

# Kurs-Dokumentation



**Zentrum für Informatik ZFI AG**

**Objective C Grundlagen (IOCG)**

<http://mobile.zfi.ch/IOCG>

Weitere Infos finden Sie unter [mobile.zfi.ch](http://mobile.zfi.ch) oder via Adresse:

Zentrum für Informatik ZFI AG  
Zentralsekretariat  
Rütistrasse 28  
CH-8952 Zürich-Schlieren  
Telefon: 043 433 64 80  
Telefax: 041 530 31 68

Zürich, Basel, Bern, ZÄ¼rich, Schweiz

<b>Titel</b>	<b>Objective C Grundlagen</b>
<b>Untertitel</b>	<b>der Basis-Kurs für künftige Entwickler mobiler Applikationen</b>
<b>Einleitung</b>	Objective-C, auch kurz ObjC genannt, erweitert die Programmiersprache C um Sprachmittel zur objektorientierten Programmierung. Objective-C ist eine strikte Obermenge von C, sprich, jedes C-Programm kann mit einem Objective-C-Compiler kompiliert werden. Objective-C ist die primäre Sprache von Cocoa (Mac OS X) und GNUstep. Die Syntax und Konzeption der objektorientierten Erweiterungen ist an Smalltalk angelehnt und von der gewöhnlichen prozeduralen C-Syntax strikt getrennt. Diese Trennung erlaubt es, das gleiche Konzept zur Erweiterung auf andere imperative Sprachen anzuwenden; so gibt es z. B. ebenfalls Objective Pascal und Objective-J. Objective-C++ erlaubt teilweise die Mischung von Objective-C mit C++-Code mit dem Ziel, älteren Code verwenden zu können. Einer der Design-Gedanken von Objective-C war es, sich der Flexibilität von Smalltalk anzunähern, jedoch auf das zu verzichten, was das Laufzeitverhalten verschlechtern könnte. Der offensichtlichste Verzicht gegenüber Smalltalk ist das Fehlen von Blöcken. Daher ist ein Objective-C-Programm bereits zur Übersetzungszeit vollständig compilierbar. Viele Konzepte sind gar nicht in der Sprachdefinition selbst festgelegt, sondern werden erst durch das Framework, also etwa Cocoa oder GNUStep ermöglicht. Insbesondere ist das gesamte Laufzeitsystem nicht im Compiler implementiert, sondern besteht aus C-Funktionen. Bei Aufruf einer Member-Funktion in C++ etwa fügt grundsätzlich der Compiler den entsprechenden Code ein, während in Objective-C (bei Versenden einer Nachricht an ein Objekt, dazu später mehr) die C-Funktion <code>objc_msg_send()</code> aufgerufen wird. Daher ist eine Darstellung ohne das entsprechende Laufzeitsystem kaum denkbar und nicht sinnvoll. Originäre Objective-C-Schlüsselwörter erkennt man indessen an dem vorangestellten <code>@</code> . Dieser ZFI-Kurs führt in die aktuelle Version 2.0 ein.
<b>Ihr Nutzen</b>	Die Teilnehmenden werden befähigt, eigene kleinere Programme korrekt und effizient in Objective C zu erstellen und zu pflegen.
<b>Voraussetzungen</b>	sehr gute Programmier-Kenntnis einer anderen Sprache.
<b>Teilnehmerkreis</b>	Software-Entwickler mit geringen oder keiner OO-Entwicklungs-Erfahrung, beispielsweise von C oder von ähnlichen nicht-OO-Sprachen herkommend, welche sich mit den Grundlagen der Sprache Objective C vertraut machen möchten.
<b>Unterlagen</b>	ZFI-Kursordner, Fachbuch
<b>Folgekurse</b>	IAEG; iPhone Applikations-Entwicklung Grundlagen
<b>Inhalt</b>	<ul style="list-style-type: none"> <li>- The Objective-C 2.0 Language</li> <li>- Programming in Objective-C</li> <li>- Classes, Objects, and Methods</li> <li>- Data Types and Expressions</li> <li>- Program Looping</li> <li>- Making Decisions</li> <li>- More on Classes</li> <li>- Inheritance</li> <li>- Polymorphism, Dynamic Typing, and Dynamic Binding</li> <li>- More on Variables and Data Types</li> <li>- Categories and Protocols</li> </ul>

- The Preprocessor
- Underlying C Language Features
  
- The Foundation Framework
- Introduction to the Foundation Framework
- Numbers, Strings, and Collections
- Working with Files
- Memory Management
- Copying Objects
- Archiving
  
- Cocoa and the iPhone SDK
- Introduction to Cocoa
- Writing iPhone Applications

**Beitrag**

**Der Teilnehmerbeitrag versteht sich rein netto. Das ZFI ist (gemäss MwSt-Gesetz) nicht Mehrwertsteuerpflichtig und erhebt somit keine MwSt. Bei länger als einen Monat dauernden Lehrgängen ist die Zahlung des Teilnehmerbeitrages in mehreren Raten möglich (pro rata temporis).**