

Vorbereitung auf das Programmierdiktat

Michael Strassberger
michael.strassberger@uni-hamburg.de
3strassb@informatik.uni-hamburg.de

April 19, 2017

- 1 Syntax einer Klasse
- 2 Datenstrukturen
- 3 Kontrollstrukturen
- 4 Fragen

Klassen und Interfaces definieren (Begriffe Rumpf und Kopf sollten auch bekannt sein)

Klassen und Interfaces definieren (Begriffe Rumpf und Kopf sollten auch bekannt sein)

```
class Buch // Kopf
{
    // Rumpf
}
```

Klassen und Interfaces definieren (Begriffe Rumpf und Kopf sollten auch bekannt sein)

```
class Buch // Kopf
{
    // Rumpf
}
```

```
interface Medium // Kopf
{
    // Rumpf
}
```

Implementationsbeziehungen zwischen einer Klasse und einem Interface definieren

Implementationsbeziehungen zwischen einer Klasse und einem Interface definieren

```
class Buch implements Medium  
{  
  
}
```

Variablen deklarieren, initialisieren, zuweisen

Variablen deklarieren, initialisieren, zuweisen

```
int nummer;
```

Variablen deklarieren, initialisieren, zuweisen

```
int nummer;
```

```
int nummer = 42;
```

Variablen deklarieren, initialisieren, zuweisen

```
int nummer;
```

```
int nummer = 42;
```

```
    nummer = 43;
```

(Zustands-) Felder (alias Exemplarvariablen) deklarieren

```
class Buch implements Medium  
{
```

(Zustands-) Felder (alias Exemplarvariablen) deklarieren

```
class Buch implements Medium
{
    int _seiten;
}
```

Zugriffsmodifikatoren verwenden können (public/private)

```
class Buch implements Medium  
{
```

Zugriffsmodifikatoren verwenden können (public/private)

```
class Buch implements Medium
{
    private int _seiten;
}
```

Konstruktor definieren

```
class Buch implements Medium  
{  
    private int _seiten;  
}
```


Konstruktor definieren

```
class Buch implements Medium
{
    private int _seiten;

    public Buch()
    {
        _seiten = 42;
    }
}
```

Werte von Methoden zurckgeben lassen (Schlsselwort return)

```
class Buch implements Medium
{
    private int _seiten;
```

Werte von Methoden zurckgeben lassen (Schlsselwort return)

```
class Buch implements Medium
{
    private int _seiten;

    public int gibSeitenAnzahl()
    {
        return _seiten;
    }
}
```

```
class Buch implements Medium  
{
```

Klassenvariablen deklarieren, initialisieren

```
class Buch implements Medium
{
    private static int anzahlBuecher;
```

Klassenvariablen deklarieren, initialisieren

```
class Buch implements Medium
{
    private static int anzahlBuecher;

    static
    {
        anzahlBuecher = 0;
    }
}
```

Private und öffentliche Methoden/Operationen definieren,
sowohl als Exemplar- als auch als Klassenmethoden
(Begriffe Rumpf und Kopf sollten auch bekannt sein)

```
public int gibSeitenAnzahl()  
{  
    return _seiten;  
}
```

Private und öffentliche Methoden/Operationen definieren,
sowohl als Exemplar- als auch als Klassenmethoden
(Begriffe Rumpf und Kopf sollten auch bekannt sein)

```
public int gibSeitenAnzahl()  
{  
    return _seiten;  
}
```

```
private int gibSeitenAnzahlGeheim()  
{  
    return _seiten;  
}
```



```
class Buch implements Medium
{
    private static int anzahlBuecher;
    static
    {
        anzahlBuecher = 0;
    }
}
```

```
class Buch implements Medium
{
    private static int anzahlBuecher;
    static
    {
        anzahlBuecher = 0;
    }

    public static int gibAnzahlBuecher()
    {
        return anzahlBuecher;
    }
}
```

Recap: Rumpf von Klassen und Methoden

```
class Buch implements Medium // Klassen Kopf
{
    // Klassen Rumpf

    public Buch() // Methoden Kopf
    {
        // Methoden Rumpf
    }
}
```

Parameter deklarieren und verwenden

```
class Buch implements Medium  
{  
    private int _seiten;  
}
```

```
class Buch implements Medium
{
    private int _seiten;

    public void setzeSeitenAnzahl(int seiten)
    {
```

Parameter deklarieren und verwenden

```
class Buch implements Medium
{
    private int _seiten;

    public void setzeSeitenAnzahl(int seiten)
    {
        _seiten = seiten;
    }
}
```

```
public class SE2Tutorium  
{
```

```
public class SE2Tutorium
{
    public static void main(String [] args)
    {
    }
}
```



```
public class SE2Tutorium
{
    public static void main(String [] args)
    {
```

Objekte erzeugen

```
public class SE2Tutorium
{
    public static void main(String [] args)
    {
        Buch bluej = new Buch();
    }
}
```

Umgang mit Referenzvariablen und der Punktnotation

```
public class SE2Tutorium
{
    public static void main(String [] args)
    {
        Buch bluej = new Buch();
    }
}
```

Umgang mit Referenzvariablen und der Punktnotation

```
public class SE2Tutorium
{
    public static void main(String [] args)
    {
        Buch bluej = new Buch();

        int bluejSeiten = bluej.getSeitenAnzahl();
    }
}
```

Umgang mit Referenzvariablen und der Punktnotation

```
public class SE2Tutorium
{
    public static void main(String [] args)
    {
        Buch bluej = new Buch();

        int bluejSeiten = bluej.getSeitenAnzahl();

        bluej.setzeSeitenAnzahl(bluejSeiten + 1);
    }
}
```

Noch Fragen zu Klassen und Methoden Rümpfen

Umgang mit Basisdatentypen

- (1) byte
- (2) short
- (3) int
- (4) long

Umgang mit Basisdatentypen

- (1) byte
- (2) short
- (3) int
- (4) long
- (5) float
- (6) double

Umgang mit Basisdatentypen

- (1) byte
- (2) short
- (3) int
- (4) long
- (5) float
- (6) double
- (7) char

Umgang mit Arrays (deklarieren, initialisieren, schreibender und lesender Zugriff)

Umgang mit Arrays (deklarieren, initialisieren, schreibender und lesender Zugriff)

```
int [] zahlenkette
```

Umgang mit Arrays (deklarieren, initialisieren, schreibender und lesender Zugriff)

```
int [] zahlenkette
```

```
zahlenkette = new int [42]
```

Umgang mit Listen und Mengen

```
List<Integer> zahlenListe;
```

Umgang mit Listen und Mengen

```
List<Integer> zahlenListe;  
zahlenListe = new ArrayList();
```

```
List<Integer> zahlenListe;  
zahlenListe = new ArrayList();  
zahlenListe.add(100);
```


Umgang mit Listen und Mengen

```
List<Integer> zahlenListe;  
zahlenListe = new ArrayList();  
zahlenListe.add(100);  
zahlenListe.remove(0);
```

Switch-Anweisungen schreiben

Switch-Anweisungen schreiben

```
switch (zahl) {  
    case 10:  
        // do things  
        break;  
    default :  
        break  
}
```

Gibt es noch Fragen?

Gibt es noch Fragen?
Dann viel Erfolg beim Diktat!