

FRAGE 1 VON 10

Erreichbare Punkte: 3,00

Welche Eigenschaft trifft auf eine Unterklasse zu?

Wählen Sie eine Antwort:

- ☐ Eine Unterklasse erbt alle Methoden und Attribute einer Oberklasse und kann zusätzlich weitere Methoden und Attribute besitzen.
- ☐ Die Unterklasse enthält lediglich alle Attribute der Oberklasse. Es ist dazu möglich, dass sie eigene Methoden implementiert.
- ☐ In Java kann die Unterklasse Attribute und Methoden aller Oberklassen haben. Zusätzliche, d.h. eigene Methoden, sind nicht möglich.
- ☐ Methoden der Oberklasse können in der Unterklasse überladen werden.

Die richtige Antwort lautet: Eine Unterklasse erbt alle Methoden und Attribute einer Oberklasse und kann zusätzlich weitere Methoden und Attribute besitzen.

MUSTERLÖSUNG

FRAGE 2 VON 10

Erreichbare Punkte: 3,00

Was sind Getter- und Setter-Methoden?

Wählen Sie eine Antwort:

- ☐ Getter- und Setter-Methoden sind spezielle Methoden, die zum Lesen und Ändern von Attributen einer Klasse eingesetzt werden.
- ☐ Getter-Methoden setzen den Wert und Setter-Methoden lesen die Werte der Attribute einer Klasse.
- ☐ Getter-Methoden berechnen und geben das Ergebnis zurück. Setter-Methoden setzen Werte und haben keinen Rückgabewert.
- ☐ Getter- und Setter-Methoden dienen ausschließlich der internen Verwendung: Nur die Konstruktoren einer Klasse dürfen diese Methoden nutzen, um Werte in den Attributen zu setzen.

Die richtige Antwort lautet: Getter- und Setter-Methoden sind spezielle Methoden, die zum Lesen und Ändern von Attributen einer Klasse eingesetzt werden.

MUSTERLÖSUNG

FRAGE 3 VON 10

Erreichbare Punkte: 3,00

Was passiert bei folgendem Konstruktor-Quellcode:

```
public Haus(){
    this.fertiggestellt = true;

    public Haus(float groesse, int etagen){
        this.etagen = etagen;
        this();
        this.groesse = groesse;
    }
}
```

Wählen Sie eine Antwort:

- ☐ Der zweite Konstruktor kann nicht verwendet werden, und es gibt zur Laufzeit einen Fehler, weil der Aufruf des Standard-Konstruktors zwischen den Wertezuweisungen steht.
- ☐ Nach dem Aufruf des Standard-Konstruktors wird der zweite Attributwert für "groesse" nicht mehr gesetzt.
- ☐ Der Compiler gibt einen Fehler aus und kann den Quellcode nicht compilieren.
- ☐ Beim Aufruf des 2. Konstruktors wird der Standard-Konstruktor mit genutzt.

Die richtige Antwort lautet: Der Compiler gibt einen Fehler aus und kann den Quellcode nicht compilieren.

MUSTERLÖSUNG

FRAGE 4 VON 10

Erreichbare Punkte: 3,00

Was muss beim Implementieren von statischen Methoden beachtet werden?

Wählen Sie eine Antwort:

- ☐ In statischen Methoden dürfen keine Konstruktoren aufgerufen werden.
- ☐ In statische Methoden dürfen keine Klassenvariablen genutzt werden.
- ☐ In statischen Methoden kann nicht frei auf Instanzvariablen der Klasse zugegriffen werden.
- ☐ In statischen Methoden dürfen keine switch-Anweisungen oder Schleifen-Konstrukte verwendet werden.

Die richtige Antwort lautet: In statischen Methoden kann nicht frei auf Instanzvariablen der Klasse zugegriffen werden.

MUSTERLÖSUNG

FRAGE 5 VON 10

Erreichbare Punkte: 3,00

Folgende Aufgabenstellung sei gegeben:

In einer Tierarztpraxis werden die Haustiere von Patientenbesitzern behandelt. Zu jedem Patientenbesitzer werden seine Tiere mit Namen, Geburtsdatum, Gewicht und Farbe aufgenommen. Die Tierärztin dokumentiert die Behandlung in einem Krankenblatt und notiert die je Behandlung gegebenen Medikamente.

Welches Attribut macht für eine Klasse "Tier" **keinen** Sinn?

Wählen Sie eine Antwort:

- ☐ Gewicht
- ☐ Farbe
- ☐ Geburtsdatum
- ☐ Tierärztin

Die richtige Antwort lautet: Tierärztin

FRAGE 6 VON 10

Erreichbare Punkte: 3,00

Was ist in folgendem Code für die Konstruktoren falsch?

```
public class Buch {  
  
    private String titel;  
    private Autor autor;  
    private int anzahlSeiten;  
    private float preis;  
  
    //Konstruktor 1
```

MUSTERLÖSUNG

```
public Buch(){  
}  
  
//Konstruktor 2  
public Buch(String titel){  
    this.titel = titel;  
}  
  
//Konstruktor 3  
public Buch(String titel,int anzSeiten){  
    this.titel = titel;  
    this.anzahlSeiten = anzSeiten;  
}  
  
//Konstruktor 4  
public Buch(int anzSeiten, String titel){  
    this.anzahlSeiten = anzSeiten;  
    this(titel);  
}  
}
```

Wählen Sie eine Antwort:

- ☐ Konstruktor 3 und 4 haben die gleiche Parameterliste. Dies ist nicht erlaubt.
- ☐ Es gibt keinen Fehler bei diesem Code-Beispiel.
- ☐ Konstruktor 1 ist der Standard-Konstruktor und wird deshalb ohne Inhalt nicht akzeptiert.
- ☐ In Konstruktor 4 muss der erste Aufruf der Aufruf eines anderen Konstruktors sein.

Die richtige Antwort lautet: In Konstruktor 4 muss der erste Aufruf der Aufruf eines anderen Konstruktors sein.

MUSTERLÖSUNG

FRAGE 7 VON 10

Erreichbare Punkte: 3,00

Was sind unchecked Exceptions?

Wählen Sie eine Antwort:

- ☐ Exceptions, deren Konstruktor überladen ist.
- ☐ Exceptions, die nicht von der Klasse Exception abgeleitet werden.
- ☐ Alle selbst definierten Exceptions.
- ☐ Alle Exceptions, die von der Klasse Runtime-Exception abgeleitet werden.

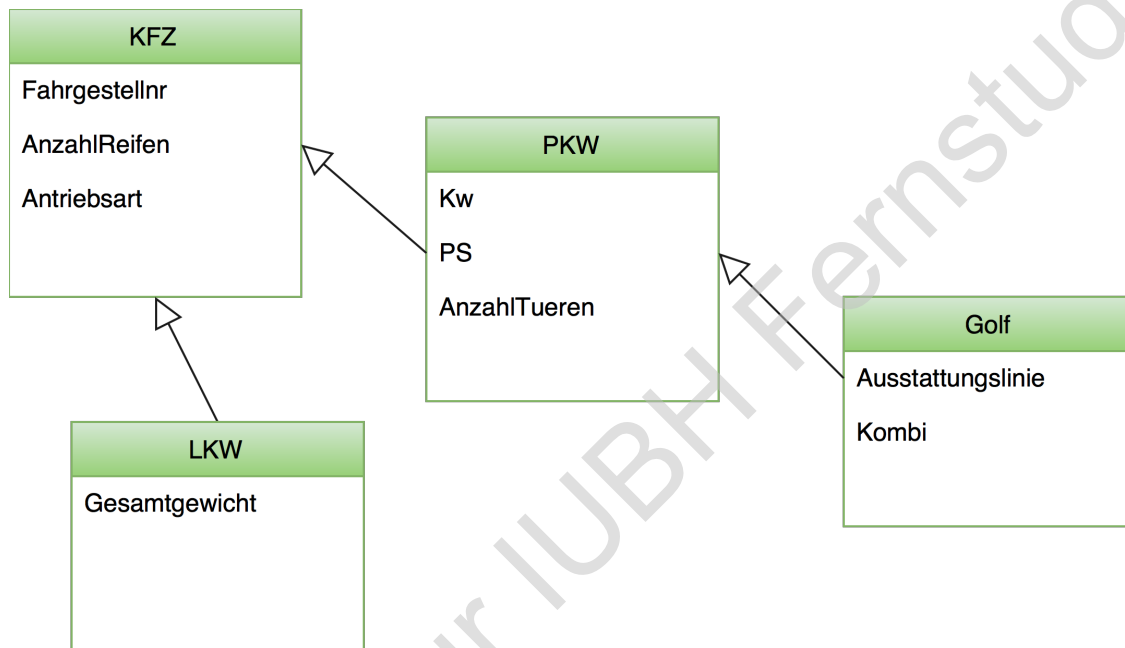
Die richtige Antwort lautet: Alle Exceptions, die von der Klasse Runtime-Exception abgeleitet werden.

MUSTERLÖSUNG

FRAGE 8 VON 10

Erreichbare Punkte: 6,00

Im Bild ist eine Vererbungshierarchie dargestellt.



Geben Sie an, welche Attribute die Klasse *Golf* besitzt.

- FahrgestellNr
- AnzahlReifen
- Antriebsart
- KW
- PS
- AnzahlTueren
- Ausstattungslinie
- Kombi

MUSTERLÖSUNG

FRAGE 9 VON 10

Erreichbare Punkte: 8,00

Kann es mehr als einen Kopierkonstruktor für eine Klasse geben?
Begründen Sie Ihre Antwort.

Nein, da ein Kopierkonstruktor immer ein Objekt desselben Typs übergeben bekommt, ist die Parameterliste eindeutig. Da es mehrere Konstruktoren nur geben kann, wenn die Parameterliste sich unterscheidet, kann es demnach nur einen Kopierkonstruktor geben.

FRAGE 10 VON 10

Erreichbare Punkte: 10,00

Warum ist folgende Anweisung richtig?

```
java.lang.Object obj;  
obj = new Tier();
```

Aufgrund der Polymorphie kann jedes Object einer Unterklasse die Gestalt seiner Oberklasse annehmen. „Object“ ist die Oberklasse aller Klasse, also auch von Tier. Deshalb funktioniert diese Anweisung.