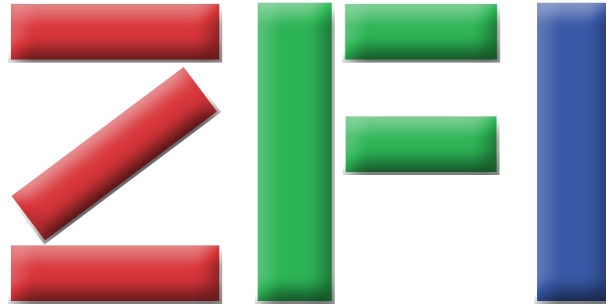


# Fernkurs-Dokumentation



## Zentrum für Informatik ZFI AG

### Fernkurs: Oracle Certified Associate, Java SE

### 7 Programmierer I, 1Z0-803 (FOCA)

<http://www.zfi.ch/FOCA>

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>Fernkurs: Oracle Certified Associate, Java SE 7 Programmier I, 1Z0-803</b>
<b>Untertitel</b>	<b>Der Vorbereitungslehrgang auf die offizielle Oracle Certified Associate, Java SE 7 Programmier 1Z0-803 Prüfung als Fernkurs.</b>
<b>Einleitung</b>	<p>Die Oracle Certified Associate, Java SE 7 Programmier I Zertifizierung richtet sich an Software-Entwickler, welche ihre Kenntnisse der Java Programmiersprache attestiert haben möchten. Das Bestehen der Prüfung Java Programmier I (1Z0-803) bestätigt, dass der Programmierer den Syntax und die fundamentale Struktur der Java Programmiersprache versteht.</p> <p>Die OCA Prüfung 1Z0-803 wird für die Folgeprüfung 1Z0-804, Oracle Certified Professional, Java Programmier II, vorausgesetzt.</p> <p>Mit der Variante Fernkurs erarbeiten Sie sich die theoretischen Grundlagen selbständig und festigen Ihr Wissen anhand von vielen Übungen. Die wichtigsten Fakten werden in einem Factsheet zusammengefasst mitgeliefert und sind ein wichtiger Teil der Prüfungsvorbereitung.</p> <p>Eine begleitende grössere Programmierübung rundet den Fernkurs ab. Bei Fragen können Sie den ZFI Kursleiter per EMail kontaktieren und Ihre Fragen werden umgehend und kompetent beantwortet.</p> <p>Als Entwicklungsumgebung setzen wir die Eclipse Java IDE ein. Sie können aber auch die Oracle Netbeans IDE oder IntelliJ verwenden.</p> <p>Nach der Anmeldung für diesen Fernkurs erhalten Sie von uns das Begleitbuch per Post zugestellt. Zugleich werden Sie von uns per EMail kontaktiert und Sie erhalten von uns den Link auf die elektronischen Unterlagen (Dokument, Übungen und Beispiele) für den Download. In einer ersten Phase arbeiten Sie die Unterlagen inkl. den Übungen durch. Die Übungen enthalten Musterlösungen. Eine Übungsreihe gilt als Repetition, die Lösung senden Sie per EMail an uns zur Kontrolle. Sie können uns jederzeit über den vereinbarten Mail-Account kontaktieren, wir helfen Ihnen gerne weiter. In einer 2. Phase erfolgt das Training auf die Prüfung selbst indem Sie viele echte Prüfungsfragen beantworten. Zu diesen Fragen gibt es wiederum Lösungen. Planen Sie für diese Vorbereitung mehrere Wochen ein, dann sind Sie für die Prüfung bereit und optimal vorbereitet.</p> <p>Die aktuellen Java SE 7/8 Programmier Zertifizierungen von Oracle:</p> <ul style="list-style-type: none"><li>Java SE 7 Programmier I (1Z0-803)</li><li>Java SE 7 Programmier II (1Z0-804)</li><li>Upgrade to Java SE 7 Programmier (1Z0-805)</li><li>Java SE 8 Programmier I (1Z0-808)</li><li>Java SE 8 Programmier II (1Z0-809)</li></ul>
<b>Ihr Nutzen</b>	<b>Dieser ZFI-Lehrgang richtet sich an Personen, welche die Sprache Java systematisch lernen möchten, um anschliessend Java-Applikationen in ihrem beruflichen Umfeld zu entwickeln und zu warten. Lernziel ist selbstverständlich das Bestehen der von Oracle durchgeführten Prüfung zum Oracle Certified Associate (OCA) Java Programmier I (1Z0_803).</b>
<b>Voraussetzungen</b>	

<b>Teilnehmerkreis</b>	<b>Dieser ZFI-Lehrgang richtet sich an Programmierer, welche bereits erste Schritte in Java gemacht haben oder bereits in einer anderen Programmiersprache entwickelt haben.</b>
<b>Unterlagen</b>	<ul style="list-style-type: none"><li>- Begleitbuch</li><li>- Tutorials</li><li>- Intranet Site</li></ul>
<b>Folgekurse</b>	
<b>Inhalt</b>	<b>Oracle Certified Associate Java SE 7 Programmer I</b> <ul style="list-style-type: none"><li>- Java Basics<ul style="list-style-type: none"><li>- Define the scope of variables</li><li>- Define the structure of a Java class</li><li>- Create executable Java applications with a main method</li><li>- Import other Java packages to make them accessible in your code</li></ul></li><li>- Working With Java Data Types<ul style="list-style-type: none"><li>- Declare and initialize variables</li><li>- Differentiate between object reference variables and primitive variables</li><li>- Read or write to object fields</li><li>- Explain an object's lifecycle</li><li>- Manipulate data using the <code>StringBuilder</code> class and its methods</li><li>- Create and manipulate strings</li></ul></li><li>- Using Operators and Decision Constructs<ul style="list-style-type: none"><li>- Use Java operators</li><li>- Use parentheses to override operator precedence</li></ul></li></ul>

- Test equality between strings and other objects using == and equals ()
- Create if and if/else constructs
- Use a switch statement
  
- Creating and Using Arrays
- Declare, instantiate, initialize and use a one-dimensional array
- Declare, instantiate, initialize and use multi-dimensional array
- Declare and use an ArrayList
  
- Using Loop Constructs
- Create and use while loops
- Create and use for loops including the enhanced for loop
- Create and use do/while loops
- Compare loop constructs
- Use break and continue
  
- Working with Methods and Encapsulation
- Create methods with arguments and return values
- Apply the static keyword to methods and fields
- Create an overloaded method
- Differentiate between default and user-defined constructors

- Create and overload constructors
- Apply access modifiers
- Apply encapsulation principles to a class
- Determine the effect upon object references and primitive values when they are passed into methods that change the values
  
- Working with Inheritance
- Implement inheritance
- Develop code that demonstrates the use of polymorphism
- Differentiate between the type of a reference and the type of an object
- Determine when casting is necessary
- Use super and this to access objects and constructors
- Use abstract classes and interfaces
  
- Handling Exceptions
- Differentiate among checked exceptions, RuntimeExceptions and Errors
- Create a try-catch block and determine how exceptions alter normal program flow
- Describe what exceptions are used for in Java
- Invoke a method that throws an exception
- Recognize common exception classes and categories

**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).