



HOCHSCHULE ZITTAU/GÖRLITZ  
University of Applied Sciences

**PH**Bern  
Pädagogische Hochschule

# PROGRAMMING-WIKI: ONLINE PROGRAMMIEREN UND KOMMENTIEREN

M. Hielscher und Chr. Wagenknecht

# Handouts, Grundgerüste und Hilfsfunktionen

## Aufgabe 1: Die Klasse KFZ

In der letzten Woche haben wir die Klasse **Autohaus** erarbeitet. Fügen Sie in der Klasse **Autohaus** nun eine Variable für die Liste aller Fahrzeuge hinzu, die dieses Autohaus verkauft. Verwenden Sie hierfür den Datentyp *ArrayList*.

Im Ordner `H:\Informatik\GK10-1\Java\Woche 3` finden Sie ein Grundgerüst für die Klasse **KFZ**.



Arbeitsplatz  
Computerpool



Server  
Netzlaufwerk



privater  
Computer

# Handouts, Grundgerüste und Hilfsfunktionen

swisseduc.ch

Home · Info · Kontakt · Suchen

Informatik » Tic Tac Toe in Java

**Informatik auf SwissEduc**  
» Programmiersprachen

- Karatojava
- Turtles
- Turtles mit Java
- Povray: Grafik programmieren
- Java-Kurs für Einsteiger
- Java: Apfelmännchen
- Lego-Robotik mit Java
- Handy-Programmierung mit Java
- Java: Tic Tac Toe**
  - Programmieren des Spiels
- Geschichte der Programmiersprachen
- Rekursive Programme
- Programmier-Paradigmen
- Corewars - Duell der Programme
- Einführung in Prolog
- Brainfuck - esoterische Programmiersprache
- Scratch - Werkstatt
- Dynamische Webseiten
- XML: Einführung
- XML: Übersicht

## Programmieren des Spiels "Tic Tac Toe" in Java

Verfasst von *Thomas Dübendorfer*



**Aufgaben** + **Server**

Worum geht es

Anhand eines anderen...  
Voraussetzung so...  
schrittweise zu einen...  
Java AWT, Event Handling (inkl. Interfaces) und File I/O (inkl. Exceptions) anzuwenden und zu vertiefen. Zusätzlich werden einige weitere Übungen und Materialien zur Einführung dieser Techniken bereitgestellt. Zwei anschauliche Beispiele führen Interfaces (Monsterbeispiel) und Exceptions (Münzen teilen) spielerisch ein.

...erfaces); File I/O (inkl. Exceptions). Als  
...wie das Programmgerüst von Tic Tac Toe  
...gegner erweitert werden kann, um die Techniken von

## Downloads zu "Programmieren des Spiels "Tic Tac Toe" in Java"

	Tic Tac Toe Framework, ZIP-Datei	ZIP [13 KB]
	Tic Tac Toe Lösung, ZIP-Datei	ZIP [15 KB]
	Handout mit Sourcecode zum Tic Tac Toe Programmgerüst	PDF [202 KB]
	Handout mit Sourcecode zur fertigen Version von Tic Tac Toe	PDF [188 KB]
 	Arbeitsblatt GUI 1 (AWT: Frames, Label, List)	PDF [23 KB]

# Entwicklungsumgebungen wie Eclipse

The image shows a screenshot of the Eclipse IDE interface. The main editor window displays the code for `SplashWindow.java`, which includes several private static fields and a Singleton constructor. A central overlay, consisting of a rounded rectangle with a blue border, contains two illustrations of computer setups. The left illustration, labeled 'Arbeitsplatz' (Workplace), shows a desktop computer with a monitor, keyboard, and mouse, crossed out with a large red 'X'. The right illustration, labeled 'Computer zu hause' (Computer at home), shows a similar setup with a CRT monitor and a tower PC, also crossed out with a large red 'X'. Below this overlay, a red callout box with a white border contains the text: 'Komplex, für die ersten Unterrichtseinheiten viel zu viele Funktionen!' (Complex, for the first teaching units too many functions!). The IDE's Package Explorer on the left shows a project structure with packages like `matchix` and `matchix.controller`. The Problems view at the bottom shows an `InternalError` message.

```
//  
////////////////////////////////////  
private Image image;  
private Image background;  
private static int x;  
private static int y;  
private static SplashWindow instance = null;  
private boolean paintCalled = false;  
  
new InternalError  
error.initCause(e);  
throw error;  
  
/*  
 * Singleton Const
```

Arbeitsplatz

Computer zu hause

**Komplex, für die ersten Unterrichtseinheiten viel zu viele Funktionen!**

# Entwicklungsumgebungen wie BlueJ

The screenshot shows the BlueJ development environment. The main window displays a class hierarchy with 'Database' and 'Staff' classes, and an abstract class 'Person'. A 'Person' class editor is open, showing the source code for 'setYearOfBirth'. A 'Database' class editor is also open, showing the source code for 'Database()'. A callout box highlights two computer icons: 'Arbeitsplatz' (workplace) and 'Computer zu hause' (computer at home). A red callout box contains the text: 'Installationsaufwand, Konfigurations- und Kompatibilitätsprobleme für SuS sind nicht zu unterschätzen.' (Installation effort, configuration and compatibility problems for students are not to be underestimated.)

BlueJ: people

Project Edit Tools View Help

New Class...  
-->  
->  
Compile

Database

Database

Staff

Person

Class Edit Tools Options

Compile Undo Cut Copy Paste Find... Find Next Close Source Code

```
public void setYearOfBirth(int newYearOfBirth)
```

Arbeitsplatz

Computer zu hause

Database

Class Edit Tools Options

Compile Undo Cut Copy Paste Find... Find Next Close Source Code

```
public Database()  
{  
    persons = new ArrayList();  
}  
  
/**  
 * Add a person to the database.  
 */  
public void addPerson(Person p)  
{  
    persons.add(p);  
}
```

Generating documentation... D

saved

**Installationsaufwand,  
Konfigurations- und  
Kompatibilitätsprobleme für SuS  
sind nicht zu unterschätzen.**

# Unterrichtsmaterialien online



[Seite](#) [Diskussion](#) [Quelltext betrachten](#) [Versionen/Autoren](#)

## Java/for-Schleife

[< Java](#)

Häufig benötigt man beim Programmieren eine mehrfache Ausführung eines Teilbereichs. Denkbar ist es, dass ein Programmierer dazu einfach Zeilen wiederholt. So lässt sich z.B. das kleine 1 x 1 von 3 wie folgt ausgeben:

```
System.out.println("3 x 1 = 3");
System.out.println("3 x 2 = 6");
System.out.println("3 x 3 = 9");
System.out.println("3 x 4 = 12");
System.out.println("3 x 5 = 15");
System.out.println("3 x 6 = 18");
System.out.println("3 x 7 = 21");
System.out.println("3 x 8 = 24");
System.out.println("3 x 9 = 27");
```

Eleganter ist da schon die Verwendung der [while-Schleife](#)

```
int faktor = 1;
while ( faktor <= 9 ) {
    System.out.println("3 x " + faktor + " = " + 3*faktor );
    faktor++;
}
```

Für genau diesen Fall gibt es aber auch eine Schleife: die for-Schleife. Sie ist immer dann sinnvoll einsetzbar, wenn

- Eine bestimmte Anzahl von Wiederholungen durchgeführt werden soll
- Eine Variable von einem bestimmten Wert bis zu einem anderen gezählt werden soll

Das folgende Beispiel macht genau das selbe wie die while-Schleife im obigen Beispiel

```
for ( int faktor = 1; faktor <= 9; faktor ++ ) {
    System.out.println("3 x " + faktor + " = " + 3*faktor );
}
```

Weitere Beispiele

### Kurzinfo



Dieser Artikel gehört zum Thema Programmieren in Java.



Dieser Artikel enthält Unterrichtsideen, die du ausprobieren oder hier diskutieren kannst.

Suche

Navigation

- [Hauptseite](#)
- [Forum](#)
- [Letzte Änderungen](#)
- [Zufällige Seite](#)
- [Spickzettel](#)
- [Hilfe](#)

ZUM Internet

- [ZUM.de](#)
- [ZUM-Grundschulnetz](#)
- [ZUM-Unity](#)

Google-Anzeigen

### Partnersuche

Ich bin nicht gerne Single-und Sie? Treffen wir uns bei PARSHIP.ch [www.PARSHIP.ch](http://www.PARSHIP.ch)

### java programmierung

Java Programmierer und Projekte finden Sie auf Zeitgeist at

# Unterrichtsmaterialien online



Seite Diskussion Quelltext betrachten Versionen/Autoren

## Java/for-Schleife

< Java

Häufig benötigt man beim Programmieren eine mehrfache Ausführung eines Teilbereichs. Denkbar ist es, dass ein Programmierer dazu einfach Zeilen wiederholt. So lässt sich z.B. das kleine 1 x 1 von 3 wie folgt ausgeben:

```
System.out.println("3 x 1 = 3");
System.out.println("3 x 2 = 6");
System.out.println("3 x 3 = 9");
System.out.println("3 x 4 = 12");
System.out.println("3 x 5 = 15");
System.out.println("3 x 6 = 18");
System.out.println("3 x 7 = 21");
System.out.println("3 x 8 = 24");
System.out.println("3 x 9 = 27");
```

### Programm abarbeiten

```
3 x 1 = 3
3 x 2 = 6
3 x 3 = 9
3 x 4 = 12
3 x 5 = 15
3 x 6 = 18
3 x 7 = 21
3 x 8 = 24
3 x 9 = 27
```

Eleganter ist da schon die Verwendung der [while-Schleife](#)

```
int faktor = 1;
while ( faktor <= 9 ) {
    System.out.println("3 x " + faktor + " = " + 3*faktor );
    faktor++;
}
```

Für genau diesen Fall gibt es aber auch eine Schleife: die for-Schleife. Sie ist immer dann sinnvoll einsetzbar, wenn

• Eine bestimmte Anzahl von Wiederholungen durchgeführt werden soll

### Kurzinfo



Dieser Artikel gehört zum Thema Programmieren in Java.



Dieser Artikel enthält Unterrichtsideen, die du ausprobieren oder hier diskutieren kannst.

Suche

Navigation

- [Hauptseite](#)
- [Forum](#)
- [Letzte Änderungen](#)
- [Zufällige Seite](#)
- [Spickzettel](#)
- [Hilfe](#)

ZUM Internet

- [ZUM.de](#)
- [ZUM-Grundschulnetz](#)
- [ZUM-Unity](#)

Google-Anzeigen

### Partnersuche

Ich bin nicht gerne Single-und Sie? Treffen wir uns bei PARSHIP.ch [www.PARSHIP.ch](http://www.PARSHIP.ch)

### java programmierung

Java Programmierer und Projekte finden Sie auf Zeitgeist at

# Wie sollte es sein ?

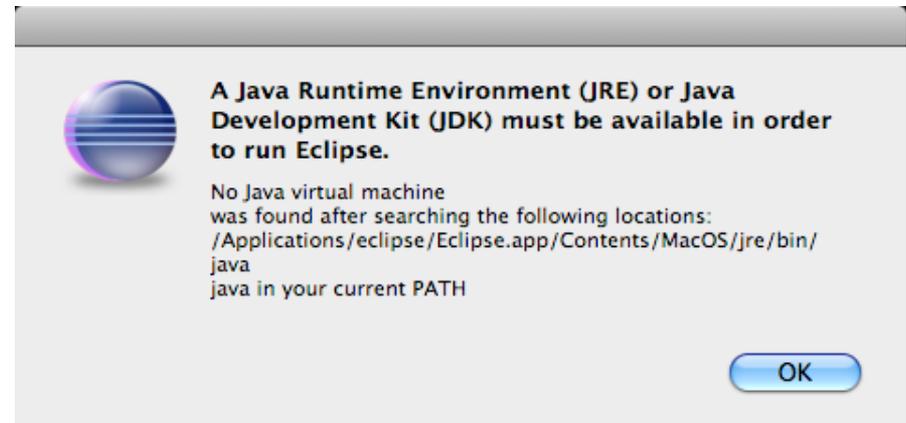


es sollte einfach sein  
(Komplexität der IDE)

Lehrmaterialien

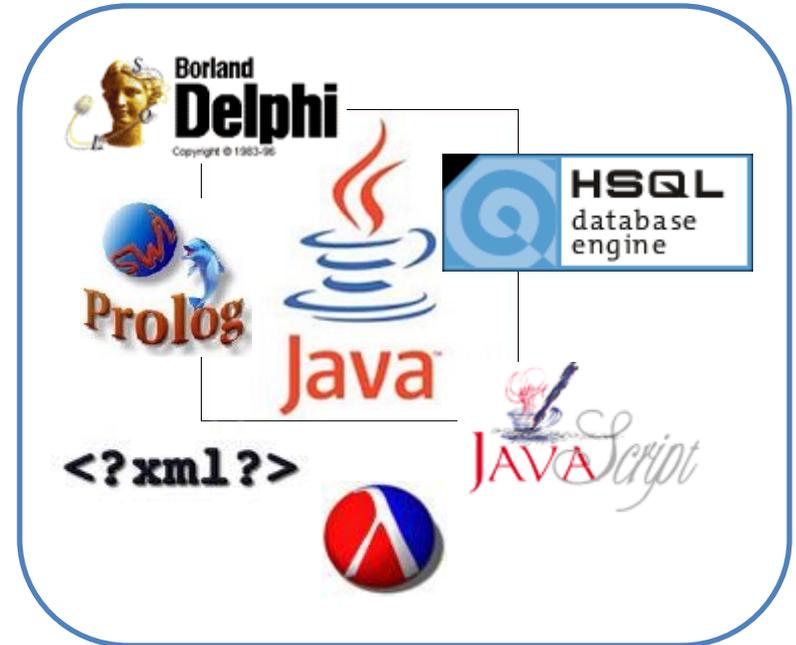


Programmierübungen



wenig Installationsaufwand

# Online programmieren und kommentieren



**PROGRAMMING WIKI**



# Programming Wiki

Java/for-Schleife - Program...

Benutzer Eigene Diskussion Einstellungen Beobachtungsliste Eigene Beiträge Abmelden

## PROGRAMMING WIKI



### Navigation

- Startseite
- Letzte Änderungen
- Hilfe

### Suche

### Werkzeuge

- Links auf diese Seite
- Änderungen an verlinkten Seiten
- Hochladen
- Spezialseiten
- Druckversion
- Permanentlink

Chat zur Seite anzeigen

Seite

Diskussion

Bearbeiten

Versionen/Autoren

Verschieben

Beobachten

## Java/for-Schleife

Häufig benötigt man beim Programmieren eine mehrfache Ausführung eines Teilbereichs. Denkbar ist es, dass ein Programmierer dazu einfach Zeilen wiederholt.

So lässt sich z.B. das kleine 1 x 1 von 3 wie folgt ausgeben:

```
x 1 System.out.println("3 x 1 = 3");
2 System.out.println("3 x 2 = 6");
3 System.out.println("3 x 3 = 9");
4 System.out.println("3 x 4 = 12");
5 System.out.println("3 x 5 = 15");
6 System.out.println("3 x 6 = 18");
7 System.out.println("3 x 7 = 21");
8 System.out.println("3 x 8 = 24");
9 System.out.println("3 x 9 = 27");
10
```

### speichern & ausführen

```
> 3 x 1 = 3
3 x 2 = 6
3 x 3 = 9
3 x 4 = 12
3 x 5 = 15
3 x 6 = 18
3 x 7 = 21
3 x 8 = 24
3 x 9 = 27
```

**Programmieren direkt im Lehrtext.**

Fertig

# Demo

## Die Spieluhr

[Bearbeiten]

Du kennst die Tonleiter und die Notenschrift aus dem Musikunterricht. Für Drehorgeln und Spieluhren wird hingegen eine Art Code in Form von Stiften verwendet, die auf einer Walze aufgebracht sind. Im Bild rechts sieht man das Innenleben einer Spieluhr.

In dieser Aufgabe sollst du versuchen



Ausführung

```
String Lied = "E-4 E-4 E-2 E-4 E-4 E-2  
"D-4 E-1 F-4 F-4 F-4 F-4  
"E-8 E-8 E-4 D-4 D-4 E-4
```

eine Darstellung für eine zugehörige Walze einer S das Lied auch mit Hilfe der Methoden, die wir letzte lassen.



```
1 canvas.clear();  
2 String Lied = "E-4 E-4 E-2 E-4 E-4  
3 "D-4 E-1 F-4 F-4 F-4 F-4  
4 "E-8 E-8 E-4 D-4 D-4 E-4  
5  
6 orgel = new Sound(19); // 19. MIDI-  
7  
8 String[] Noten = Lied.split(" ");  
9  
10 int posX = 0;  
11 int posY = 0;  
12 for(int i=0; i < Noten.length; i++)  
13 int key = 0;  
14 if (Noten[i].charAt(0) == 'C') ke  
15 if (Noten[i].charAt(0) == 'D') ke  
16 if (Noten[i].charAt(0) == 'E') ke  
17 if (Noten[i].charAt(0) == 'F') ke  
18 if (Noten[i].charAt(0) == 'G') ke  
19 if (Noten[i].charAt(0) == 'A') ke  
20 if (Noten[i].charAt(0) == 'H') ke  
21 int dauer = 0;  
22 if (Noten[i].charAt(2) == '1') da  
23 if (Noten[i].charAt(2) == '2') da  
24 if (Noten[i].charAt(2) == '4') da  
25 if (Noten[i].charAt(2) == '8') da  
26  
27 canvas.fillRect(posX, 80-(key-60)*  
28 posX = posX + dauer/20+5;  
29  
30 orgel.play(key, dauer);  
31 }  
32
```

ausführen

## SQL Bankraub

Inhaltsverzeichnis [Verbergen]

- 1 Was ist passiert
- 2 Die Hinweise
- 3 Die Datenbank
- 4 Die Aufgabe

## Was ist passiert

[Bearbeiten]

Ein Gängstertrio hat am 23.08.2009 die Central Bank in New York ausgeräumt. Es waren Profis wie der Polizeichef bestätigte: "Hier waren ganz klar Kenner mit Erfahrung am Werk. Sie haben den Coup sicher schon über Wochen im Vorfeld geplant."

## Die Hinweise

[Bearbeiten]

Von den Augenzeugen haben wir nur zwei Hinweise auf die **drei** männliche Täter erhalten:

### 1. Hinweis:

Ein Täter bedrohte die Angestellten und wurde von einem Komplizen „Lie“ gerufen. Er soll mindestens 1'80 groß gewesen sein, zwischen 20 und 30 Jahre alt und trug einen auffallend chinesischen Kleidungsstil.

### 2. Hinweis:

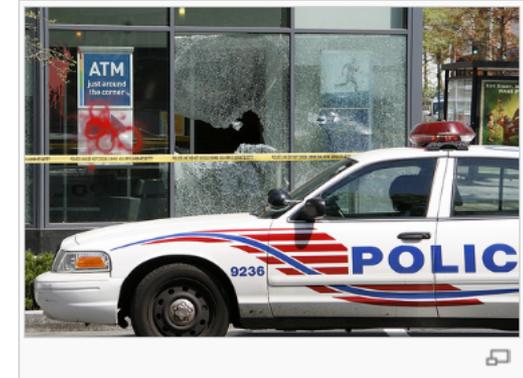
Eine Augenzeugin erinnert sich an das Fluchtfahrzeug. Es ist ein blauer oder schwarzer BMW gewesen sagt sie und das Kennzeichen soll auf 557 geendet haben.

## Die Datenbank

[Bearbeiten]

Als FBI-Ermittler und Datenbank-Spezialist bist du für diesen Fall zuständig. Du hast bereits 100 Personen herausgesucht, die als Täter in Frage kommen könnten. Du hast Zugriff auf folgende Tabellen der FBI-Datenbank:

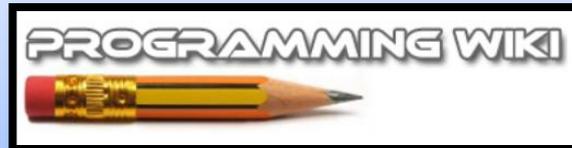
Die Spalten Vater, Mutter, Fahrzeughalter und Person sind Fremdschlüssel auf die Spalte ID der Tabelle Einwohner.



# Programming Wiki Architektur



**PHP Erweiterung**



# Programming Wiki Architektur



## PHP Erweiterung



# Programming Wiki Architektur

The screenshot shows a web browser window with the URL "Java/for-Schleife - Program...". The page title is "PROGRAMMING WIKI" with a pencil icon. The main content is the article "Java/for-Schleife", which includes a code block for a Java program that prints multiplication results from 3 x 1 to 3 x 9. Below the code is a "speichern & ausführen" button. A diagram overlay on the right side of the page illustrates the architecture of a "Java - Applet". The applet acts as a "einheitliche Schnittstelle" (uniform interface) between the user's code and an "Interpreter" (HSQL database engine). The diagram shows a box labeled "Java - Applet" containing a box labeled "Interpreter" which includes the "HSQL database engine" logo. Arrows indicate the flow of data and control between the code, the applet interface, and the interpreter.

Navigation

- Startseite
- Letzte Änderungen
- Hilfe

Suche

Seite Suchen

Werkzeuge

- Links auf diese Seite
- Änderungen an verlinkten Seiten
- Hochladen
- Spezialseiten
- Druckversion
- Permanentlink

Chat zur Seite anzeigen

Fertig

## Java/for-Schleife

Häufig benötigt man beim Programmieren eine mehrfach...  
Denkbar ist es, dass ein Programmierer dazu einfach Zeilen...  
So lässt sich z.B. das kleine 1 x 1 von 3 wie folgt ausgeben:

```
x 1 System.out.println("3 x 1 = 3");  
2 System.out.println("3 x 2 = 6");  
3 System.out.println("3 x 3 = 9");  
4 System.out.println("3 x 4 = 12");  
5 System.out.println("3 x 5 = 15");  
6 System.out.println("3 x 6 = 18");  
7 System.out.println("3 x 7 = 21");  
8 System.out.println("3 x 8 = 24");  
9 System.out.println("3 x 9 = 27");  
10
```

**speichern & ausführen**

```
> 3 x 1 = 3  
3 x 2 = 6  
3 x 3 = 9  
3 x 4 = 12  
3 x 5 = 15  
3 x 6 = 18  
3 x 7 = 21  
3 x 8 = 24  
3 x 9 = 27
```

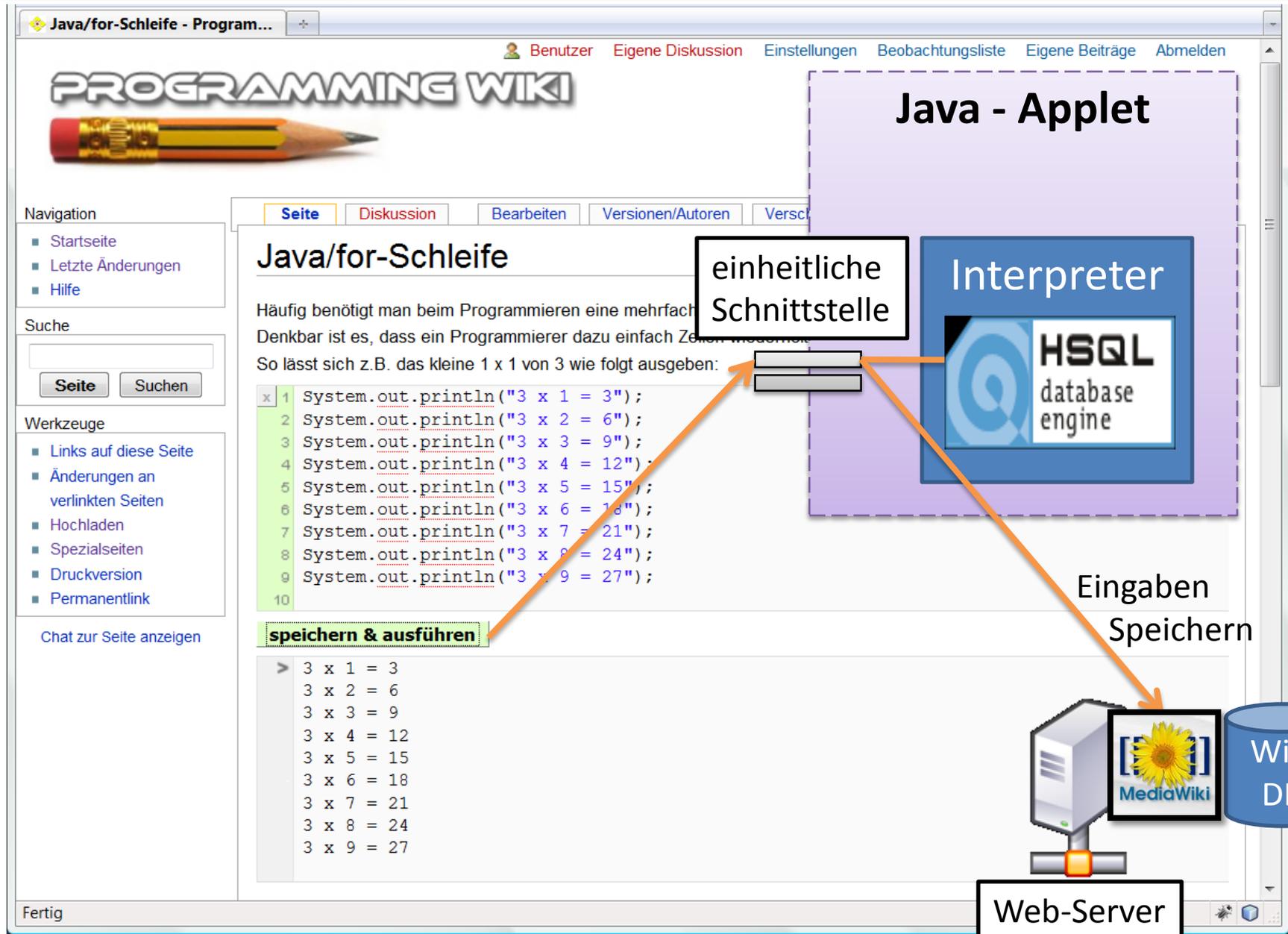
**Java - Applet**

einheitliche Schnittstelle

Interpreter

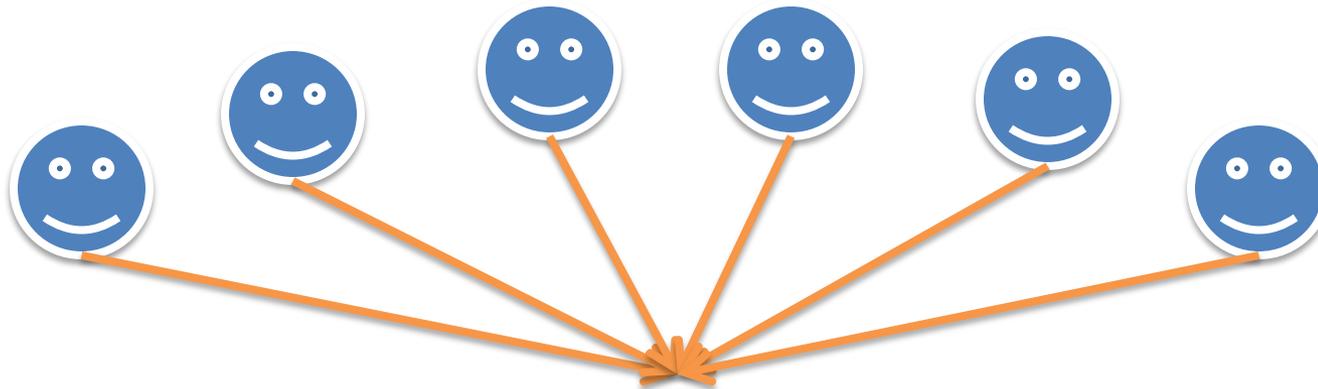
HSQL database engine

# Programming Wiki Architektur



# Programming Wiki Architektur

Schüler und Schülerinnen



```
x 1 System.out.println("Hallo Welt");  
2
```



Prima, deine Lösung scheint zu stimmen.

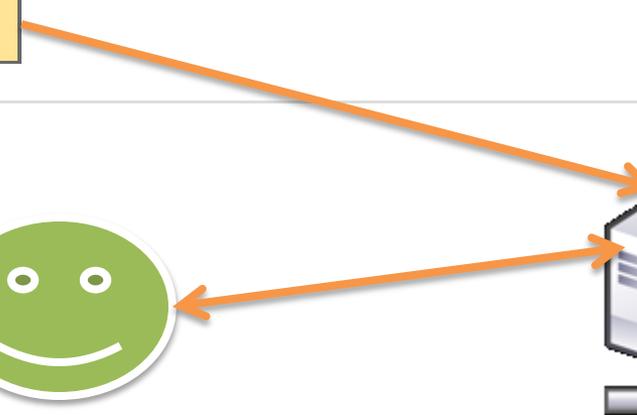
jetzt prüfen



Lehrperson

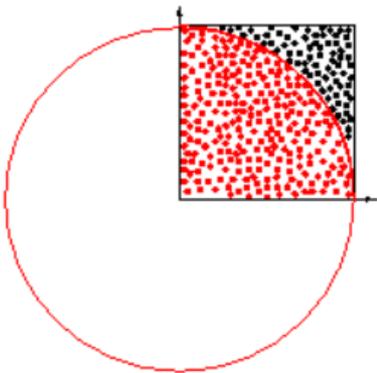


Web-Server



# Erfahrungen

Erarbeiten von Studi-Referaten: am Beispiel der LV "Algorithmen und Komplexität"



Da man eine Chancengleichheit der Treffer auf der gesamten Fläche voraussetzen kann, auf einen Viertelkreis im ersten Quadranten zu beschränken, wie das Bild darstellt. Für gilt dann:  $x^2 + y^2 \leq 1$  für reelle  $x$  und  $y$  und  $0 < x, y \leq 1$ .

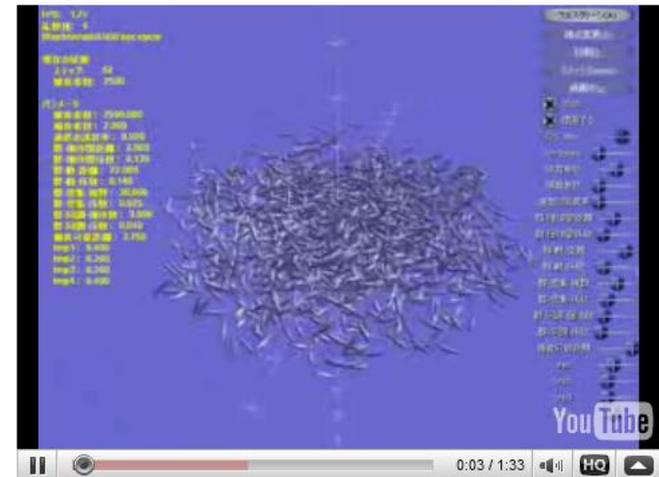
$$\text{Also gilt } \frac{A_{\text{Kreis}}}{A_{\text{Quadrat}}} = \frac{\pi r^2}{4r^2} \approx \frac{T_{\text{Kreis}}}{T_{\text{Quadrat}}}$$

$$\pi \approx 4 \frac{T_{\text{Kreis}}}{T_{\text{Quadrat}}}$$

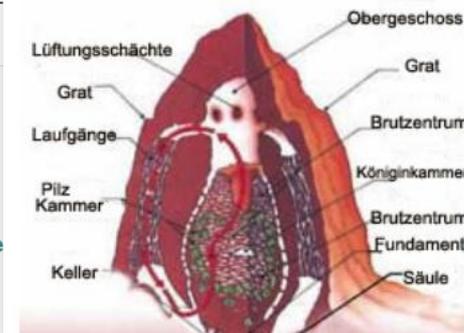
Die Genauigkeit der Berechnung wird dabei mit zunehmender "Tropfenzahl" erhöht. Er implementiert ist, findet sich hier:

```
x | 1 public static double berechne_pi(int tropfenzahl) {
  | 2     double pi = 0;
  | 3     int innerhalb = 0;
  | 4     int gesamt = tropfenzahl;
  | 5
  | 6     while (tropfenzahl > 0) { // generiere Tropfen und addiere
  | 7         double dotx = Math.random();
  | 8         double doty = Math.random();
  | 9
  |10        if (dotx*dotx + doty*doty <= 1) { // Punkt liegt innerhalb des Kreises
```

Im Videobeispiel sehen wir etwa einen Fischschwarm, bei dem sich jeder Fisch einfach dem nächsten Fisch und der Entfernung zu den Raubfischen orientiert. Es gibt also keinen wer wo lang zu schwimmen hat.



Durch Schwarmverhalten schaffen es Termiten z. B. auch extrem komplexe Bauten zu weiß, wie der gesamte Bau aussehen und es keine "Super-Termite" gibt, welche den



# Erfahrungen

## Selbststudienkurs am Beispiel der Logik-basierten Programmierung mit Prolog

### Navigation



[Startseite](#)

[Was ist ein Wiki?](#)

[Hilfe zur Formatierung](#)

[Compiler hochladen](#)

[Canvas Beschreibung](#)

[Turtle Beschreibung](#)

[Letzte Änderungen](#)

### Suche



Seite

Suchen

### Werkzeuge



[Links auf diese Seite](#)

[Hochladen](#)

[Spezialseiten](#)

[Druckversion](#)

[Permanenter Link](#)

[PDF Version](#)

[Chat zur Seite anzeigen](#)

### Fakten und Anfragen

[\[Bearbeiten\]](#)

Die folgenden Angaben bilden die Faktenbasis, auf die wir uns im Folg. immer wieder beziehen werden.

```
x| 1 /* Fakten */
   2
   3 weiblich(christine) .
   4 weiblich(anna) .
   5 weiblich(doris) .
   6 maennlich(hans) .
   7 maennlich(gerhard) .
   8 maennlich(christian) .
   9 maennlich(martin) .
  10
```

Einstellige Prädikate (ohne Variablen) sind *Fakten*, genauer: *Eigenschaften* von Objekten, wie z.B. "Christine ist weiblich." Da gerade die Stelligkeit dieser Prädikate für deren Verwendung von großer Bedeutung ist, gibt man diese durch einen Schrägstrich getrennt nach dem Prädikatnamen an: weiblich/1.

Fakten kann man erfragen (*Anfrage*) und erhält erwartungsgemäß eine Bestätigung:

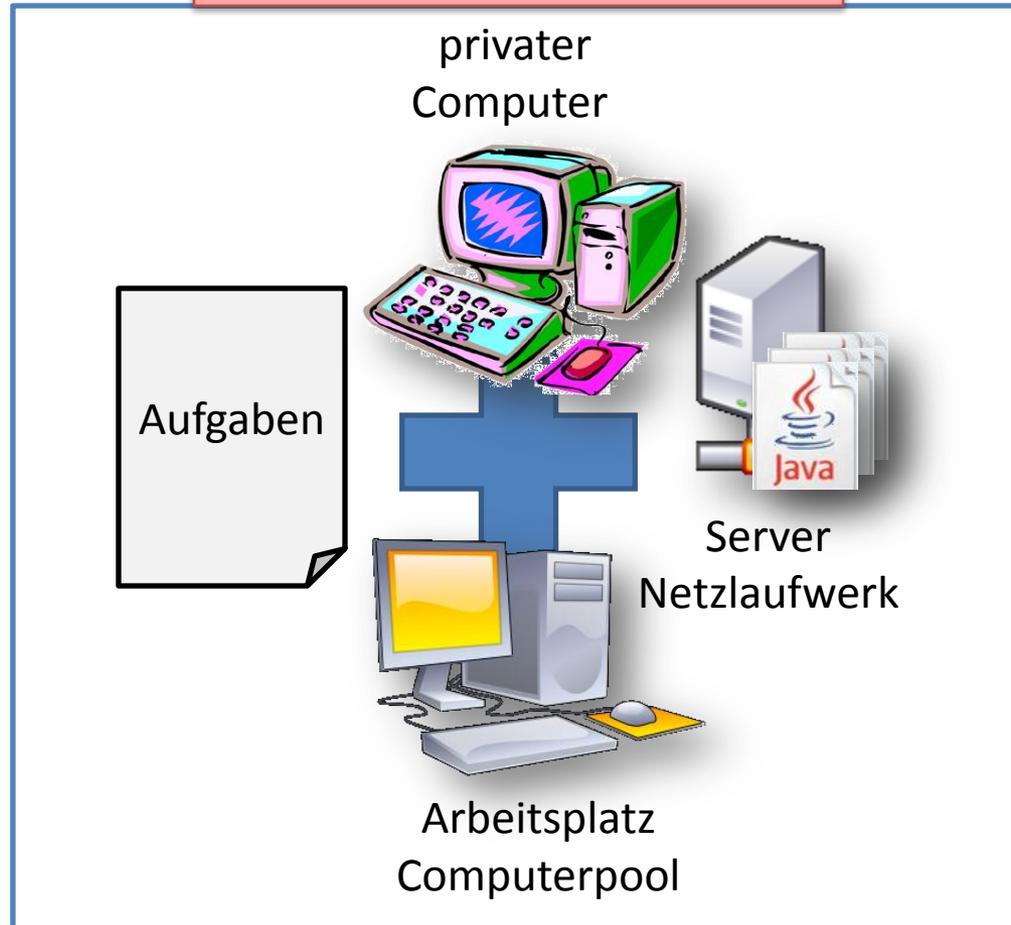
```
x| 1 weiblich(christine) .
   2
```

**ausführen**

```
> weiblich(christine)
   yes
```

# Online Programmieren: Vorteile

## All-in-One-Lösung



# Online Programmieren: Vorteile



Keine Installation für Schüler und Schülerinnen  
(Browser + JRE ausreichend)

# Online Programmieren: Vorteile

© Original Artist

Reproduction rights obtainable from  
[www.CartoonStock.com](http://www.CartoonStock.com)



keine verlorenen Dateien durch falsches Speichern/Transport/Programmversionen

I DON'T REMEMBER THE TITLE, BUT IT WAS  
ON A LITTLE PIECE OF WHITE PAPER.

# Online Programmieren: Vorteile

SuS-Lösungen können von der Lehrperson geprüft, kommentiert oder verbessert werden.

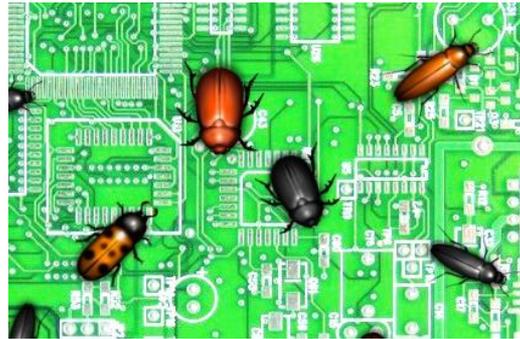
**Programming-Wiki Lösungen** (Nachfolgende Benutzern haben auf dieser Seite ihre Ergebnisse abgespeichert.):

Benutzer;

## Lösungen von: Benutzer

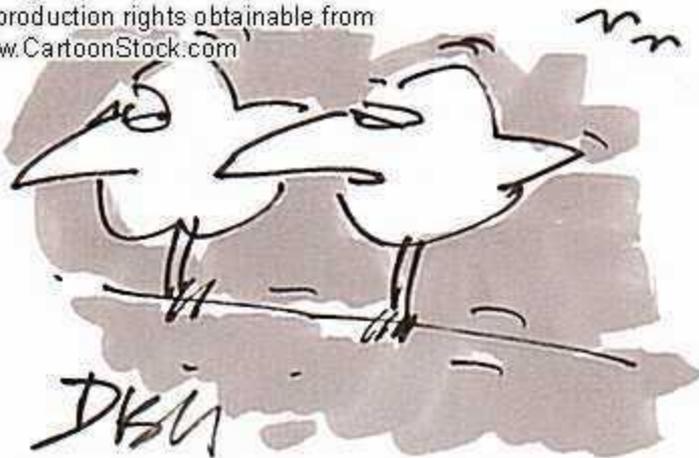
```
x 1 canvas.clear();
2 String Lied = "E-4 E-4 E-2 E-4 E-4 E-2 E-4 G-4 C-4 "+
3             "D-4 E-1 F-4 F-4 F-4 F-4 F-4 E-4 E-4 "+
4             "E-8 E-8 E-4 D-4 D-4 E-4 D-2 G-2";
5
6 orgel = new Sound(19); // 19. MIDI-Gerät = Orgel
7
8 String[] Noten = Lied.split(" ");
9
10 int posx = 0;
11 int posy = 0;
12 // Diese Schleife führt über das Ziel hinaus. Beachte, dass die Zählung bei 0 beginnt!
13 for(int i=0; i <= Noten.length; i++){
14     int key = 0;
```

# Online Programmieren: Nachteile



man hat nicht den vollen Umfang einer IDE

© Original Artist  
Reproduction rights obtainable from  
[www.CartoonStock.com](http://www.CartoonStock.com)



search ID: ndi0633

"I see you're on line."

Internetzugang nötig



Lehrperson muss  
ein Wiki betreuen

gestalten Sie selbst eine interaktive Wikiseite...

[http://michael-hielscher.de/wiki/index.php/Ihre\\_Seite](http://michael-hielscher.de/wiki/index.php/Ihre_Seite)

Seite

Diskussion

Bearbeiten

Versionen/Autoren

## Bearbeiten von Ihre Seite

Du bearbeitest diese Seite unangemeldet. Wenn du speicherst, wird deine aktuelle IP-Adresse in der Versionsgeschichte aufgezeichnet und ist damit unwiderruflich **öffentlich** einsehbar.



Programmierung: **Java**

Eval

Code

Hidden

Check

Canvas

Vielen Dank für die  
Aufmerksamkeit  
<eval>answerQuestions()</eval>