

Einstieg in Java und OOP

Christian Silberbauer

Kapitel 1: Einführung

Gliederung

1. Einführung
2. Grundelemente der Programmierung
3. Objektorientierung
4. GUI-Programmierung
5. Objekte verwalten
6. Persistenz
7. Threads
8. Reflection

Methodik der Vorlesung

- Wenige themenübergreifende Anwendungen, im Wesentlichen:
 - Zeichenprogramm (Kapitel 3 und 5)
 - Kundenverwaltung (Kapitel 4, 6 und 8)
 - Chat (Kapitel 7)
- Übergreifende Übungsaufgaben:
 - Z.B. Lohn- und Gehaltsabrechnung
- Die Übungsaufgaben sind wesentlich für den Stoff und ergänzen die Vorlesung.
- Vorteile:
 - Wenige fachliche Kontexte reduzieren den Aufwand, sich mit „Nebensächlichkeiten“ zu beschäftigen.
 - Erlaubt die einfachere Bildung von Assoziation zwischen den Themen.
- Nachteil:
 - Ein „am Ball bleiben“ das ganze Semester hinweg ist notwendig.

Methodik der Vorlesung

- Folien dienen in der Vorlesung und als Skript
- Nachteil:
 - Für die Präsentation teilweise etwas überladene Folien
- Vorteile:
 - Zum Nachlesen gut geeignet
 - Der Bezug zur Vorlesung kann gut hergestellt werden.

Literatur

- **Einstieg in Java und OOP**, Christian Silberbauer, Springer-Verlag, 2020
- **Sprechen Sie Java?**, Hanspeter Mössenböck, dpunkt.verlag, 2011
- **Programmieren in Java**, Fritz Jobst, Hanser-Verlag, 2014
- **Design Patterns**, Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, Addison-Wesley, 1994
- **Entwurfsmuster von Kopf bis Fuß**, Eric Freeman, Elisabeth Freeman, Kathy Sierra, Bert Bates, O'Reilly, 2015

Geschichte von Java

- ❑ Java wurde bei Sun Microsystems ab 1991 unter der Federführung von James Gosling entwickelt.
- ❑ Zunächst unter dem Namen OAK (Object Application Kernel).
- ❑ OAK wurde schließlich in Java umbenannt.
- ❑ Der offizielle Start für Java war die Veröffentlichung des Java Development Kit (JDK) 1.0 im Jahre 1996.

Geschichte von Java (2)

- ❑ 2004 erschien Java 5 wodurch an der Sprache umfangreiche Änderungen durchgeführt wurden.
- ❑ 2010 wurde Sun Microsystems von Oracle übernommen.
- ❑ Mit der Version 8 wurden wieder größere Änderungen vorgenommen.
- ❑ Seit September 2017 bzw. Java 9 wird ein neues Java-Release alle 6 Monate veröffentlicht.
- ❑ Aktuelle Version ist Java 13.

Sprachmerkmale

- Starke und statische Typsicherheit
- Plattformunabhängigkeit
- Objektorientierte Programmierung

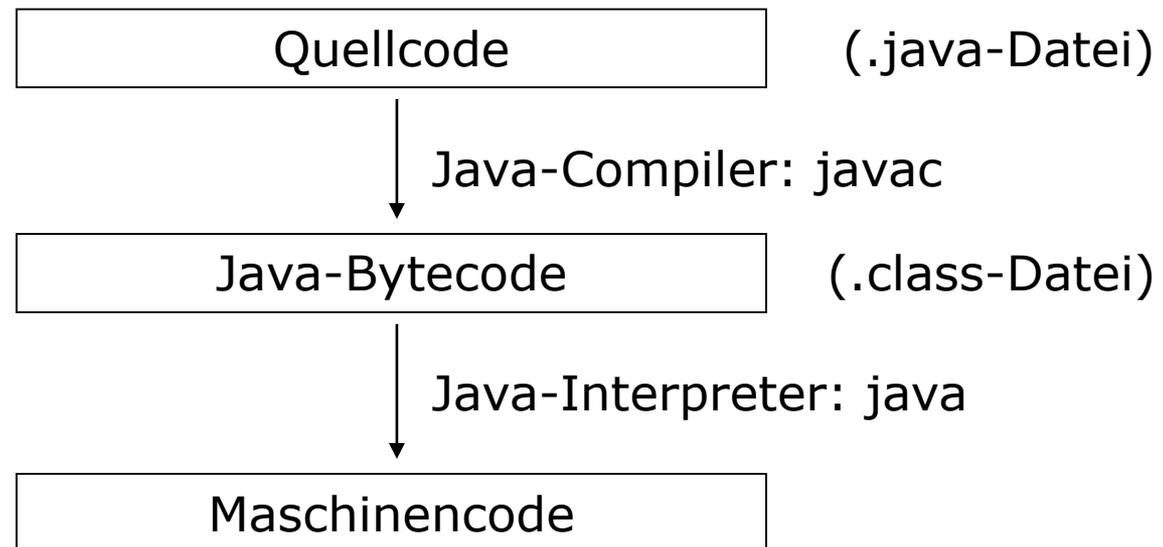
Typen in Java

- Aus der Java-Spezifikation, Chapter 4:
(siehe <http://docs.oracle.com/javase/specs/jls/se8/html/jls-4.html>)

“The Java programming language is a *statically typed* language, which means that every variable and every expression has a type that is known at compile time.

The Java programming language is also a *strongly typed* language, because types limit the values that a variable (§4.12) can hold or that an expression can produce, limit the operations supported on those values, and determine the meaning of the operations. Strong static typing helps detect errors at compile time.”

Übersetzen eines Java-Programms



Paradigma der Objektorientierung

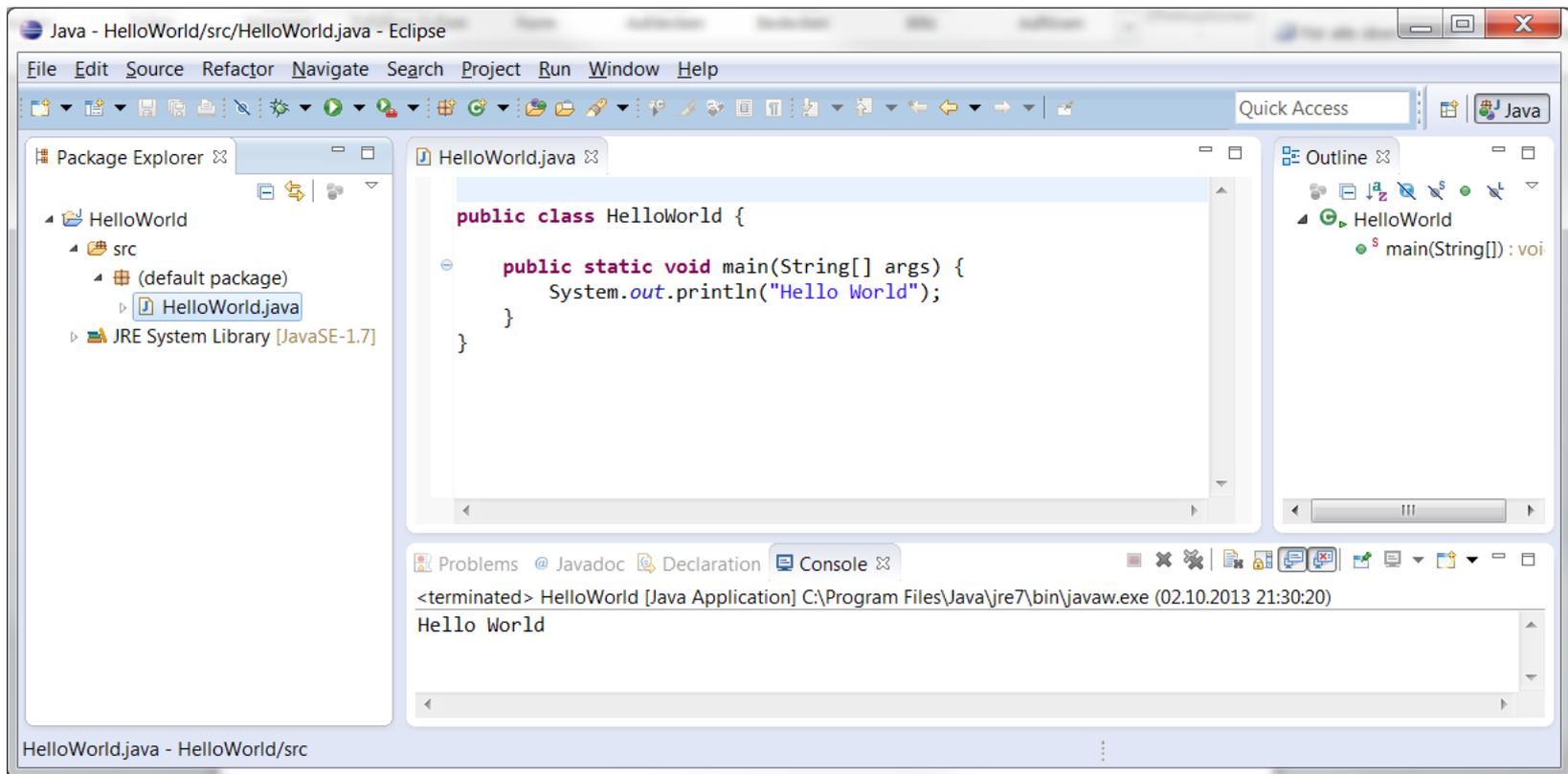
- Bisherige (prozedurale) Betrachtung:
 - Funktionen rufen sich gegenseitig auf
 - Daten werden per Funktionsparameter ausgetauscht
 - Gemeinsame Daten werden „global“ definiert

- Objektorientierte Betrachtung:
 - Gemeinsame Daten und Funktionen werden in einer Klasse zusammengefasst.
 - Klassen implementieren einen Typ, deren Exemplare bzw. Instanzen werden als Objekte bezeichnet.

„Hello World“

```
public class HelloWorld {  
  
    public static void main(String[] args) {  
  
        // Gibt "Hello World" aus  
        System.out.println("Hello World!");  
    }  
}
```

Die Entwicklungsumgebung Eclipse



Wiederholung mit Java

- Umgang mit Variablen (Elementare Datentypen)
- Logische Ausdrücke
- Verzweigungen (if-else, switch)
- Schleifen (for, while, do-while)
- Funktionen
- Rekursion
- Arrays
- (Exceptions)

