



Aufgabenblatt 3

letzte Aktualisierung: 26. März, 12:55

Ausgabe: 04.04.2006

Thema: Eigene Klassen anlegen, instanzieren, Werte belegen und ausgeben, Methoden aufrufen

Lernziele:

- Eine Klasse als Struktur (Studenten-Klasse)
- new
- Arrays kennen lernen
- Methoden als neue Qualität für eine Struktur

1. Aufgabe: Eine Studentendatenbank

Im Rahmen dieser Aufgabe wollen wir eine kleine „Studentendatenbank“ erstellen.

Hinweis für Tutoren:

Die angegebene Beispiellösung arbeitet zwar mit extends, aber die Studenten sollen das natürlich nicht so machen. Ich habe nur diesen Weg gewählt um aufzuzeigen was sich von Aufgabe zu Aufgabe an der Klasse verändert! Die Studenten erweitern natürlich einfach ihre Student-Klasse.

- 1.1. Schreibe eine Klasse `Student`, die Informationen wie Name, Matrikelnummer etc. über einen Studenten speichert. Erzeuge ein Objekt dieser Klasse und belege die Objektvariablen mit Werten. Lese diese Werte aus den Objektvariablen wieder aus und gib sie auf dem Bildschirm aus.

Beispiellösung:

```
4 class Student1 {
5
6     String nachname;
7     String vorname;
8     int matrikelnummer;
9     String strasse;
10    int hausnummer;
11    String wohnort;
12    int plz;
13    String studiengang;
14
15 }

13 static void aufg1() {
14     printAufg("Beginn_Aufgabe_1");
15
16     Student1 klaus = new Student1();
```

```
17     klaus.nachname = "Mustermann";
18     klaus.vorname = "Klaus";
19     klaus.matrikelnummer = 424242;
20     klaus.plz = 10587;
21     klaus.wohnort = "Berlin";
22     klaus.hausnummer = 23;
23     klaus.strasse = "Straße_des_17._Juni";
24     klaus.studiengang = "Informatik";
25
26     System.out.println(klaus.vorname + " " + klaus.nachname);
27     System.out.println(klaus.strasse + " " + klaus.hausnummer);
28     System.out.println(klaus.plz + " " + klaus.wohnort);
29     System.out.println(klaus.matrikelnummer + "\t" + klaus.studiengang);
30
31     printAufg("Ende_Aufgabe_1");
32 }
```

- 1.2. Schreibe einen Konstruktor für deine Klasse, der die Objektvariablen des erzeugten Objekts mit Werten füllt. Erzeuge mit Hilfe dieses Konstruktors ein Objekt, lies seine Werte aus und gib sie auf dem Bildschirm aus.

Beispiellösung:

```
4 class Student2 extends Student1 {
5
6     Student2(String nachname, String vorname,
7         int matrikelnummer, String studiengang,
8         String strasse, int hausnummer,
9         String wohnort, int plz) {
10         this.nachname = nachname;
11         this.vorname = vorname;
12         this.matrikelnummer = matrikelnummer;
13         this.studiengang = studiengang;
14         this.strasse = strasse;
15         this.hausnummer = hausnummer;
16         this.wohnort = wohnort;
17         this.plz = plz;
18     }
19 }

34 static void aufg2() {
35     printAufg("Beginn_Aufgabe_2");
36
37     Student2 klaus = new Student2("Mustermann", "Klaus", 424242, "Informa
38         "Straße_des_17._Juni", 23, "Berlin", 10587);
39
40     System.out.println(klaus.vorname + " " + klaus.nachname);
41     System.out.println(klaus.strasse + " " + klaus.hausnummer);
42     System.out.println(klaus.plz + " " + klaus.wohnort);
43     System.out.println(klaus.matrikelnummer + "\t" + klaus.studiengang);
44
45     printAufg("Ende_Aufgabe_2");
46
47 }
```

- 1.3. Erweitere deine Klasse um eine Methode `printMe()`, die die Daten des Studentenobjekts auf dem Bildschirm ausgibt. Teste deine Methode!

Beispiellösung:

```
4 class Student3b extends Student2 {
15     Student3b(String nachname, String vorname, int matrikelnummer, S
16         String strasse, int hausnummer, String wohnort, int plz)
17         super(nachname, vorname, matrikelnummer, strasse, studiengan
18             plz);
19     }
20
21     void printMe() {
22         System.out.println(this.vorname + " " + this.nachname);
23         System.out.println(this.strasse + " " + this.hausnummer);
24         System.out.println(this.plz + " " + this.wohnort);
25         System.out.println(this.matrikelnummer + "\t" + this.studien
26     }
27 }
28 }

49     static void printMe(Student1 studi) {
50         System.out.println(studi.vorname + " " + studi.nachname);
51         System.out.println(studi.strasse + " " + studi.hausnummer);
52         System.out.println(studi.plz + " " + studi.wohnort);
53         System.out.println(studi.matrikelnummer + "\t" + studi.studi
54     }
55
56     static void aufg3() {
57         printAufg("Beginn_Aufgabe_3a");
58
59         Student2 klaus = new Student2("Mustermann", "Klaus", 424242,
60             "Straße_des_17._Juni", 23, "Berlin", 10587);
61
62         printMe(klaus);
63
64         printAufg("Ende_Aufgabe_3a");
65
66
67         printAufg("Beginn_Aufgabe_3b");
68
69         Student3b franz = new Student3b("Mustermann", "Franz", 424242,
70             "Straße_des_17._Juni", 23, "Berlin", 10587);
71
72         franz.printMe();
73
74         printAufg("Ende_Aufgabe_3b");
75
76         printAufg("Beginn_Aufgabe_3c");
77
78         Student3c hans = new Student3c("Mustermann", "Hans", 424242,
79             "Straße_des_17._Juni", 23, "Berlin", 10587);
80
81         System.out.println(hans);
82     }
```

```
83         printAufg("Ende_Aufgabe_3c");
84
85     }
```

- 1.4. Lege eine Feld (Array) mit 100 Studentenobjekten an, die verschiedene Matrikelnummern haben und gib alle auf dem Bildschirm aus.

Beispiellösung:

```
87     static void aufg4() {
88         printAufg("Beginn_Aufgabe_4");
89
90         Student3c[] studenten = new Student3c[100];
91
92         for (int i = 0; i < studenten.length ; i++) {
93             studenten[i] = new Student3c("Mustermann", "Hans", 40000+i , "Inf
94                 "Straße_des_17._Juni", 23, "Berlin", 10587);
95         }
96
97
98         for (int i = 0; i < studenten.length ; i++) {
99             System.out.println(studenten[i]);
100         }
101
102         printAufg("Ende_Aufgabe_4");
103     }
```

- 1.5. Das in der vorherigen Aufgabe angelegte Feld ist schon eine Art Studentendatenbank. Schreibe daher eine neue Klasse `StudentDatabase`, die eine Feld von Studentenobjekten enthält. Die Klasse soll einen Konstruktor haben, dem die Größe des zu erzeugenden Feldes übergeben wird. Implementiere auch eine Methode `printMe()`, die alle Datensätze ausgeben soll, sowie eine Methode `addStudent()`, die einen Studentenobjekt mit den zu übergebenen Daten anlegen soll und dieses in die Datenbank einfügen soll.

Hinweis für Tutoren:

Achtung! Methode `addStudent` ist nicht genau spezifiziert!

Beispiellösung:

```
4 class StudentenDatenbank {
5
6     Student3c[] studenten;
7
8     int aktuell;
9
10    /**
11     *
12     */
13    StudentenDatenbank(int anzahl) {
14        if (anzahl > 0)
15            studenten = new Student3c[anzahl];
16        aktuell = 0;
17    }
18
19    boolean studentenHinzufuegen(String nachname, String vorname,
```

```

20         int matrikelnummer, String studiengang, String strass
21         int hausnummer, String wohnort, int plz) {
22
23         if (aktuell >= studenten.length)
24             return false;
25
26         studenten[aktuell] = new Student3c(nachname, vorname, mat
27             studiengang, strasse, hausnummer, wohnort, plz);
28         aktuell++;
29
30         return true;
31     }
32
33     void printMe() {
34         for (int i = 0 ; i < aktuell && i < studenten.length; i++)
35             System.out.println(studenten[i]);
36     }
37 }
38
39 String toString() {
40     String ret = "";
41     for (int i = 0 ; i < aktuell && i < studenten.length; i++)
42         ret += studenten[i] + "\n";
43 }
44 return ret;
45 }
46
47 }

```

```

105     static void aufg5() {
106         printAufg("Beginn_Aufgabe_5");
107         StudentenDatenbank datenbank = new StudentenDatenbank(100
108
109         for (int i = 0; i < 100 ; i++) {
110             datenbank.studentenHinzufuegen("Mustermann", "Hans",
111                 "Straße_des_17._Juni", 23, "Berlin", 10587);
112         }
113
114         System.out.println(datenbank);
115         printAufg("Ende_Aufgabe_5");
116     }

```

- 1.6. Zusatzaufgabe:** Erweitere die Studentendatenbank um eine Methode `deleteStudent()`, die das Studentenobjekt mit der übergebenen Matrikelnummer aus der Datenbank löschen soll. Funktionert nach dem Löschen einiger Objekte deine Ausgabemethode (`printMe()`) noch?
- 1.7. Zusatzaufgabe:** Erweitere die Studentendatenbank um eine Methode `numberOfStudents()`, die die Anzahl der in der Datenbank gespeicherten Studentenobjekte zurückliefert.
- 1.8. Zusatzaufgabe:** Schreibe Methoden, die statistische Daten über die Studenten ermitteln. Beispiele sind: die Durchschnittsnote, das Durchschnittsalter ...