



Reinhard Schiedermeier

Programmieren mit Java II

Programmieren mit Java II

Programmieren mit Java II

Inhaltsverzeichnis

Programmieren mit Java II

Inhaltsverzeichnis

Vorwort

Kapitel 1 Ein- und Ausgabe (I/O)

1.1 Standardein- und -ausgabe

1.1.1 Vordefinierte I/O-Objekte

1.1.2 Redirection

1.1.3 Standard-Fehlerausgabe

1.2 Byteströme

1.2.1 Abstrakte Basisklassen

1.2.2 Lesen und Schreiben

1.2.3 Pufferung

1.2.4 Schließen von Streams

1.2.5 ARM

1.3 File-I/O

1.3.1 Konkrete Datenquellen und -senken

1.3.2 Filestreams

1.3.3 Blockweise Übertragung

1.3.4 Abstrakte und konkrete Methoden

1.4 Transformationen mit Filterklassen

1.4.1 Ableitung und Komposition

1.4.2 Abstrakte Filterklassen

1.4.3 Konkrete Filterklassen

1.4.4 Filter der Laufzeitbibliothek

1.4.5 Zip-Dateien

Inhaltsverzeichnis

1.5 Decorator-Pattern

1.6 Umgang mit Textdateien

 1.6.1 Interne und externe Darstellung

 1.6.2 Basisklassen Reader und Writer

 1.6.3 Konkrete Quellen und Senken

 1.6.4 Textfilter

 1.6.5 Ersetzen der Standardein- und -ausgabe-Objekte

 1.6.6 Zeichensätze und Encodings

1.7 Definition neuer I/O-Klassen

1.8 Datei-Operationen und Directories

 1.8.1 Datei-Objekte

 1.8.2 Eigenschaften von Dateien

 1.8.3 Zugriffsrechte

 1.8.4 Pfade

 1.8.5 Directories

 1.8.6 Schreiben und Lesen

 1.8.7 Filesystem-Operationen

Kapitel 2 Serialisierung

2.1 Fragen

2.2 Object-Streams

 2.2.1 Methoden readObject und writeObject

 2.2.2 Interface Serializable

 2.2.3 Objektgraphen

 2.2.4 Stream-Struktur

 2.2.5 Aufruf von Konstruktoren

 2.2.6 Cloning

 2.2.7 Caching

 2.2.8 Nicht serialisierte Variablen

 2.2.9 Serialisierungsversionen

 2.2.10 Eingriff in die Serialisierung



Inhaltsverzeichnis

2.3 JavaBeans

- 2.3.1 Properties
- 2.3.2 XMLEncoder und XMLDecoder
- 2.3.3 Speicherformat XML
- 2.3.4 Inkonsistenzen
- 2.3.5 Objektgraphen
- 2.3.6 Serialisierer-Objekte
- 2.3.7 Neue Serialisierer
- 2.3.8 Deserialisierung

2.4 XStream

- 2.4.1 toXML und fromXML
- 2.4.2 Referenzen
- 2.4.3 Converter
- 2.4.4 Neue Converter, statische und transiente Variablen
- 2.4.5 Encoding
- 2.4.6 Kompatibilität mit Object-Streams

Kapitel 3 XML

3.1 Struktur, Grammatik und Validierung

- 3.1.1 Aufbau eines XML-Dokuments
- 3.1.2 Validierung

3.2 Arbeit mit dem DOM

- 3.2.1 DOM-Parser
- 3.2.2 Validierung
- 3.2.3 Objektbaum
- 3.2.4 Arbeiten mit dem DOM
- 3.2.5 Ausgabe
- 3.2.6 Modifikation
- 3.2.7 Fehlerbehandlung

3.3 SAX-Parser

- 3.3.1 Initialisierung



Inhaltsverzeichnis

- 3.3.2 Handler
- 3.3.3 Validierung
- 3.3.4 Umgang mit Text
- 3.3.5 Fehlerbehandlung

Kapitel 4 Rekursion

- 4.1 Arbeitsweise
 - 4.1.1 Selbstauftrag und Abbruchkriterium
 - 4.1.2 Parameter und Ergebnis
 - 4.1.3 Indirekte Rekursion
 - 4.1.4 Laufzeitstack
 - 4.1.5 Rekursive Datenstrukturen
- 4.2 Rekursion und Iteration
 - 4.2.1 Endrekursion
 - 4.2.2 Eliminierung der Endrekursion
- 4.3 Klassifizierung
 - 4.3.1 Lineare Rekursion
 - 4.3.2 Verzweigte Rekursion
 - 4.3.3 Geschachtelte Rekursion
- 4.4 Anwendungen
 - 4.4.1 Filesystem
 - 4.4.2 Parser
 - 4.4.3 Permutationen
- 4.5 Memoizing
 - 4.5.1 Redundante Methodenaufrufe
 - 4.5.2 Bedingungen für Memoizing
 - 4.5.3 Implementierung von Caches

Kapitel 5 Geschachtelte Klassen

- 5.1 Statisch geschachtelte Klassen
 - 5.1.1 Definition und Syntax
 - 5.1.2 Bedeutung von private



Inhaltsverzeichnis

5.1.3 Anwendungsbeispiel: Factory-Methoden

5.1.4 Bytecode-Dateien

5.1.5 Statisch geschachtelte Interfaces

5.2 Innere Klassen

5.2.1 Anwendungsbeispiel: Iterierbare Strings

5.3 Lokale Klassen

5.4 Anonyme Klassen

5.4.1 Anwendungsbeispiele

5.5 Lambda-Ausdrücke (Java 8)

5.5.1 Funktionsinterfaces

5.5.2 Lambda-Ausdrücke

5.5.3 Lambda-Ausdrücke im Typsystem

5.5.4 Closure

5.5.5 Gültigkeitsbereiche

5.5.6 Exceptions in einem Lambda-Ausdruck

5.5.7 Geschachtelte Lambda-Ausdrücke

5.5.8 Funktionsinterfaces in der Bibliothek

5.6 Default-Methoden (Java 8)

5.6.1 Definition von Default-Methoden

5.6.2 Implementierung von Default-Methoden

5.6.3 Einschränkungen von Default-Methoden

5.6.4 Vererbung und Default-Methoden

5.6.5 Evolution von Interfaces

5.6.6 ABCs und Default-Methoden

Kapitel 6 Nebenläufigkeit

6.1 Paralleler Programmablauf

6.1.1 Basisklasse Thread

6.1.2 Start eines neuen Threads

6.1.3 Eigenschaften von Threads

6.1.4 Ende eines Threads



Inhaltsverzeichnis

- 6.1.5 Interface Runnable
- 6.1.6 Threads am Programmende
- 6.2 Kommunikation mit Interrupts**
 - 6.2.1 sleep und InterruptedException
- 6.3 Scheduling und Multiprozessoren**
 - 6.3.1 Lebenslauf eines Threads
 - 6.3.2 Scheduler
 - 6.3.3 Expliziter Zustandswechsel
 - 6.3.4 Threads und Prozessoren
 - 6.3.5 Hintergrund-Threads
 - 6.3.6 Threads und Prozesse
- 6.4 Konkurrierender Zugriff und Synchronisation**
 - 6.4.1 Thread-lokale und geteilte Daten
 - 6.4.2 Gleichzeitiger Zugriff
 - 6.4.3 Zusammengesetzte Operationen
 - 6.4.4 Synchronisation auf einem Monoprozessor
 - 6.4.5 Atomare Operationen
 - 6.4.6 Länge von synchronized-Blöcken
 - 6.4.7 synchronized-Methoden
- 6.5 volatile und Deadlocks**
 - 6.5.1 Anwendungsbeispiel Parkhaus
 - 6.5.2 Optimierung von Variablenzugriffen
 - 6.5.3 Modifier volatile
 - 6.5.4 Deadlocks
 - 6.5.5 Unveränderliche Klassen
- 6.6 Bedingtes Warten**
 - 6.6.1 Aktives Warten
 - 6.6.2 wait und notify
 - 6.6.3 Test der wait-Bedingung
 - 6.6.4 notify und notifyAll

Inhaltsverzeichnis

Kapitel 7 Netzwerkprogrammierung

7.1 Grundlagen

- 7.1.1 ISO/OSI-Schichtenmodell
- 7.1.2 Protokollstapel
- 7.1.3 MAC-Adressen (Ethernet)
- 7.1.4 Hostadressen (IP)
- 7.1.5 Loopback-Device
- 7.1.6 Symbolische Hostnamen
- 7.1.7 Domainname-Server (DNS)
- 7.1.8 Ports
- 7.1.9 Dienste und Protokolle
- 7.1.10 Client-Server-Systeme
- 7.1.11 Peer-to-Peer-Netzwerke

7.2 Clients und Server

- 7.2.1 Sockets
- 7.2.2 Portscanner
- 7.2.3 Lesen und Schreiben
- 7.2.4 Beenden einer Verbindung
- 7.2.5 Pufferung
- 7.2.6 Beispiel: Echo-Client
- 7.2.7 Telnet-Client
- 7.2.8 Serversockets
- 7.2.9 Beispiele: Time- und Echo-Server
- 7.2.10 Verbindung auf einzelnen Netzwerkschnittstellen
- 7.2.11 Teilweiser Abbau
- 7.2.12 Verschlüsselte Verbindung

7.3 HTML, HTTP und Webserver

- 7.3.1 Idee
- 7.3.2 Webserver und Browser
- 7.3.3 URLs

Inhaltsverzeichnis

- 7.3.4 HTML
- 7.3.5 DOM
- 7.3.6 Aufbau eines Dokuments
- 7.3.7 HTTP-Request
- 7.3.8 HTTP-Response
- 7.3.9 HTTP-Zugriff mit Bibliotheksklassen
- 7.3.10 Minimaler Webserver
- 7.3.11 Statischer Webserver
- 7.3.12 Dynamische Webseiten
- 7.3.13 Nebenläufiger Webserver
- 7.3.14 Gefährlicher dynamischer Server
- 7.3.15 Webserver-Basisklasse

Kapitel 8 Reflection

8.1 Factory-Methoden

- 8.1.1 Typobjekte und die Methode `forName`
- 8.1.2 Erzeugen von Objekten mit `newInstance`
- 8.1.3 Beispielanwendung: Pizzas
- 8.1.4 Konstruktor-Objekte
- 8.1.5 Beispielanwendung: Pizzas, weitere Verbesserung

8.2 Analyse der Codestruktur

- 8.2.1 Modifier
- 8.2.2 Konstruktoren
- 8.2.3 Variablen
- 8.2.4 Methoden
- 8.2.5 Anwendung: UML-artige Skizze einer Klassendefinition

8.3 Analyse und Modifikation von Objekten

- 8.3.1 Zugriff auf Variablen
- 8.3.2 Umgang mit Arrays
- 8.3.3 Verändern von Objekten
- 8.3.4 Anwendung: Reflektive `toString`-Methode

Inhaltsverzeichnis

8.3.5 Reflektive Methodenaufrufe

8.3.6 Erzeugen neuer Objekte

Kapitel 9 Annotationen

9.1 Idee

9.2 Vordefinierte Annotationen

9.2.1 Warnungen des Java-Compilers

9.2.2 Annotation SafeVarargs

9.3 Neue Annotationen

9.3.1 Definition

9.3.2 Annotation-Werte

9.3.3 Meta-Annotation Target

9.3.4 Meta-Annotation Retention

9.3.5 Meta-Annotation Inherited und Documented

9.3.6 Marker-Annotationen

9.4 Auswertung zur Laufzeit

9.4.1 Anwendung: Reflektive `toString`-Methode

9.5 Prozessoren

9.5.1 Arbeitsweise

9.5.2 Diagnoseausgabe des Compilers

9.5.3 Anwendung: Generierte Bean-Klassen

9.6 Packages

Anhang A Voraussetzungen

Anhang B I/O-Pipelines

Anhang C Beispiel Decorator-Pattern

Anhang D Java-8-Entwicklerversion

D.1 Lambda-Ausdrücke

D.2 Default-Methoden

Literaturverzeichnis

Inhaltsverzeichnis

Index

Ins Internet: Weitere Infos zum Buch, Downloads, etc.

Copyright

Copyright

Daten, Texte, Design und Grafiken dieses eBooks, sowie die eventuell angebotenen eBook-Zusatzdaten sind urheberrechtlich geschützt. Dieses eBook stellen wir lediglich als **persönliche Einzelplatz-Lizenz** zur Verfügung!

Jede andere Verwendung dieses eBooks oder zugehöriger Materialien und Informationen, einschließlich

- der Reproduktion,
- der Weitergabe,
- des Weitervertriebs,
- der Platzierung im Internet, in Intranets, in Extranets,
- der Veränderung,
- des Weiterverkaufs und
- der Veröffentlichung

bedarf der **schriftlichen Genehmigung** des Verlags. Insbesondere ist die Entfernung oder Änderung des vom Verlag vergebenen Passwortschutzes ausdrücklich untersagt!

Bei Fragen zu diesem Thema wenden Sie sich bitte an: info@pearson.de

Zusatzdaten

Möglicherweise liegt dem gedruckten Buch eine CD-ROM mit Zusatzdaten bei. Die Zurverfügungstellung dieser Daten auf unseren Websites ist eine freiwillige Leistung des Verlags. **Der Rechtsweg ist ausgeschlossen.**

Hinweis

Dieses und viele weitere eBooks können Sie rund um die Uhr und legal auf unserer Website herunterladen:

<http://ebooks.pearson.de>