

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.

Ihr Nutzen

Nach dem Seminar können Sie Anwendungen in der Programmiersprache Objecting C erstellen und verändern. Dies ist die Voraussetzung für erfolgreiche Anwendungsentwicklung unter Cocoa Betriebssystemen (MacOS und iPhone).

Voraussetzungen

Allgemeine Programmierkenntnisse

Preis pro Teilnehmer

EUR 1440,- exklusive der gesetzlichen MwSt.

Hinweise

-

Seminardauer

3 Tag(e)/Day(s)

Version: 2.0

Seminarinhalte

1. Tag

- * Variablen, Operatoren und Kontrollstrukturen
- C im Terminal, C mit Xcode
- Variablen und Operatoren
- Weitere Operatoren
- Kontrollstrukturen und Vergleichsoperatoren
- if/else-Strukturen, switch-Statement
- Schleifen
- Schleifen mit continue und break steuern und logische
- Weitere Variablentypen, Format specifier modifier,
- Speicherbedarf von Variablen
- * Funktionen, Zeiger und Arrays
- Funktionsdeklarationen und -definitionen
- Funktion mit Parametern und Rückgabewert, Rekursion
- Gültigkeitsbereich von Variablen
- "Call by Value" und "Call by Reference"
- Zeigergrößen und adressierbarer Speicher
- "Pointer-to-Pointer"
- Arrays und die Compiler-Direktive #define
- Eigenheiten von Arrays und Zeigern, Makros mit Parametern
- Zeichenketten (Textstrings), char-Arrays und char-Pointer
- Arrays und Pointer
- * Structs, Enumerations und Typedef
- Structs, Linked-Lists, malloc() und free(), Enumerations
- C-Grundlagen und das typedef-Statement

2. Tag

- * Objective-C-Grundlagen
- Einführung in Objective-C
- Überblick über das Foundation-Framework
- * Einstieg in die objektorientierte Programmierung
- Objektorientierte Programmierung und Klassen-Konzept
- Objekte erzeugen und Setter-Methoden
- Datenkapselung und Accessor-Methoden
- Direktzugriff auf iVars und Zugriffsprivilegien
- Vererbung und Klassenhierarchie, Methoden überschreiben
- Klassenmethoden, Klassen erweitern
- Quellcode auf Dateien verteilen
- Arrays und Typisierung
- Compiler-Warnungen als Fehler behandeln lassen
- NSArray und NSMutableArray
- Composition
- Anpassungen an der Order-Klasse, Nachrichten an nil senden
- Initialisierungs-meth.: Convenience- und Designated Initializer

* Objective-C-Internals

- NSObject, isa-Pointer, Klassen-Objekte und Meta-Klassen
- Selectors und Methoden
- Einblick in die Objective-C Runtime-Bibliothek
- 3. Tag
- * Memory-Management
- Garbage Collection
- Manuelles Memory-Management
- NSAutoreleasePool
- Beispiele für manuelles Memory-Management
- iPhone und AutoreleasePools
- * Objective-C 2.0: Properties
- Properties, Properties und @synthesize
- Accessor-Methoden überschreiben
- * Protocols & Categories
- Kategorien, Sourcecode-Splitting und informelle Protokolle
- * Collections & Fast Enumeration
- Fast Enumeration
- NSArray, NSNumber und NSNull
- NSDictionary und NSMutableDictionary

