9. Programmierrichtlinien

Programmierrichtlinien

Code-Konventionen, Style Guides

Anforderungen an die äußere Form von Programmen zur Erhöhung ihrer Lesbarkeit

- Nicht Teil der Sprachdefinition
  (keine Überprüfung durch den Compiler)
- Oft firmenintern
  - z. T. abweichend
  - trotzdem allgemein akzeptierte Prinzipien

Java-Programm: Probleme?

```
class S     {
   int x, y; P p, q;
   public void m()   {
      switch(s)   {
         case c1: d1();
         break;
         case c2:
         d2();   break;
         default: dA();
         break;   }   }
   }
```


Java-Programm: Was fällt auf?

```
class S{
   int x, y; P p, q;
   public void m()   {
      switch(s)   {
         case c1: d1();
         break;
         case c2:
         d2();   break;
         default: dA();
         break;   }   }
   }
```

```
class Figur     {
   // nur Ausschnitt
   int x, y;
   Point pLinks, pRechts;

   public void zeichneFigur()   {
      switch(typFigur)   {
         case LINE:
            drawLine();
            break;

         case CIRCLE:
            drawCircle();
            break;

         default:
            drawAll();
            break;
      }
   }
}
```

Einrückungen und Leerzeilen

```
class Figur     {
    int x, y;
    Point pLinks, pRechts;

    public void zeichneFigur()    {
        switch(typFigur)    {
            case LINE:
                drawLine();
                break;

            case CIRCLE:
                drawCircle();
                break;

            default:
                drawAll();
                break;
        }
    }
}
```


Namenswahl: aussagefähig

Variablen

- Beginn mit Kleinbuchstaben
- Teilworte beginnen groß
- kurz, aber mit Semantik, oft: Substativ enthalten
- temporäre Variablen (z. B. Laufvariablen):
  - Typ int: i, j, k, m, n
  - Typ char: c, d, ch

```
einKunde, actValue, farbe
```

Konstanten

- in Großbuchstaben
- Teilworte durch '_' getrennt

```
ROT, GRUEN, ROT_GRUEN, ROT_SCHWARZ
```


Namenswahl (2)

Methoden:

- Beginn mit Kleinbuchstaben
- Teilworte beginnen groß
- sollten Verben beinhalten

```
suchen, kundenEintragen
```

Klassen:

- Beginn mit Großbuchstaben
- Teilworte beginnen groß
- sollten Substantiv sein

```
Baum, Kunde, GeometrischeFigur,
  Keyboard, System
```


L a y o u t : Anordnung des Programms

(Text- und Bildgestaltung)

Ausdrücke:

```
help = a * b;     i++;
```

- Operatoren von Operanden durch Leerzeichen trennen, (außer: unäre Operatoren)

```
if (kundenNr > NULL)
```

- Leerzeichen innerhalb / außerhalb von Klammern

Layout (2)

Methodenaufruf, -deklaration

```
x = setValue(kundenNr, 25);
```

```
private static setValue(int kunde, String s) {
```


Layout (3)

Anweisungen, insb. Blöcke: `{ ... }`

C-Stil

```
while (a > b)   {
   ...
}
```

Pascal-Stil

```
while (a > b)
{
   ...
}
```


Layout (4)

If – Anweisung

```
if (a > 0) {
   ...
} else if (a < 0)   {
   ...
} else   {
   ...
}
```


Layout (5)

```
class Shapes   {
   int x, y;
   Point pFirst, pSecond;

   public void selectColor () {
     switch (selectedShape) {
       case LINE:
         drawLine();
         break;
   }
}
```

Leerzeile zwischen Variablen und Methoden

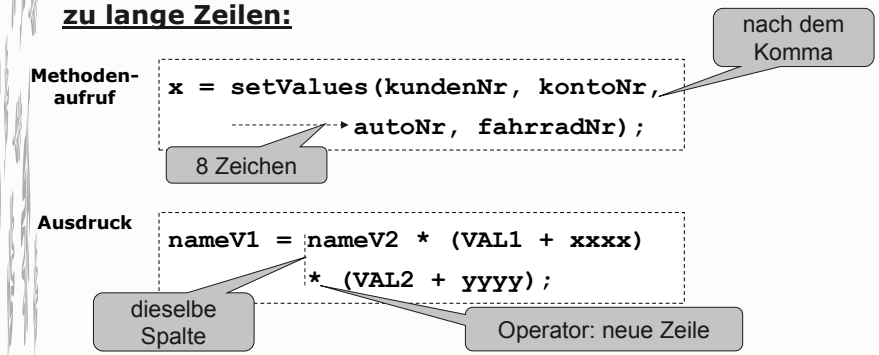
je Zeile höchstens eine Anweisung

Layout (6)

Einrückungstiefe: 2, 3, 4

Zeilenlänge: < 80 Zeichen

zu lange Zeilen:



Sonstiges

- Kommentierung: ausreichend und aussagefähig
- Eine Variablendeklaration pro Zeile
- Methoden durch eine Leerzeile voneinander getrennt
- In Methoden: lokale Variablen und Anweisungen durch Leerzeilen getrennt
- In Methoden: verschiedene logische Abschnitte einer Methode durch Leerzeilen getrennt
  (z.B. Eingabe, 1. Teilalgorithmus, … Ausgabe)

Programmierrichtlinien: Adressen

- **JAVA Coding Conventions von Oracle (SUN)**

http://www.oracle.com/technetwork/java/codeconvtoc-136057.html

- **JAVA Coding Standards von Ambysoft Inc**

http://www.ambysoft.com/essays/javaCodingStandards.html

Tools zur automatischen Überprüfung des Programmierstils

Tool 'AssessStyle': automatische Stilüberprüfung

Einbeziehung in das Praktikum in GdP (in den letzten Jahren)

automatische Bewertung der Praktikumsprogramme mit Punktabzügen

→ jetzt: Plugin in Eclipse zur Stilprüfung