
Allgemeines zu:

I) Grundlagen der Programmierung

VV-Nr.: 10890-01

II) Erweiterte Grundlagen der Programmierung

VV-Nr.: 45398-01

Webseiten zur Vorlesung

<https://dmi.unibas.ch/de/studium/computer-science-informatik/aktuelles-semester-hs/vorlesung-grundlagen-der-programmierung/>

->Grundlagen der Programmierung

- hier gibt es alle Folien zur Vorlesung
- aber auch sonstige aktuelle Informationen

-> Erweiterte Grundlagen der Programmierung

- Übungsblätter
- aber auch sonstige aktuelle Informationen

Lernziele

A) Grundlagen der Programmierung

Die Programmiersprache Java kennen.

Kleine Programme lesen und verstehen können und kurze Programmteile selbst schreiben und entwickeln können.

B) Erweiterte Grundlagen der Programmierung

Ganze Programme in Java selbst schreiben können und Erfahrung in der Umsetzung von Problemstellungen zu lauffähigen Programmen sammeln.

Überblick zur Vorlesung

Inhalt:

- Anweisungen und Datenstrukturen
- Objektorientiertes Programmieren
- Programmentwurf
- Rekursion
- Fehlerbehandlung
- Generics
- Verwenden von Java-API Bibliotheksfunktionen

Literatur

Folien zur Vorlesung werden im Web abgelegt

Folien allein sind NICHT ausreichend!!

Bücher:

- 1) Sprechen sie Java,
Hanspeter Mössenböck
dpunkt Verlag etwa sFr 48,-



5. Auflage



(2.Auflage)

Ein erstes Java Programm.

Allgemeines zu Java

Java ist eine objektorientierte Programmiersprache in der Tradition von C und C++.

Die Java-Syntax lehnt sich an C an, nimmt aber einige deutliche Bereinigungen vor

- z. B. Abwesenheit von goto und dem Datentyp Zeiger (pointer).

Die Erweiterung um Objekte geschieht ähnlich wie bei C++

- Allerdings gibt es doch einige Unterschiede:
 - In erster Näherung könnte man sagen, dass Java aus der riesigen Vielfalt der C++ Konstrukte eine sinnvolle Auswahl trifft und gleichzeitig einiges umstellt und bereinigt.

„Home Page“ von Java

Java wurde bei der Firma SUN Microsystems entwickelt

- Von James Gosling
 - Ursprünglich für die Programmierung von in Haushaltselektronik eingebetteten Prozessoren

Die zentrale Internet Site der Firma Oracle zum Thema Java ist

- <http://www.oracle.com/technetwork/java/index.html>

Eine freie (Open Source) Implementation von Java ist das OpenJDK

Besonders empfehlenswert (und sehr empfohlen zur Ergänzung des Selbststudiums):

- Das „offizielle“ Java-Tutorial: <http://docs.oracle.com/javase/tutorial/>

Java Bibliotheken

Zu Java gehört außerdem eine Sammlung von **standardisierten Bibliotheken**, die häufige Programmieraufgaben stark erleichtern.

Beispiele solcher Bibliotheken sind

- **java.awt** für das Erzeugen graphischer Benutzeroberflächen
- **java.math** für Arithmetik auf beliebig langen Zahlen
(wichtig z. B. für die Kryptographie),
- **java.net** zum Betreiben von Verbindungen über das Internet
- **java.rmi** zum Aufrufen von Methoden auf entfernten Rechnern
- ...



Installation der Entwicklungsumgebung

Die Standard-Entwicklungsumgebung, Java Development KIT (JDK) kann direkt im Internet heruntergeladen werden

<https://adoptopenjdk.net>

- Eine Installationsanleitung finden Sie auf der Vorlesungsseite.
- Bei Problemen helfen die Tutoren gerne.

Die Installation vom JDK ist bis Ende 2. Woche zwingend notwendig.



Java-Programme erstellen und ausführen

3-Stufen-Prozess:

1. Programm schreiben und in einer Datei mit der Endung „**.java**“ abspeichern. (z.B. *Dateiname.java*).
Hierzu benötigen wir einen Texteditor (z.B. Visual Studio code)
2. Programm mit dem Java-Compiler übersetzen.
Dies geschieht durch aufrufen von **javac** und der Compiler erzeugt dabei eine Datei mit der Endung „**.class**“ (z.B. *Dateiname.class*).
3. Programm mit der Java-Virtual-Machine ausführen.
Dies geschieht durch aufrufen von **java**

Grundstruktur von Java- Programmen

```
class ProgramName {  
    public static void main (String[] arg) {  
        ... // Deklarationen  
        ... // Anweisungen  
    }  
}
```

Text muß in einer Datei namens *ProgramName.java* stehen

Beispiel

```
class MyFirstProgram{  
    public static void main (String[] arg) {  
        System.out.println( "Hallo World!");  
    }  
}
```

Text steht in Datei *MyfirstProgram.java*



Editoren

Programme zum „Schreiben“ und „Verändern“ (Editieren) von Text-Dateien heißen **Editoren**.

Quellcode-Dateien von Java-Programmen sind **Text-Dateien**

Benutzen Sie Ihren „Lieblingseditor“

- Windows: Notepad++, Geany, Jedit...
- Linux: vi, VIM, (EMACS), Geany, ...
- Mac OSX: TextWrangler,

Wir empfehlen: „**Visual Studio Code**“ !



Übersetzen und Ausführen mit JDK

```
class MyFirstProgram{  
    public static void main (String[] arg) {  
        System.out.println( "Hallo World!");  
    }  
}
```

Text steht in Datei
MyFirstProgram.java

Übersetzen

```
C:\ cd MySamples  
C:\ javac MyFirstProgram.java
```

wechselt ins Verzeichnis mit der Quelldatei
erzeugt Datei *MyFirstProgram.class*

Ausführen

```
C:\ java MyFirstProgram  
Hallo World!
```

ruft main- Methode der Klasse
MyFirstProgram auf



Pragmatisches zur Programmierung in Java

- **Jedes Programm wird in (mindestens) einer Datei abgelegt, deren Dateiname die Endung **.java** aufweisen muss.**

z.B.:

Helloworld.java

- **Vor der ersten Ausführung eines Programms muss das Programm in eine interne Form in einer internen, maschinennahen Sprache übersetzt werden. Die Übersetzung geschieht durch Aufruf des **Java-Compilers** unter Angabe des Dateinamens:**

javac Helloworld.java

- **Der Java-Compiler generiert aus jeder Klassendeklaration in der übersetzten Datei eine eigene Datei mit internem Code, die den Namen der jeweiligen Klasse mit Endung **.class** trägt - also im Beispiel nur eine Datei**

Helloworld.class



Pragmatisches zur Programmierung in Java

- Nach dem Übersetzen kann dann die Datei mit internem Code zu einer der aufrufbaren Klassen gestartet werden:

`java Helloworld`

- Dadurch wird das **Java-Laufzeitsystem** zum Aufruf der main-Methode dieser Klasse aufgefordert. Alle weiteren Aktivitäten des Programms hängen dann vom Inhalt dieser Methode ab (Objekterzeugung, Aufruf anderer Methoden ggf. anderer Klassen)
- Etwas gewöhnungsbedürftig:



Jupyter Notebooks

File Edit View Insert Cell Kernel Help Trusted Java O

In [43]:

```
Turtle t = new Turtle();
t.home();
t.backward(30);
t.turnRight(90);
t.backward(75);
t.penDown();
drawSierpinskit(t, 150, 10);
t.toImage();
```

Out [43]:

A complex, black-and-white fractal pattern known as a Sierpinski triangle. It is composed of numerous small, nested triangles that form a larger, overall triangular shape. The pattern is highly detailed and symmetric, with each side of the main triangle containing approximately 10 levels of recursion.

Jshell: Interaktives Programmieren

```
C:\Users\luetma00>jshell
| Welcome to JShell -- Version 10.0.1
| For an introduction type: /help intro

jshell> System.out.println("hello world");
hello world

jshell>
```

- Jshell: Interaktives Programmieren in Java
- Teil von Java seit Java 9

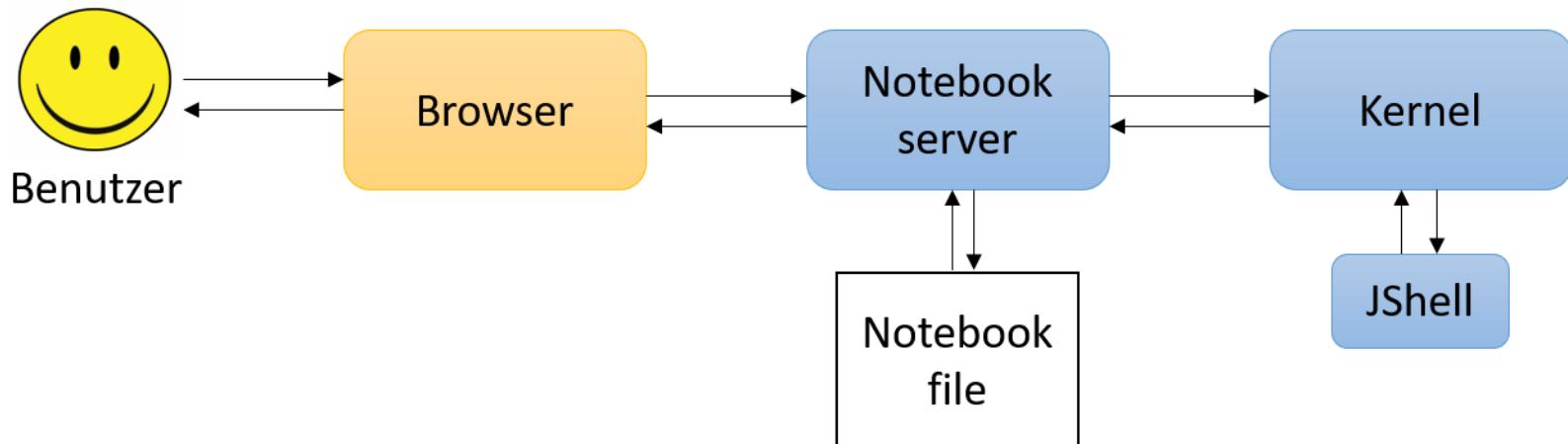
Jshell: Vorteile

- Einfach kurze Teile auszuprobieren
- Kein Schreiben von Klassenrumpf und main Methode nötig
- Kompilieren und Ausführen stark vereinfacht

Gut zum lernen und ausprobieren.

Was sind Jupyter Notebooks

- Webbasierte Programmierumgebung
 - Idee: Vereinen von Dokumentation und Programme
 - Unterstützt viele verschiedene Programmiersprachen.
- In Java: Basierend auf JShell



Jupyter Notebooks: Unser Einsatz

- Unterstützung Theorie während Vorlesung
- Mini-übungen als interaktives Element
- Individuelles Lernen und Prüfungsvorbereitung

Zuweisungen und arithmetische Funktionen

Durch das Arbeiten mit Variablen können wir Programme allgemeiner schreiben und von konkreten Beispielen abstrahieren. Als Beispiel nehmen wir die Berechnung der Hypotenuse eines rechtwinkligen Dreiecks. Anstatt die Berechnung direkt mit konkreten Werten durchzuführen, können wir Variablen einführen und die Formel dann allgemein hinschreiben.

```
In [15]: M double a = 3;  
double b = 4;
```

Nun wissen wir vom Satz von Pythagoras, dass die Länge der Hypotenuse c mit der Formel $c^2 = a^2 + b^2$ berechnet werden kann.

```
In [20]: M double aSquared = a * a;  
double bSquared = b * b;  
double cSquared = aSquared + bSquared;
```

Um c zu erhalten müssen wir noch die Wurzel ziehen. Dafür stellt uns Java die Funktion `Math.sqrt` zur Verfügung.

```
In [21]: M double c = Math.sqrt(cSquared);
```

Die Werte der Variablen können wir dies mit dem Befehl `System.out.println(variable)` ausgeben lassen.

```
In [22]: M System.out.println(c);  
5.0
```

Mini Übung

- Definieren Sie eine Variable mit dem Namen `z` die nur ganze Zahlen enthalten kann und weisen Sie dieser den Wert 3 zu.
- Was passiert, wenn Sie statt 3 den Wert 3.0 zuordnen?
- Wie müssen Sie den Typ ändern, damit die Zuweisung `z = 3.0` funktioniert?

```
In [24]: M // Schreiben Sie Ihre Lösungen hier hin. Neue Zellen können Sie mit Alt + Enter hinzufügen.
```

Verzweigung

Im Programmieren muss man oft zwischen verschiedenen Fällen unterscheiden. Dafür können wir die `if else` Anweisung benutzen. Als erstes Beispiel schauen wir uns ein Programm an, das entscheidet ob die Variable `z` einen positiven oder negativen Wert beinhaltet.

```
In [97]: M int z = 5;
```

Demo

In [43]:

```
Turtle t = new Turtle();
t.home();
t.backward(30);
t.turnRight(90);
t.backward(75);
t.penDown();
drawsierpinskis(t, 150, 10);
t.toImage();
```

Out[43]:

A fractal Sierpinski triangle, also known as a Sierpinski gasket or Sierpinski sieve. It is a self-similar fractal pattern formed by recursively subdividing a large triangle into smaller triangles. The pattern consists of many small black triangles on a white background, creating a complex, organic shape.

Jupyter Notebooks: Nachteile

- Entwicklung grösserer Programme nicht möglich
- Keine Unterstützung von professionellen Entwicklungsumgebungen
- Teile von Notebooks können in beliebiger Reihenfolge ausgeführt werden → Verwirrungspotenzial

Super zum lernen, nicht geeignet zur Entwicklung grosser Programme.

Installation

- Installation via Docker oder Virtualbox auf lokalem Computer
- Link zu Installationsanleitung auf Vorlesungswebseite
- Hilfe bei Problemen in den Übungsgruppen



Ziel: Alle TeilnehmerInnen haben Jupyter-Notebooks auf Ihren Laptops installiert.

Laptops bitte mit in die Vorlesung bringen.

Was ist Virtualbox / Docker?



- Statt Anwendungen werden ganze “virtuelle” Maschinen installiert.
- Maschine enthält alle Abhängigkeiten (Bibliotheken, Daten, Betriebssystem ...).
- Gewünschte Anwendung (Jupyter Notebooks) wird in virtueller Maschine ausgeführt.

Übungen

Durchführung der Übungen

A) Grundlagen der Programmierung

Alle Übungen werden über ein Web-Interface in einen Web-Browser durchgeführt. Die Auswertung und die Korrekturhilfen werden automatisch erstellt.
Bei Fragen steht ein Forum auf Adam zur Verfügung.

B) Erweiterte Grundlagen der Programmierung

Sie entwickeln Programme mit den Standard Java Entwicklungswerkzeugen. Ihr Code wird von unseren Tutoren korrigiert. Hilfe erhalten Sie in den Tutoraten.

Plagiate

Ein Plagiat ([. . .] aus lateinisch *plagiarius*, deutsch "Seelenverkäufer, Menschenräuber") ist die Anmaßung fremder geistiger Leistungen. Dies kann sich auf die Übernahme fremder Texte oder anderer Darstellungen[. . .], fremder Ideen [.] oder beidesgleichzeitig [.] beziehen.

Folge:

- Verwarnung und 0 Punkte in Übungsblatt (1 Mal)
- Nicht-zulassung zur Prüfung (Wiederholungsfall)

Bei Unsicherheit: Fragen, wir helfen gerne!

Plagiatserkennung: JPlag

Java Programm um Ähnlichkeit von Java Programmen zu erkennen.

Matches sorted by average similarity ([What is this?](#)):

[download csv](#)

anonymized-31	->	anonymized-0 (100.0%)	anonymized-\$i (100.0%)						
anonymized-100	->	anonymized-36 (100.0%)	anonymized-21 (100.0%)	anonymized-108 (100.0%)	anonymized-71 (100.0%)	anonymized-65 (100.0%)	anonymized-52 (100.0%)	anonymized-42 (100.0%)	
anonymized-75	->	anonymized-1 (100.0%)							
anonymized-108	->	anonymized-36 (100.0%)	anonymized-21 (100.0%)	anonymized-71 (100.0%)	anonymized-65 (100.0%)	anonymized-52 (100.0%)	anonymized-42 (100.0%)		
anonymized-15	->	anonymized-104 (100.0%)							
anonymized-43	->	anonymized-103 (100.0%)							
anonymized-21	->	anonymized-71 (100.0%)	anonymized-65 (100.0%)	anonymized-52 (100.0%)	anonymized-42 (100.0%)	anonymized-36 (100.0%)			
anonymized-91	->	anonymized-19 (100.0%)							
anonymized-2	->	anonymized-17 (100.0%)							
anonymized-63	->	anonymized-12 (100.0%)							

7er Gruppen?

2-er Gruppen

Übungsgruppen

Tutorate erste Woche:

- Hilfe bei Java Installation
- Hilfe bei Jupyter Installation
- Beantwortung allgemeiner Fragen
- Besprechung erstes Übungsblatt (nur erweiterte Grundlagen)

Termine

Mo	08:15 - 10:00
Mo	16:15 - 18:00
Di	16:15 - 18:00
Mi	12:15 - 14:00
Mi	14:15 - 16:00
Mi	16:15 - 18:00
Fr	13:15 - 15:00

Erste Woche (also 23-28 September): Offen für alle
Danach: Nur für Studierende der erweiterte Grundlagen

Organisation der Übungsgruppen

Für Studierende erweiterte Grundlagen:

- Jetzt bitte 3 Prioritäten für Übungsgruppen angeben

The screenshot shows the homepage of Tweedback. At the top, there is a large, stylized illustration of a diverse crowd of people. Below the illustration, there are two main sections: 'Join a session' and 'Create a session'. The 'Join a session' section contains a text input field labeled 'Enter Session-ID ...', a 'Join' button, and a link 'Sessions of this browser'. The 'Create a session' section contains a 'Create session' button and a link 'Sessions of this browser'. Below these sections, there are four icons with corresponding text descriptions: 'HONEST' (a hand icon), 'INTERACTIVE' (two speech bubbles icon), 'REALTIME' (a clock icon), and 'PRIVACY' (a lock icon). The 'REALTIME' and 'PRIVACY' sections include small explanatory text. At the bottom of the page, the website URL 'www.tweedback.de' is displayed.

HONEST
Increase participation through anonymous feedback.

INTERACTIVE
Transform listeners into active participants.

REALTIME
Use immediate feedback with live evaluation.

PRIVACY
GDPR compliant with servers in German datacenters.

www.tweedback.de