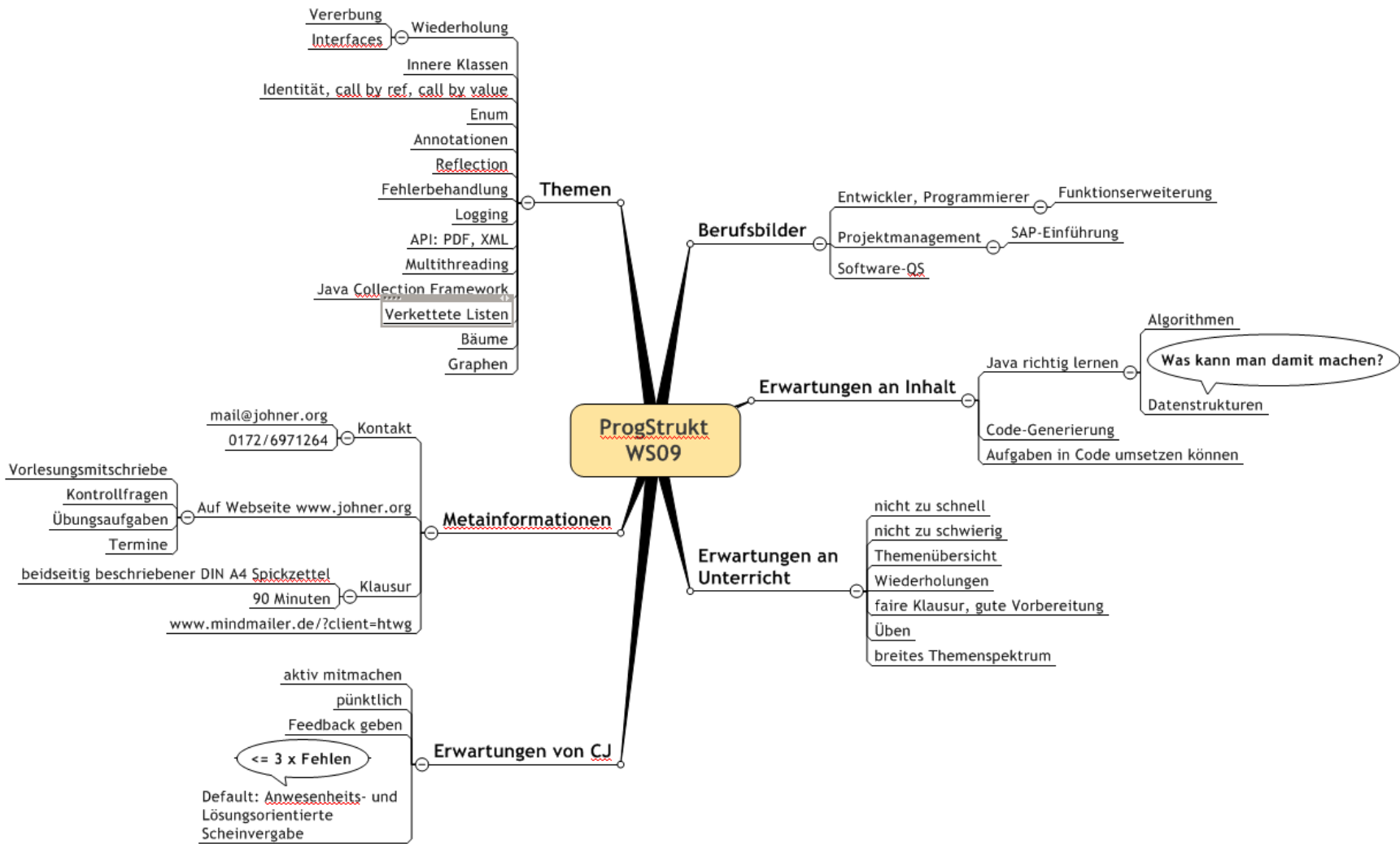


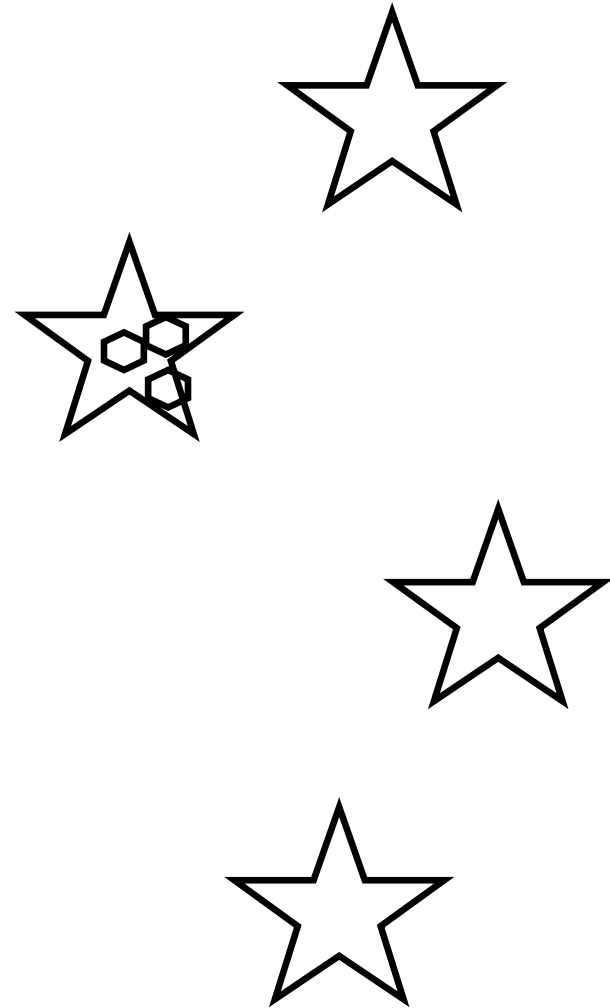
ProgStrukt



Aufbau von Java-Klassen

- Package
- Imports
- Klassenname
- Konstruktoren
- Attribute
- Methoden
 - Blöcke (Klammern)
 - Anweisungen, Kontrollstrukturen
- Kommentare
- innere Klassen

heutiges Thema



Innere Klassen

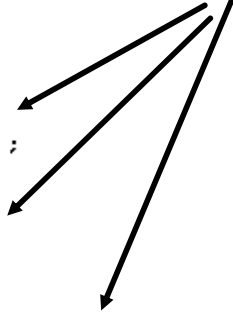
1. Typ Mitgliedsklassen (member class)

```
package innereklassen;  
  
public class AuessereKlasse {  
    public void sagHallo(){  
        System.out.println("Hallo von der Äußeren");  
    }  
  
    public class InnereKlasse {  
        public void sagHallo(){  
            System.out.println("Hallo von der Inneren");  
        }  
  
        public class GanzInnere {  
            public void sagHallo(){  
                System.out.println("Hallo von der GanzInneren");  
            }  
        }  
    }  
}
```

Innere Klassen können hierarchisch sein
(siehe links)

Instanziierung innerer Klassen

```
public static void main(String[] args) {  
    AuessereKlasse auessere = new AuessereKlasse();  
    auessere.sagHallo();  
  
    InnereKlasse innere = auessere.new InnereKlasse();  
    innere.sagHallo();  
  
    GanzInnere ganzinnen = innere.new GanzInnere();  
    ganzinnen.sagHallo();  
  
    new AuessereKlasse().new InnereKlasse().new GanzInnere().sagHallo();  
}
```



2. Lokale Klassen

```
package innereklassen;

public class AndereAuessere {
    private int i = 17;

    public void sagHallo(){
        final int j = 18;
        System.out.println("Hallo von der Anderen Äußeren");

        class LokaleInnereKlasse {
            public void sagHallo(){
                System.out.println("Hallo von der Inneren" + i);
                System.out.println("Hallo von der Inneren" + j);
            }
        }

        new LokaleInnereKlasse().sagHallo();
    }
}
```

Lokale Klassen werden innerhalb (!) von Methoden deklariert und dort verwendet.

Lokale Klassen haben keine Modifier (public, private)

Lokale Klassen können nur finale Variablen der umgebenden Methode nutzen.

3. Anonyme innere Klassen

```
package innereklassen;  
  
public class Werkstatt {  
    public static void main(String[] args) {  
        Auto auto = new Auto();  
        auto.setMotor(new IMotor() {  
            public int getPs() {  
                return 42;  
            }  
  
            public void setPs(int ps) {  
            }  
        });  
    }  
}
```

Anonyme innere Klasse
(muss ein Interface
implementieren)