

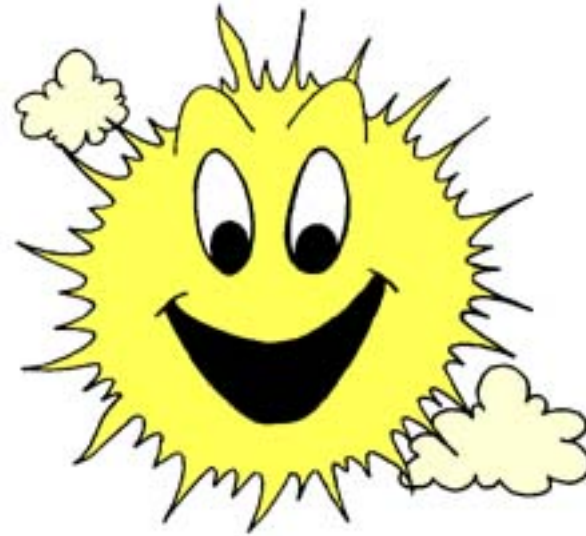
Entwurf und Realisierung grafischer Oberflächen



Einführung in
Benutzungsoberflächen, Java, ...

FH Darmstadt, Fachbereich Informatik
WS 2005/06, Prof. Dr. Ralf Mayer

Welcome ...



Herzlich willkommen bei der Lernveranstaltung

Entwurf und Realisierung grafischer Oberflächen

1. Themen für heute

1. Organisation
2. Lernziele
3. Einführung

1.1 Organisatorisches

1. Leistungsnachweis (Klausur)
2. Vorlesung
3. Praktikum
4. Lernunterlagen

1.1.1 Organisatorisches

- Prof. Ralf Mayer, Raum 1.07, Gebäude D14
- Web: <http://www.fbi.fh-darmstadt.de/~rmayer>
- Email: r.mayer@fbi.fh-darmstadt.de
- Sprechzeiten
donnerstags 11:00 – 11:45 ab 13.10.
und nach Vereinbarung
- **Klausur** am **01.02.2006** 10:15 - 11:45
D14/404+D14/403
 - Online Belegung der Klausur bis **23.01.2006**
 - Abmeldung bis **28.01.2006**



1.1.2 Struktur der Vorlesung

- Vorlesung
 - Wiederholung
 - am Anfang der Vorlesung
 - Hörsaalübungen
 - neue Themen
 - Life Programmierung
- Nachbearbeitung
 - Skripte, Literatur
 - Aufgaben, Übungen, ...
- Fragen auch im Rahmen der Wiederholung
- Klausurvorbereitung

1.1.2 Methoden des Lernens

- Lehrgespräch
 - Gemeinsame Erarbeitung neue Themen
 - Wiederholung
 - Ihre Fragen
- Vortrag mit Tafel und Beamer
 - Vermittlung neuer Inhalte
- Verarbeiten und Festigen der Lerninhalte
 - Hörsaalübungen (jeder für sich)
 - 2-er Gruppen
 - Murmelgruppe („6x6“)
 - → Zusammentragen der Ergebnisse
- Individuelle Arbeit – Wiederholung – Übung

1.1.3 Allgemeines zum Praktikum

- 6 Termine
- Anwesenheitspflicht
Fehlen nur mit ärztlichem Attest
- Testat bei jedem Termin
 - Vorbereitung
 - Erfolgreiche Durchführung der Aufgabe
- Aufgaben
 - 1. Termin: Einführung Java, IDE, Programmiertechniken
 - 2. – 6. Termin: Entwicklung einer Anwendung
- Bestätigung der erfolgreichen Teilnahme
 - alle 6 Testate
 - Abnahme der Gesamtlösung
 - Dokumentation der Lösung

1.1.3c Praktikum (3C)

FH Darmstadt FB Informatik		Bachelor 3C					Stundenplan WS 05/06	
	Montag	Dienstag	Mittwoch	Donnerstag	Freitag			
08:30 - 10:00			Wolfgang Weber P: Softwaretechnik II D14/211 30.4316					
10:15 - 11:45	Ralf S. Mayer V: Entwurf und Realisierung grafischer Oberflächen D14/204 30.4306		Wolfgang Weber P: Softwaretechnik II D14/211 30.4316	Sabine Lauer V: Grundlagen der Statistik D14/404 30.4310	Stephan Karczewski V: Datenbanken I D14/404 30.4304			
12:00 - 13:30	Elke Hergenröther V: Grafische Datenverarbeitung I D14/304 30.4308	Wolfgang Weber V: Softwaretechnik II D14/404 30.4316		Gerhard Raffius V: Mikroprozessorsysteme II D14/304 30.4314	Uta von Grootte V: Betriebssysteme D14/404 30.4302			
14:15 - 15:45	Steffen Lange, Sandra Zilles V: Grundlagen der Theoretischen Informatik D14/104 30.4312		Uta von Grootte V: Betriebssysteme D14/404 30.4302	Uta von Grootte P: Betriebssysteme y D14/310 30.4302 Ralf S. Mayer P: Entwurf und Realisierung grafischer Oberflächen x D15/107 30.4306 Gerhard Raffius P: Mikroprozessorsysteme II D10/032 30.4314 Wolfgang Weber P: Softwaretechnik II D14/211 30.4316	Uta von Grootte P: Betriebssysteme y D14/310 30.4302			
16:00 - 17:30	Steffen Lange, Sandra Zilles V: Grundlagen der Theoretischen Informatik x D14/104 30.4312 Steffen Lange, Sandra Zilles P: Grundlagen der Theoretischen Informatik y D14/104, D14/103 30.4312		Ramin Seghatoleslam P: Entwurf und Realisierung grafischer Oberflächen y D15/107 30.4306	Uta von Grootte P: Betriebssysteme y D14/310 30.4302 Christian Rehm P: Entwurf und Realisierung grafischer Oberflächen x D15/107 30.4306 Gerhard Raffius P: Mikroprozessorsysteme II D10/032 30.4314 Wolfgang Weber P: Softwaretechnik II D14/211 30.4316	Uta von Grootte P: Betriebssysteme y D14/310 30.4302			
17:45 - 19:15				Christian Rehm P: Entwurf und Realisierung grafischer Oberflächen x D15/107 30.4306				

[Fb Informatik] [B. Kreling] [05.10.2005]

1.1.3c Termine Praktikum 3B

Praktikum Zug 3C:

Ort	D14/1.07			
Zeit (Betreuer)	Mi 5y 16:00-17:30 (R. Seghatoleslam)	Do 4x 14:15-15:45 (Prof. R. Mayer)	Do 5x 16:00-17:30 (C. Rehm)	Do 6x 17:45-19:15 (C. Rehm)
Termin 1	02.11.05	20.10.05	20.10.05	20.10.05
Termin 2	23.11.05	03.11.05	03.11.05	03.11.05
Termin 3	07.12.05	24.11.05	24.11.05	24.11.05
Termin 4	21.12.05	08.12.05	08.12.05	08.12.05
Termin 5	11.01.06	05.01.06	05.01.06	05.01.06
Termin 6	25.01.06	19.01.06	19.01.06	19.01.06

1.1.3b Praktikum (3B)

FH Darmstadt
FB Informatik

Bachelor 3B

Stundenplan
WS 05/06

	Montag	Dienstag	Mittwoch	Donnerstag	Freitag
08:30 - 10:00	Klaus Frank P: Mikroprozessorsysteme II D10/032 30.4314	Wolfgang Weber V: Softwaretechnik II D14/104 30.4316	Dankwart Zimmerling V: Betriebssysteme D14/004 30.4302		Dankwart Zimmerling V: Betriebssysteme D14/004 30.4302
10:15 - 11:45	Klaus Frank P: Mikroprozessorsysteme II D10/032 30.4314	Wolf-Dieter Groch V: Grafische Datenverarbeitung I D14/304 30.4308	Horst Mayer V: Grundlagen der Statistik B11/4 30.4310	Wolfgang Weber P: Softwaretechnik II D14/211 30.4316	Dankwart Zimmerling P: Betriebssysteme D15/102 30.4302
12:00 - 13:30		Klaus Frank V: Mikroprozessorsysteme II D14/104 30.4314	Herwig Meyer P: Grundlagen der Theoretischen Informatik D14/404 30.4312	Ralf S. Mayer P: Entwurf und Realisierung grafischer Oberflächen x D15/107 30.4306 Wolfgang Weber P: Softwaretechnik II D14/211 30.4316	Dankwart Zimmerling P: Betriebssysteme D15/102 30.4302
14:15 - 15:45	Heinz-Erich Erbs V: Datenbanken I D14/404 30.4304		Wolfgang Weber P: Softwaretechnik II y D14/211 30.4316	Herwig Meyer V: Grundlagen der Theoretischen Informatik D14/004 30.4312	
16:00 - 17:30	Ralf S. Mayer V: Entwurf und Realisierung grafischer Oberflächen D14/204 30.4306		Ramin Seghatoleslam P: Entwurf und Realisierung grafischer Oberflächen x D15/107 30.4306 Wolfgang Weber P: Softwaretechnik II x, y D14/211 30.4316	bis 16:45 Herwig Meyer V: Grundlagen der Theoretischen Informatik D14/004 30.4312	
17:45 - 19:15			Ramin Seghatoleslam P: Entwurf und Realisierung grafischer Oberflächen D15/107 30.4306 Wolfgang Weber P: Softwaretechnik II x D14/211 30.4316		

[Fb Informatik] [B. Kreling] [05.10.2005]

1.1.3b Termine Praktikum 3B

Praktikum Zug 3B:

Ort	D14/1.07			
Zeit (Betreuer)	Mi 5x 16:00-17:30 (R. Seghatoleslam)	Mi 6x 17:45-19:15 (R. Seghatoleslam)	Mi 6y 17:45-19:15 (R. Seghatoleslam)	Do 3x 12:00-13:30 (Prof. R. Mayer)
Termin 1	26.10.05	26.10.05	02.11.05	20.10.05
Termin 2	16.11.05	16.11.05	23.11.05	03.11.05
Termin 3	30.11.05	30.11.05	07.12.05	24.11.05
Termin 4	14.12.05	14.12.05	21.12.05	08.12.05
Termin 5	04.01.06	04.01.06	11.01.06	05.01.06
Termin 6	18.01.06	18.01.06	25.01.06	19.01.06

1.1.4 Literatur Grundlagen

- Java Dokumentation, SUN Microsystems (Ver. 1.4.2 oder 5.0) <http://java.sun.com/docs/>
- Tutorial von Bruce Eckels: Thinking Java 3rd Ed. <http://www.mindview.net/Books>
- Handbuch der Java-Programmierung, Guido Krüger, <http://www.javabuch.de>
- Sun: The Java Tutorial + Suchapplet <http://java.sun.com/docs/books/tutorial/uiswing/components/components.html>
<http://java.sun.com/docs/books/tutorial/information/download.html>

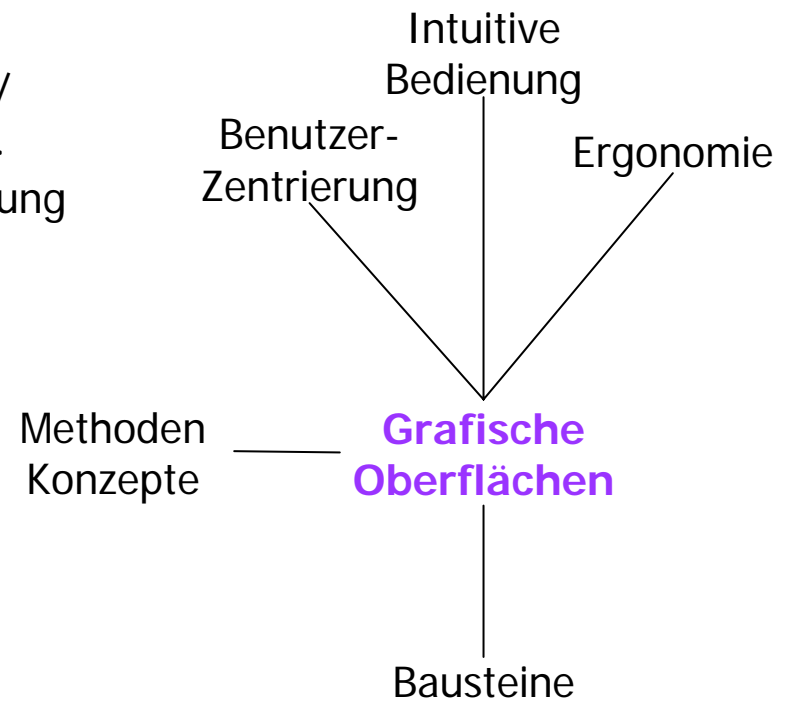
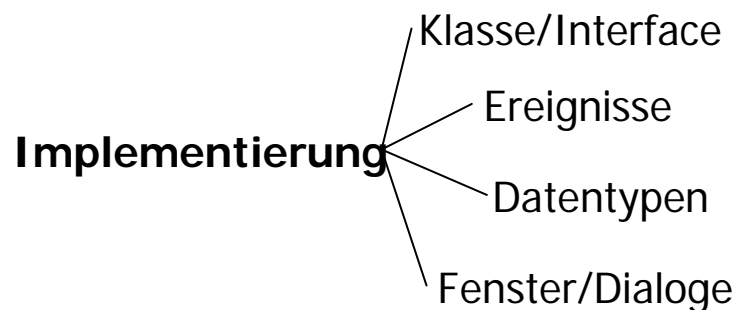
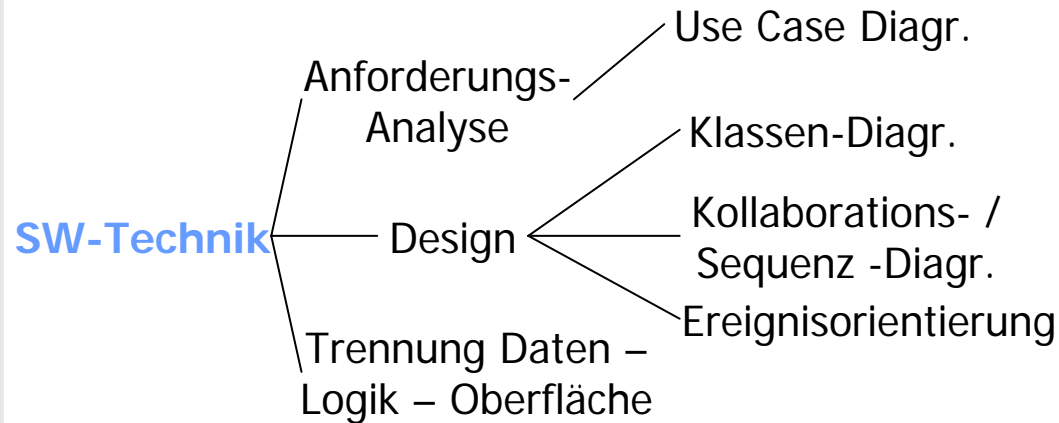
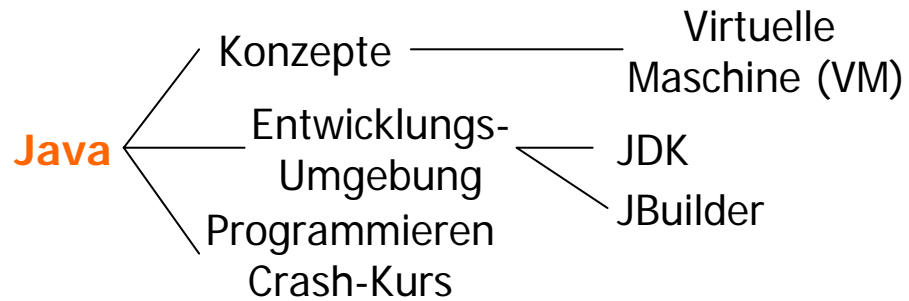
1.1.4 Literatur Grundlagen (2)

- *Grundkurs Programmieren in Java;*
Ratz, Scheffler, Seese;
Band 1; Hanser Verlag
- *Grundkurs Programmieren in Java;*
Ratz, Scheffler, Seese, Wiesenberger;
Band 2; Hanser Verlag
- *Java in a Nutshell ;* David Flanagan; O'Reilly Verlag
- *Java Foundation Classes in a Nutshell;* David Flanagan;
O'Reilly Verlag

1.1.4 Literatur weiterführend

- Java Look and Feel Design, SUN Microsystems
<http://java.sun.com/products/jlf/>

1.2 Lernziele für ERGO



1.3 Einführung

- Einführung in die Sprache JAVA
- Arten von Computeranwendungen
- Schnittstelle Mensch – Maschine
 - häufige Probleme
 - grafische Benutzungsoberflächen
- Ziele bei der Entwicklung guter „Anwendungen“
- Entwicklung von Anwendungen
 - Bedeutung der Entwurfsphase
 - Analyse – Design – Realisierung

1.3.1 Was ist Java?

Java

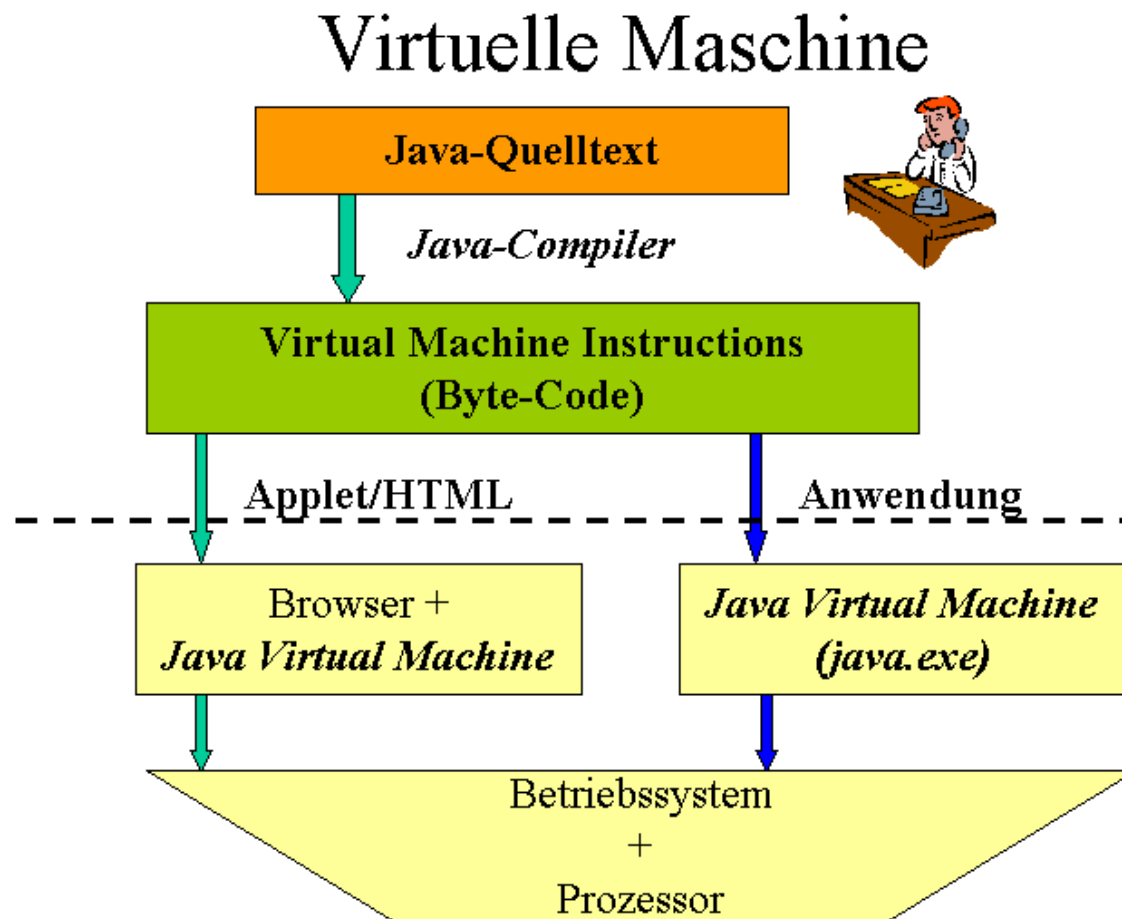
- arbeitet nach dem **Interpreter**-Prinzip
- ist **plattformunabhängig**
- ist streng **objektorientiert**
- ist **sicher**
- ist **ausgereift**



Java war ursprünglich für kleine Anwendungen konzipiert;

wurde dann aber zur **Universalsprache** mit vielen **Spezialbibliotheken** ausgebaut.

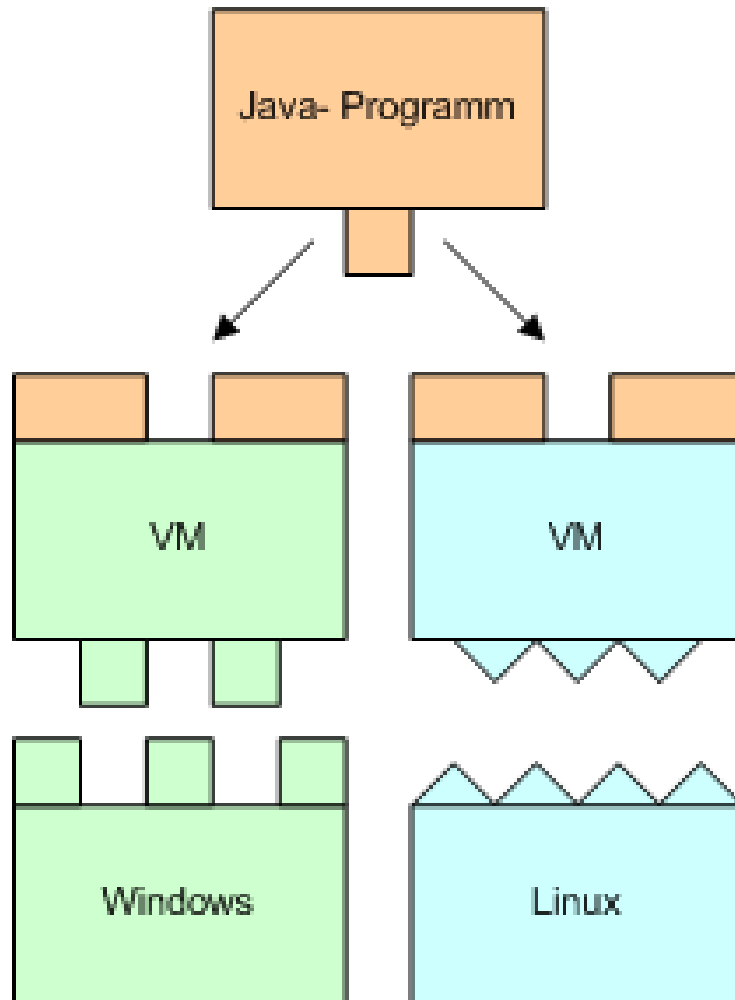
1.3.1.1 Java - Interpreter



Java-Quelltext wird vom *Compiler* in **Byte-Code** (Programm) übersetzt.

Der **Byte-Code** wird von der *virtuellen Maschine* auf dem **Betriebssystem** ausgeführt.

1.3.1.2 Java – warum plattformunabhängig?

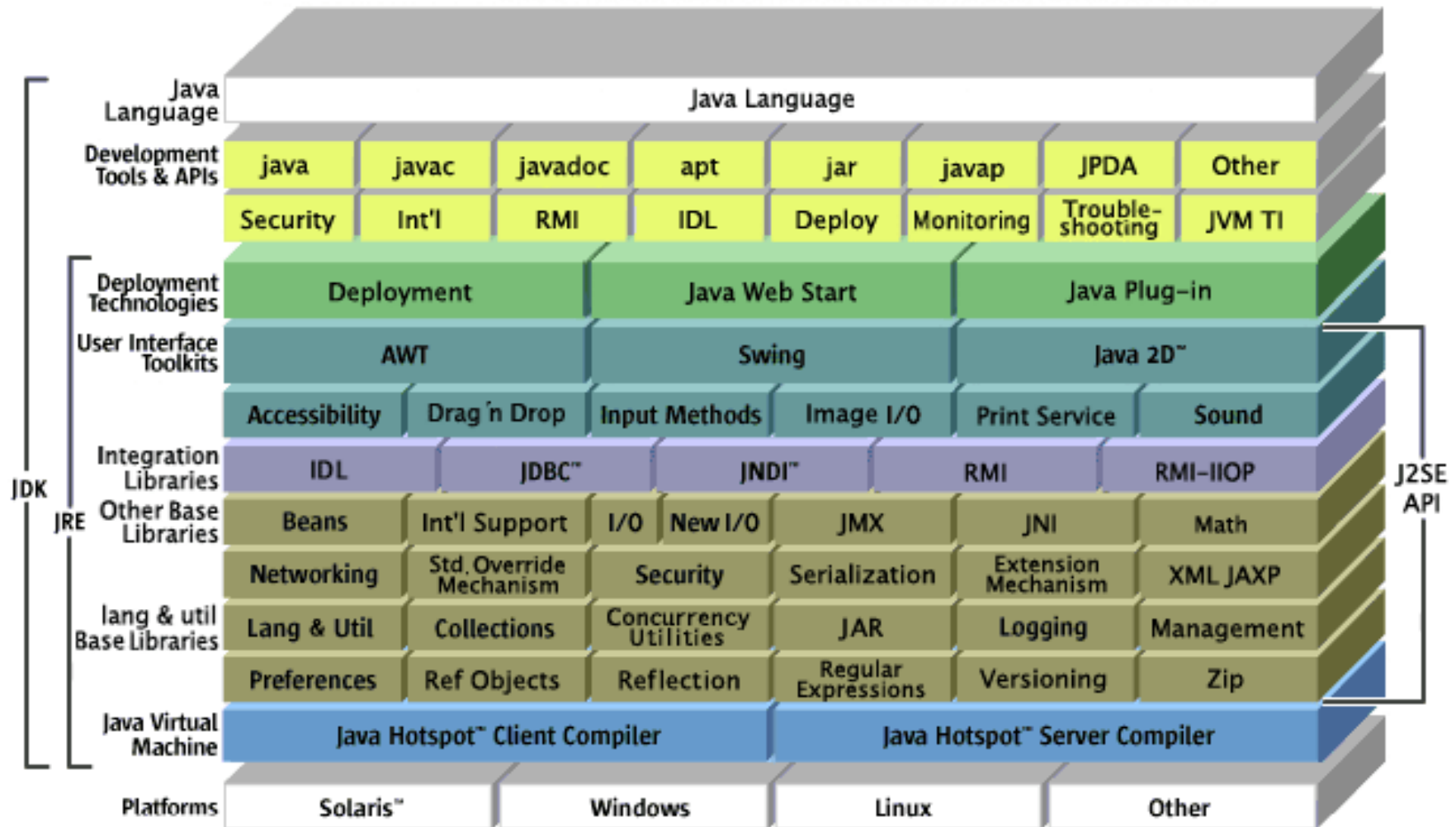


Java-**Programm** (Byte-Code) wird von der *virtuellen Maschine* (VM) auf dem jeweiligen **Betriebssystem** ausgeführt.

Für **jedes Betriebssystem** muss es daher **eine** speziell dafür geschriebene *virtuellen Maschine* geben.

1.3.1.3 Java – Konzept – Virtuelle Maschine

Java™ 2 Platform Standard Edition 5.0

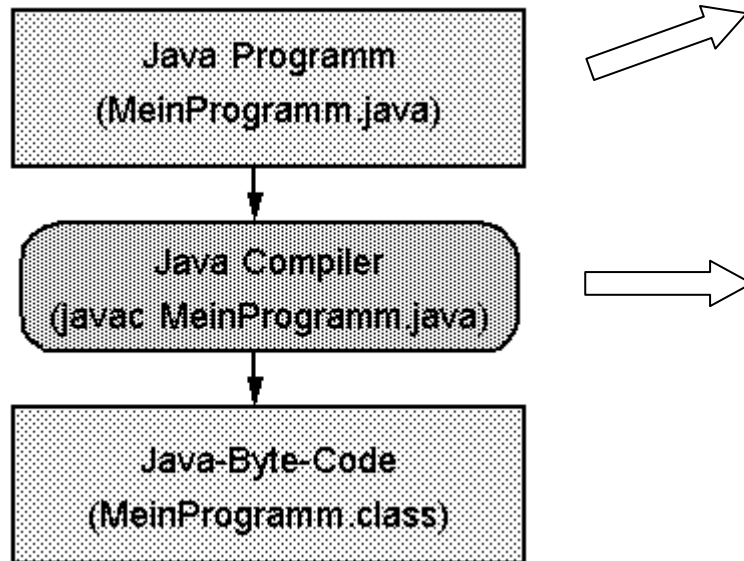


1.3.1.4 Arbeiten mit Java – Werkzeuge

Für das Arbeiten mit Java wird benötigt:

- Java Development Kit (**JDK**)
zum **Entwickeln** von Anwendungen
plus viele weitere Werkzeuge (Tools)
 - Java Runtime Environment (**JRE**)
zum **Ausführen** (Deployment) von Anwendungen.
(ist im JDK enthalten).
 - Download bei <http://java.sun.com/downloads>
als Java 2 Platform Standard Edition 5.0 (J2SE 5.0)
(derzeit als J2SE[®] Development Kit 5.0 Update 5)
 - Texteditor für den Quelltext
- ➔ alles kostenlos, also **los geht's ...**

1.3.1.4 unser erstes Programm ...



ausführen

```
public class Hello
{
    public static void main( String[] argv )
    {
        System.out.println( "HELLO" );
    }
}
```

```
C:\WINDOWS>cd C:\1_Hello
C:\1_Hello>javac Hello.java
C:\1_Hello>
```

```
C:\1_Hello>javac Hello.java
C:\1_Hello>java Hello
HELLO
C:\1_Hello>
```

1.3.1.4 Java

- Quelldatei muss wie jeweilige Klasse heißen
- Eine Datei pro Klasse
- Keine Trennung von Definition und Deklaration wie z.B. in C++
- Mindestens eine Klasse muss eine main-Funktion haben:
`public static void main(String[] argv)`
`{ ... }`
- Befehl zum **Compilieren**:
> `javac MeineKlasse.java`
- Befehl zum **Ausführen** (Aufruf der VM):
> `java MeineKlasse`

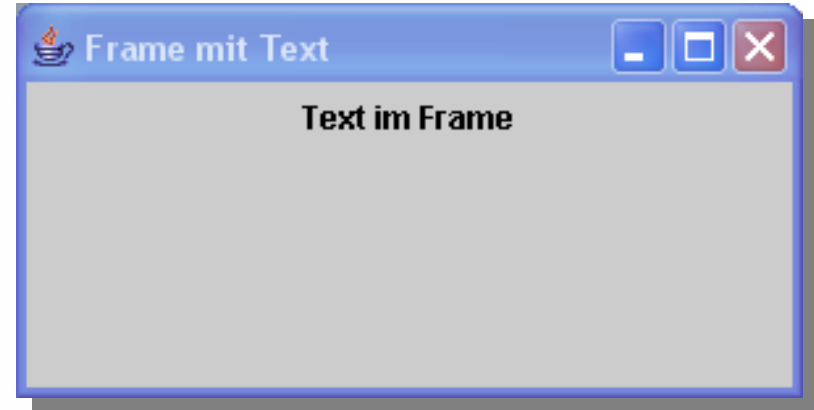
1.3.1.4 unser zweites Programm ...

```
import java.awt.*;
import javax.swing.*;

public class Hello2 extends JFrame
{
    Container c;
    JLabel beschrift;

    public Hello2()
    {
        c = getContentPane();
        c.setLayout( new FlowLayout() );
        beschrift = new JLabel( "Text im Frame" );
        c.add( beschrift );
    }

    public static void main( String[] argv )
    {
        Hello2 fenster = new Hello2();
        fenster.setTitle( "Frame mit Text" );
        fenster.setSize( 300, 150 );
        fenster.setVisible( true );
        fenster.setDefaultCloseOperation( JFrame.EXIT_ON_CLOSE );
    }
}
```



1.3.1.4 ... ein Applet ...

```
import java.awt.*;
import javax.swing.*;

public class HelloApplet extends JApplet
{
    Container c;
    JLabel beschrift;

    public void init()
    {
        c = getContentPane();
        c.setLayout( new FlowLayout() );
        beschrift = new JLabel( "Text im
        c.add( beschrift );
    }
}
```

```
<html>
  <head>
    <title>Applet mit Text</title>
  </head>
  <body>
    <b>Ein Applet mit Text</b>
    <hr></hr>
    <applet code = "HelloApplet.class" width="300" height="150">
      Hier sollte eigentlich ein Applet auftauchen und laufen
    </applet>
  </body>
</html>
```



1.3.1.4 Sicherheit bei Applets

- Applets sind meist unbekannter Code
- Bei Ausführung von Applets müssen Sicherheitsbeschränkungen eingehalten werden
- Ein Applet darf
 - nicht auf das lokale Dateisystem zugreifen - es sei denn der Benutzer läßt es ausdrücklich zu
 - keine Verzeichnisse ansehen, Existenz von Dateien prüfen
 - keine Dateiattribute (Größe, Zugriffsrechte) prüfen
 - keine Dateien lesen, schreiben, löschen, umbenennen
 - keine Netzwerkverbindungen zu irgendeinem Computer aufbauen
 - die Ausführung nicht über `exit()` beenden

1.3.1.4 Hausaufgabe – Vorbereitung n. Vorl.

- Machen Sie sich mit Java-Dokumentation und Literatur vertraut.
- Installieren Sie ein JDK
- Schreiben Sie einfache Programme auf Basis von Beispielen (Applets sind nicht so wichtig)
- Ziel:
 - Programmstruktur
 - Datentypen String, int, double, boolean, ...
 - Klassen und Vererbung
 - Struktur der Bibliotheken kennen lernen und verstehen.