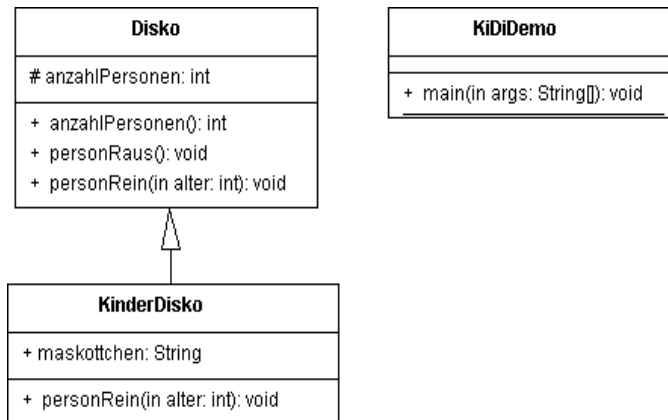


Gegeben ist folgendes UML Klassendiagramm. Realisieren Sie es in Java!



Die Methode `personRaus()` „entfernt“ eine Person aus der Disko. Es ist zu überprüfen, ob dies möglich ist.

Die Methode `personRein()` prüft, ob die Person in die Disko darf (Kinderdisko ab 6, Disko ab 18 Jahre) und gibt eine entsprechende Meldung aus.

Das Attribut `anzahlPersonen` speichert die Anzahl der Personen, die sich in der Disko aktuell aufhalten.

Jede Kinderdisko verfügt über ein `maskottchen`, dessen Name gespeichert wird.

Aufgaben:

a)

Erzeugen Sie ein Objekt `twister` der Klasse `Disko` und ein Objekt `flohzirkus` der Klasse `KinderDisko`.

Kann man aus dem Hauptprogramm direkt auf `anzahlPersonen` zugreifen?

Was passiert, wenn die Sichtbarkeit auf `private`, `public`, `paketsichtbar` geändert wird? Testen Sie und fügen Sie die Antwort als Kommentar in den Sourcecode ein!

b)

Mit dem Befehl `instanceof` kann zur Laufzeit überprüft werden, ob ein Objekt vom Typ einer bestimmten Klasse ist.


Beispiel:

```
boolean b;
```

```
String str = "Toll";
```

```
b = ( str instanceof String ); // wahr
```

Prüfen Sie, ob die Objekte `twister` bzw. `flohzirkus` vom Typ `Disko` bzw. `KinderDisko` sind und geben Sie entsprechende Meldungen am Bildschirm aus. Prüfen Sie weiters, ob die beiden Objekte vom Typ `Object` sind.

ARBEITSAUFTRAG	
Fach: SEW2	
Thema: JAVA – Vererbung, Pakete	

c) Fügen Sie nun folgendes Codestück in Ihr Testprogramm ein:

```
KinderDisco saloon = new KinderDisco();
Disco d = saloon;           // d ist eine Referenz auf saloon
```

Testen Sie, ob d vom Typ Disco bzw. Kinderdisco ist. Geben Sie das Ergebnis der Prüfung am Bildschirm aus!

Versuchen Sie, über das Objekt d auf die Attribute anzahlPersonen bzw. maskottchen zuzugreifen.

Funktioniert der Zugriff?

Warum hat der Zugriff funktioniert/nicht funktioniert? Was vermuten Sie? Notieren Sie Ihre Gedanken als Kommentar!

d)

Die Konstruktoren sind zu überladen. Die Anzahl der Personen, welche sich in der Disco befindet soll als Parameter übergeben werden können.

Standardmäßig befindet sich eine Person (der DJ) in der jeder Disco.

Konstruktoren werden nicht automatisch wie andere Methoden vererbt!

Jeder Konstruktor ruft, sofern nicht anders angegeben automatisch den Standardkonstruktor der Basisklasse auf.

Man kann im Konstruktor durch Aufruf von super(Parameterliste); einen bestimmten Konstruktor aufrufen.

super() muss dabei der erste Aufruf im Konstruktor sein!

Erweitern Sie Ihr Programm dahingehend, dass die Konstruktoren in KinderDisco explizit die Konstruktoren der Basisklasse (auch Superklasse genannt) aufrufen!

Testen Sie den Aufruf des neuen Konstruktors im Testprogramm!