

# Lösung PS Test 1 2021

## ▼ Lösung

Eine Klasse, welche im Default-Package liegt, kann nicht importiert werden.

- ☒ a. Wahr
- ☐ b. Falsch

## ▼ Lösung

Kreuzen Sie alle wahren Aussagen an!

- ☐ a. Beim Casten von `int` zu `char` werden die niederwertigsten 8 Bits berücksichtigt.
- ☐ b. `boolean` kann implizit zu `int` gecastet werden.
- ☐ c. `(int)"A"` gibt 65 zurück. (Hinweis: 65 ist der ASCII-Code von A)
- ☒ d. Beim Casten von `int` zu `short` werden die niederwertigsten 16 Bits berücksichtigt.

## ▼ Lösung

In einer abstrakten Klasse kann kein Konstruktor bereitgestellt werden, weil eine abstrakte Klasse nicht direkt instanziiert werden kann.

- ☐ a. Wahr
- ☒ b. Falsch

## ▼ Lösung

Kreuzen Sie alle Keywords an, welche in Java als Zugriffsmodifikatoren verwendet werden können!

- ☒ a. `protected`
- ☐ b. `default`
- ☐ c. `void`
- ☐ d. `static`
- ☐ e. `final`

## ▼ Lösung

Gegeben sei `Integer i = Integer.valueOf(23);`. Autoboxing ist dafür verantwortlich, dass der Vergleich `i == 23` den Wert `true` ergibt.

- ☐ a. Wahr
- ☒ b. Falsch

## ▼ Lösung

Kreuzen Sie alle wahren Aussagen an!

- ☐ a. Ein Konstruktor muss nicht den gleichen Namen wie die Klasse haben.
- ☒ b. Im statischen Initialisierer können Klassenvariablen initialisiert werden.
- ☐ c. Wird in einer konkreten Klasse kein expliziter Konstruktor definiert, so können keine Instanzen davon erzeugt werden.
- ☐ d. Konstruktoren können per Definition nur die Sichtbarkeit `public` besitzen.

### ▼ Lösung

Kreuzen Sie alle wahren Aussagen an!

- ☒ a. In Java kann beim Überschreiben von Methoden der Rückgabotyp spezialisiert werden.
- ☐ b. Java verwendet ein prototypenbasiertes Objektmodell.
- ☒ c. Java nutzt als Ausführungschema eine Mischform, d.h. das Programm wird in Bytecode übersetzt und anschließend interpretiert.
- ☐ d. Java unterstützt keine Mehrfachvererbung bei Schnittstellenklassen.

### ▼ Lösung

Weil es in Java einen String Pool gibt, ist es ausreichend Strings via Referenzvergleich zu vergleichen, um auf inhaltliche Gleichheit zu überprüfen.

- ☐ a. Wahr
- ☒ b. Falsch

### ▼ Lösung

Kreuzen Sie an, welche der fragten Zeilen kompilieren und damit nicht zu einem Kompilierfehler führen.

```
1 package at.ac.uibk.pm.exam;
2
3 public class ClassA {
4     protected String y;
5     private int x;
6
7     public ClassA(int x, String y) {
8         this.x = x;
9         this.y = y;
10    }
11
12    public void foo() {
13        System.out.println(getZ());
14    }
15
16    public void foo2() {
17        this.x %= 3;
18    }
19
20    public int getX() {
21        return this.x;
22    }
23
24    public void setX(int x) {
25        this.x = x;
26    }
27
28    public void print() {
29        System.out.println(this.x + "___" + this.y);
30    }
31 }
32
```

```
1 package at.ac.uibk.pm.exam;
2
3 public class ClassB extends ClassA {
4     private final int z;
5
6     public ClassB(int x, String y, int z) {
7         super(x, y);
8         this.z = z;
9     }
10
11    public void foo() {
12        this.x = 5;
13        this.y = "Class B again";
14        this.x = 2 * this.getX();
15        this.setX(2 * this.getX());
16    }
17
18    public void print2() {
19        super.print();
20        System.out.println(this.z);
21    }
22
23    public int getZ() {
24        return this.z;
25    }
26
27    public void print() {
28        System.out.println(this.y + "/" + this.z);
29    }
30 }
31
32
```

```
1 package at.ac.uibk.pm.exam;
2
3 public class ClassC {
4     public static void main(String[] args) {
5         ClassA a = new ClassA(0, "Class A");
6         ClassB b = new ClassB(2, "Class B", 3);
7         a.foo();
8         b.foo();
9         a.foo2();
10        a.print();
11        b.print();
12        b.print2();
13    }
14 }
15
```

- ☒ a. ClassB - Zeile 7
- ☒ b. ClassA - Zeile 29
- ☒ c. ClassB - Zeile 15
- ☐ d. ClassB - Zeile 12

#### ▼ Lösung

Kreuzen Sie alle wahren Aussagen an!

- ☐ a. Die Deklaration und Zuweisung `float f = 5.3;` sind zulässig.
- ☐ b. Die Deklaration und Zuweisung `int short = 5;` sind zulässig.
- ☐ c. Die Deklaration und Zuweisung `unsigned int u = 5;` sind zulässig.
- ☒ d. Die Deklaration und Zuweisung `boolean b = false;` sind zulässig.

#### ▼ Lösung

Java verwendet klassenbasierte Sichtbarkeit. Daher ist der Zugriff auf `a` in der Methode `foo()` gestattet.

```
public class Z {  
    private double a;  
  
    public static double foo(Z z) {  
        return z.a;  
    }  
}
```

- ☒ a. Wahr
- ☐ b. Falsch

#### ▼ Lösung

Kreuzen Sie alle wahren Aussagen an!

- ☐ a. In Java ist es nicht möglich, zwei Methoden mit demselben Namen zu definieren.
- ☒ b. Die Signatur einer Methode besteht aus dem Methodennamen und der Liste der Typen der formalen Parameter.
- ☐ c. Das Überschreiben einer Methode bedeutet, dass ein Wert von einer Methode mit `void` Rückgabotyp zurückgegeben wird.
- ☐ d. Das Überladen einer Methode bezieht sich auf das Definieren von zwei Methoden mit demselben Namen, aber unterschiedlichen Rückgabotypen.

#### ▼ Lösung

Klassen aus dem Paket `java.lang` müssen nicht explizit importiert werden.

- ☒ a. Wahr
- ☐ b. Falsch

### ▼ Lösung

Kreuzen Sie an, welche der Methoden in der Klasse `ClassB` überladen sind!

```
1 public class ClassA {  
2     private void methodA(int i) {...}  
3     public void methodB(int i) {...}  
4     public void methodC(double d) {...}  
5     private static void methodD(int i) {...}  
6     private void methodE(double d) {...}  
7     protected void methodF(double d) {...}  
8     public Object methodG(float f) {...}  
9 }  
10  
1 public class ClassB extends ClassA {  
2     public void methodA(double d) {...}  
3     public void methodB(int i) {...}  
4     public void methodC(int i) {...}  
5     public static void methodD(double d) {...}  
6     public void methodE(double d) {...}  
7     public void methodF(int i) {...}  
8     public ClassA methodG(float f) {...}  
9 }  
10
```

- ☐ a. methodA
- ☐ b. methodB
- ☒ c. methodC
- ☐ d. methodD
- ☐ e. methodE
- ☒ f. methodF
- ☐ g. methodG

### ▼ Lösung

Welchen konkreten Java Datentyp hat das Ergebnis des Ausdrucks `302e3D` ?

### ▼ Lösung

Statische Methoden beschreiben das Verhalten einzelner Objekte.

- ☐ a. Wahr
- ☒ b. Falsch

### ▼ Lösung

Kreuzen Sie alle wahren Aussagen an!

- ☐ a. Eine lokale Variable vom Typ `int` wird automatisch mit `0` initialisiert.
- ☐ b. Eine Klassenvariable vom Typ `Float` wird automatisch mit `0.0f` initialisiert.
- ☒ c. Eine Objektvariable vom Typ `String` wird automatisch mit `null` initialisiert.

▼ Lösung

Kreuzen Sie alle wahren Aussagen an!

- ☐ a. Eine Klasse kann eine Unterklasse mehrerer konkreter Klassen sein.
- ☒ b. Eine Methode einer mit `final` deklarierten Klasse kann nicht überschrieben werden.
- ☒ c. Eine mit `final` deklarierte Methode kann nicht überschrieben werden.
- ☒ d. Eine konkrete Klasse kann eine Oberklasse mehrerer konkreter Klassen sein.

▼ Lösung

In Java wird call-by-reference verwendet. Daher können die Elemente in einem übergebenen Array in einer Methode verändert werden und diese Änderungen sind für den Aufrufenden sichtbar.

- ☐ a. Wahr
- ☒ b. Falsch

▼ Lösung

Kreuzen Sie alle Elemente, welche sich auf ein Objekt beziehen, an!

- ☐ a. Klassenvariablen
- ☐ b. Klassenbezogene Methoden
- ☒ c. Objektvariablen
- ☐ d. Statische Initialisierer