

Informatik 10 Übungen zum Thema Vererbung und Polymorphie

1. Bring die Code-Stücke in die richtige Reihenfolge

1	<code>public RENNKAMEL(int g) {</code>
2	<code>private int geschw;</code>
3	<code>extends KAMEL{</code>
4	<code> this.geschw = g; }</code>
5	<code>public int nenneGeschw() { return this.geschw; }</code>
6	<code>super();</code>
7	<code>public class RENNKAMEL</code>

2. Fülle die Lücken aus

```
public class WUESTE extends SPIEL{  
// erbt von SPIEL|  
    // KAMEL-Array  
    protected  kamele;  
/**  
 * Konstruktor der Klasse Wueste  
 */  
public WUESTE()  
{  
    // Super-Konstruktor - 800 x 600, ohne Punkte und Maus  
    super(800, 600, false, false, false);  
    // Hintergrundgrafik: wueste.jpg  
    this.hintergrundgrafikSetzen("wueste.jpg");  
    // KAMEL-Array initialisieren  
    kamele =   
    // kamele[0] : "Normali"  
    // kamele[1] : "Renni"  
    // kamele[2] : "Tragi"  
    kamele[0]=new KAMEL(10,500, "Normali");  
    kamele[1]=new RENNKAMEL(10,500, "Renni",65);  
    kamele[2]=new TRANSPORTKAMEL(10,500, "Tragi",100);  
  
    // Schleife ueber alle Kamele:  
    // information ausgeben  
      
    .informationAusgeben();  
}
```

3. Kreuze alle richtigen Aussagen an

Was trifft für den Aufruf von 'super(...)' zu?

- a. Die Klasse weiß nun, dass sie erbt.
- b. Es wird ein Objekt der Superklasse erzeugt.
- c. Es wird der Code des entsprechenden Konstruktors der Superklasse abgearbeitet.
- d. 'super()' kann man zu jeder Zeit aufrufen.
- e. Das muss der erste Aufruf im Konstruktor der Subklasse sein.

Was bedeutet Vererbung?

- a. Man kann von mehreren Klassen gleichzeitig alle Attribute und Methoden übernehmen.
- b. Man kann eine andere Klasse um Attribute oder Methoden ergänzen, ohne in deren Code arbeiten zu müssen.
- c. Man kann alle Attribute von einer anderen Klasse übernehmen (aber nicht direkt verändern). Man übernimmt auch alle Methoden, die nicht 'private' sind.
- d. Man kann alle Attribute und Methoden von anderen Klassen übernehmen.

4.

```
public class HUND{
    private int alter;
    private String name;
    public HUND(){
        alter = 0;
        name = "Waldi";
    }

    public void setzeAlter(int alterNeu){
        alter = alterNeu;
    }

    public int nenneAlter(){
        return alter;
    }

    public void gibLaut(){
        System.out.print("wau ");
    }
}
```

```
public class DACKEL extends HUND{
    public DACKEL(){
        super();
    }

    @Override
    public void gibLaut(){
        System.out.print("kläff ");
    }
}
```

```
public class TESTER{
    HUND h1;
    DACKEL d1;
    public TESTER(){
        h1 = new DACKEL();
        d1 = new DACKEL();
        d1.alter = 5;
    }

    public void test(){
        h1.gibLaut();
    }
}
```

- a) Beim Übersetzen der Klassen wird eine Fehlermeldung angezeigt. Erkläre den Fehler und korrigiere ihn.
- b) Welche Ausgabe liefert in der korrigierten Version der Aufruf der Methode test() ? Begründe deine Antwort.

5.

Die stark vereinfachte Modellierung eines italienischen Restaurants hat drei Klassen:

PIZZA(preis, titel, belag, vegetarisch) , GERICHT(preis, titel) und PASTA(preis, titel, sosse).

In den Klammern stehen die Namen der Attribute. Finde jeweils einen sinnvollen Datentyp und bring die Klassen in eine sinnvolle Vererbungshierarchie.

Alle Klassen sollen auch auf eine Methode informationAusgeben() zugreifen können, die in einem Textfenster die Namen und Werte der Attribute ausgibt.

Die Attribute brauchen in dieser Übungsaufgabe nicht auf private oder protected gesetzt werden.