

# Kurs-Dokumentation



**Zentrum für Informatik ZFI AG**

**C++ Design Pattern Programmierung (CPDP-0517)**

**-IT Ausbildung nach Mass**

<http://www.zfi.ch/CPDP-0517>

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

**Zentrum für Informatik ZFI AG  
Zentralsekretariat  
Rütistrasse 28  
CH-8952 Zürich-Schlieren  
Telefon: 044 732 40 00  
Telefax: 041 530 31 68**

**Zürich, Basel, Bern, Zürich, Schweiz**

|                        |   |
|------------------------|---|
| <b>Titel</b>           | <b>C++ Design Pattern Programmierung</b>  |
| <b>Untertitel</b>      | <b>Eine praxisbezogene Einführung in die Design Pattern Programmierung mit C++</b>  |
| <b>Einleitung</b>      | <p>Mit dem Begriff "Design Patterns" werden bestehende Erfahrungsmuster des Software-Entwurfs gesammelt und als Katalog und Wissensbasis in der Praxis vielfach wiederverwendet. Das Design objektorientierter Software stellt hohe Anforderungen an die beteiligten Entwickler und Programmierer. Heute kann der Softwareentwickler jedoch auf katalogisierte Designmuster zurückgreifen, die ihn hilfreich unterstützen. Das Verstehen und Einüben solcher Designmuster ist ein Schlüssel bei der Programmierung mit frameworkartigen Entwicklungsumgebungen, insbesondere auch mit C++.</p> <p>Dieser Kurs vermittelt den praxisnahen Umgang mit Design Patterns in C++. Die effektive Programmierung der wichtigsten Patterns bildet dabei den Schwerpunkt, wobei nach dem Motto gearbeitet wird: "Patterns lernen, die in der Praxis anwendbar sind"!</p> <p>Alle Übungen basieren auf einer bestehenden Lösung, die über ein Design Pattern zu</p> <p>Das Thema Design Patterns ist ein Klassiker und ein Muss für jeden C# Developer. verbessern ist (via Refactoring).</p> <p>Das Thema Design Patterns ist ein Klassiker und ein Muss für jeden C++ Developer.</p> |
| <b>Ihr Nutzen</b>      | <ul style="list-style-type: none"> <li>- Die Ideen und Ziele der Design Patterns verstehen</li> <li>- Die wichtigsten Design Patterns anhand von einfachen Beispielen anwenden</li> <li>- Die Design Patterns katalogisieren und strukturieren können</li> <li>- Ausgewählte Design Patterns selber in C++ programmieren</li> </ul>   |
| <b>Voraussetzungen</b> | Programmiererfahrung mit C++ analog dem ZFI Kurs CBAS ("C++ Basics") inkl. C++-Praxis.  |
| <b>Teilnehmerkreis</b> | C++ Programmierer   |
| <b>Unterlagen</b>      | <ul style="list-style-type: none"> <li>- Begleitbuch</li> <li>- Tutorials</li> <li>- Intranet Site</li> </ul>   |
| <b>Folgekurse</b>      | - C++ Refactoring, Kurscode: CREF   |
| <b>Inhalt</b>          | <ul style="list-style-type: none"> <li>- Einführung</li> <li>- Was sind Design Patterns</li> <li>- Pattern Dokumentation mit UML</li> <li>- Pattern Katalog</li> <li>- Übersicht</li> <li>- Patterns und C++</li> </ul><br><ul style="list-style-type: none"> <li>- Creational Patterns</li> <li>- Factory</li> <li>- Singleton</li> <li>- Prototype</li> <li>- Builder</li> </ul><br><ul style="list-style-type: none"> <li>- Structural Patterns</li> </ul>   |

- Adapter
- Bridge
- Composite
- Decorator
  
- Behavioral Patterns
- Command
- Iterator
- Observer
- Strategy
  
- Diverses
- Weiterführende Patterns
- Patterns und Refactoring

**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).**