



## Entwurfsmuster mit Java oder C++ oder C#-Entwickler (S721)

### Überblick

#### Beschreibung

"Eigentlich" ist es ganz einfach - wie immer: Es gibt typische Aufgaben und dazu passende, typische Lösungen. Dennoch zählt das Thema Design Pattern unter Softwareentwicklern zu den anspruchsvolleren Themen. Das Wissen über verfügbare Pattern allein reicht nicht, es ist auch notwendig, ein Gefühl für Pattern zu entwickeln. Es gilt, den Blick für Design Pattern zu schärfen und zwar sowohl in der Modellierungsphase, als auch in der nachfolgenden Implementierungsphase. Der Kurs wird mit Eclipse durchgeführt, Teilnehmer können aber auch andere Entwicklungsumgebungen verwenden. Ebenfalls ist der Kurs für C++- und C#-Entwickler geeignet.

Termin	freie Plätze	Tage	Preis in Euro zzgl. MwSt.
30.07.-01.08.2012	>3	3	1.940,00 <span style="border: 1px solid black; padding: 2px;">3=2</span>
14.01.-16.01.2013	>3	3	1.940,00 * <span style="border: 1px solid black; padding: 2px;">3=2</span>

Bitte beachten Sie die aktuellen Termine und Preise auf unserer Webseite.

\* = Frühbucherrabatt

### Details

#### Seminar-Ziel

Es werden viele der Patterns aus dem Buch-Klassiker "Design Patterns" der GoF ("Gang of Four") erarbeitet. Diese Kenntnisse heben die Qualität der Implementierung, helfen aber auch bei der Erstellung von UML-Klassendiagrammen in der Designphase. Bei einem hohen Vorwissen der Teilnehmer wird die Agenda um weitere Pattern ergänzt.

#### Wer sollte teilnehmen

Softwareentwickler mit Java-Kenntnissen, die sich verbessern möchten.

#### Inhalt

- **Einführung in Design Pattern**
- **Pattern in der "Realen Welt"**
- **Beispiele und Übungen für Erzeugungsmuster**
  - "Singleton" - von manchen Objekten darf es nur eine Instanz geben
  - "Factory" - Erzeugung ähnlicher Objekte
  - "Prototyp" - komplexe Objekte anhand einer Vorlage erzeugen
- **Beispiele und Übungen für Strukturmuster**
  - "Facade" - einfache Schnittstellen für komplexe Systeme schaffen
  - "Observer" - Kommunikationsstruktur für unabhängige Komponenten
  - "Composite" - die Implementierung für "Teile-Ganze-Beziehungen"
  - "Decorator" - eine flexible Form, um eine Klasse zu erweitern
  - "Proxy" - ein vorgelagertes Stellvertreterobjekt
  - "Adapter" - inkompatible Objekte über Schnittstellen koppeln
- **Beispiele und Übungen für Verhaltensmuster**
  - "Command" - Trennung von Befehlen und Ausführung
  - "Iterator" - Elemente einer Liste durcharbeiten
  - "Visitor" - Auslagerung von Operationen in externe Klassen mit Interface-Kopplung
- **Mikropattern (bei Bedarf)**
  - Wie wird eine equals()-Methode sauber implementiert
  - Verschiedene Möglichkeiten, um clone() zu implementieren
  - Die Rolle des Konstruktors
- **Bugpatterns**
  - Typische Fehler und deren typische Implementierungen
  - Überblick über die Werkzeuge findbugs und PMD

### Buchungsalternativen

#### Firmenschulung

Inhouse-Schulung mit individuellen Inhalten zum Wunschtermin. Preis auf Anfrage.

#### Individualschulung

Schulung für eine Einzelperson mit individuellen Inhalten zum Wunschtermin. Preis auf Anfrage.

### Enthaltene Leistungen

Schulungsunterlagen  
Teilnahmezertifikat  
Frühstück und Mittagessen  
Getränke, Obst und Snacks

### Organisation

#### Teilnehmerzahl

min. 1, max. 8 Personen \*\*\* Der Kurs findet garantiert statt! \*\*\*

#### Seminarzeiten

1. Tag 10:00-17:00h, Folgetag(e)  
09:00-16:00h

#### Ort der Schulung

GFU-Schulungszentrum Köln oder bei Ihnen als Inhouse-Schulung

### Haben Sie Fragen?

Gerne beraten wir Sie persönlich per Mail oder Telefon.

- info@gfu.net
- Infoline 0221 82 80 90