

Tutorium zur Vorlesung Programmieren

1. Tutorium

Joshua Gleitze

31. Oktober 2016

IPD Koziolk

Hi!

Organisatorisches

sdqweb.ipd.kit.edu/wiki/Vorlesung_Programmieren_WS16/17

Tutoriumsbeginn "Programmieren" WS 2016/17

Die Tutorien für Programmieren beginnen am 26.10.2016.

Die Folien dieses Tutoriums sind unter

joshuagleitze.de/tutorien/programmieren1617

verfügbar.

Meine Email-Adresse ist

joshua.gleitze@student.kit.edu

- Java ist auch eine Insel
 - allgemeine Funktionen der Sprache
 - Übersicht über APIs
- Java API Javadoc
 - Vorrangige Quelle für die API
- Stack Overflow
 - Stilfragen
 - Konkrete Probleme

Entwicklungsumgebung einrichten

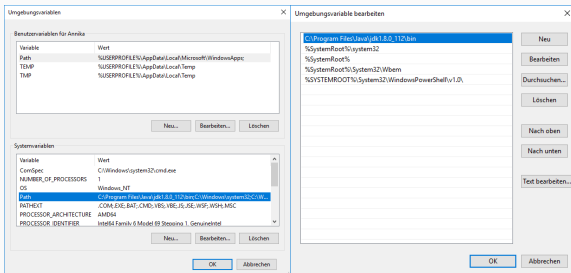
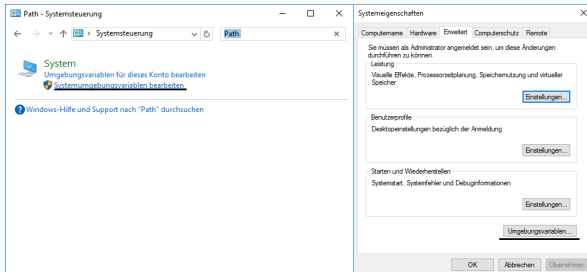
Installation unter Ubuntu:

```
sudo apt install openjdk-8-jre icedtea-8-plugin openjdk-8-jdk
```

Installation unter Windows:

- aktuelle Version von t1p.de/jdk herunterladen und Installer ausführen
- PATH anpassen (nächste Folie)

WINDOWS: JDK IN PATH EINTRAGEN



Aktuelle Version: Eclipse Neon

Download von eclipse.org/downloads

Installation unter Linux:

- Installer entpacken und ausführen
- t1p.de/eclipseDesktop herunterladen
 - in `~/.local/share/applications` speichern
 - als ausführbar markieren

- ```
mkdir -p ~/.icons
ln -s ~/eclipse/java-neon/eclipse/icon.xpm ~/.icons/eclipse.xpm
```

Installation unter Windows: Installer ausführen

Verwendet für die ersten 3 Übungsblätter  
*keine* IDE!

# Kompilieren

---

# JAVA: KOMPILIEREN UND INTERPRETIEREN

## Kompilieren

*einmalig zur Kompilierzeit*



`javac Program.java`



## Interpretieren

*immer zur Laufzeit*



`java Program`



# KOMPILIEREN UND AUSFÜHREN IM TERMINAL

Kompilieren der Datei `Test.java`:

```
javac Test.java
```

Ausführen des kompilierten Programms:

```
java Test
```

Beides gemeinsam:

```
javac Test.java && java Test
```

Aliase für Verzeichnisse:

Aktuelles Verzeichnis:

```
.
```

Elternverzeichnis:

```
..
```

Home-Verzeichnis:

```
~
```

Navigation im Dateisystem:

Verzeichnis wechseln:

```
cd <Verzeichnis>
```

Inhalt auflisten:

```
ls <Verzeichnis>
```

## Streams:

Ausgabe in Datei schreiben:

```
befehl > output.txt
```

Ausgabe an Datei anhängen

```
befehl >> output.txt
```

Inhalt der Datei als Eingabe:

```
befehl < input.txt
```

Ausgabe des 1. Befehls an den 2.:

```
befehl1 | befehl2
```



Hintereinanderausführung:

Erst 1, dann 2:

```
befehl1; befehl2
```

2 falls 1 erfolgreich:

```
befehl1 && befehl2
```

2 falls 1 nicht erfolgreich:

```
befehl1 || befehl2
```

# Ein erstes Java-Programm

---

# main-METHODE

```
1 public class TutorialProgram {
2 public static void main(String[] args) {
3 ...
4 }
5 }
```

```
java TutorialProgram Hallo Welt foo bar
```

# main-METHODE

```
1 public class TutorialProgram {
2 public static void main(String[] args) {
3 ...
4 }
5 }
```

```
java TutorialProgram Hallo Welt foo bar
```

args ist die Liste der Kommandozeilenparameter:

|         |         |
|---------|---------|
| args[0] | "Hallo" |
| args[1] | "Welt"  |
| args[2] | "foo"   |
| args[3] | "bar"   |

# Übung

---

Anmelden unter [judge.joshuagleitze.de](https://judge.joshuagleitze.de)  
Lösungen werden automatisiert geprüft.

## 1. Echo

Der 1. Kommandozeilenparameter soll auf der Konsole ausgegeben werden

## 2. Zitat

Die Kommandozeilenparameter sollen wie folgt ausgegeben werden:

```
1 Hallo <1. Parameter>, du sagst: '<2. Parameter>'
2 Das war gar nicht so schwer!
```