www.freelancemadscience.com> e in particolare la pagina FAQ all'indirizzo <http://www.freelancemadscience.com/creating-ios-apps-faq/>.