

## Hinweise

- Markieren Sie bei jeder Frage die am besten passende(n) Antwort(en)!
- Es gibt zwei Arten von Multiple-Choice-Fragen:
  - Fragen, die keine Punktzahl in Klammern angeben. Diese haben exakt eine richtige Antwort.
  - Fragen, die eine Punktzahl angeben. Hier müssen alle richtigen Antworten angekreuzt werden. Die Anzahl der korrekten Antworten kann sich von der Anzahl der Punkte unterscheiden.
- Falsch beantwortete Multiple-Choice-Fragen wirken sich negativ auf die Gesamtpunktzahl aus.

## Object Serialization, Generics

### Lektion 19

**Unter Serialisierung versteht man die Wiederherstellung eines Objekts aus einer Abfolge von Bytes.**

- ☐ Falsch
- ☐ Richtig

**Unter Deserialisierung versteht man die Überführung eines Objekts in eine Abfolge von Bytes.**

- ☐ Falsch
- ☐ Richtig

**Über einen ObjectOutputStream**

- ☐ werden Objekte direkt in eine Datei geschrieben.
- ☐ werden Java-Objekte serialisiert.
- ☐ werden Java-Objekte über das Netzwerk geschickt.

**Damit ein Objekt serialisierbar ist, genügt es das Interface Serializable zu implementieren**

- ☐ Richtig
- ☐ Falsch

**transient ist ein Schlüsselwort, um Attribute zu markieren, die während des Serialisierungsprozesses nicht serialisiert werden sollen.**

- ☐ Falsch
- ☐ Richtig

**persistent ist ein Schlüsselwort, um Attribute zu markieren, die während des Serialisierungsprozesses auf jeden Fall serialisiert werden sollen.**

- ☐ Falsch
- ☐ Richtig

**Kann es sinnvoll sein, nicht alle Attribute eines Objekts zu serialisieren? Bitte markieren Sie alle zutreffenden Aussagen! (3)**

- ☐ Nein, es ist immer sinnvoll alle Attribute zu serialisieren.
- ☐ Ja, es gibt Objekte, die plattform-/JVM-spezifisch sind.
- ☐ Ja, sicherheitsrelevante Daten sollten nicht serialisiert werden, ohne vorherige Maßnahmen wie Verschlüsselung zu treffen.
- ☐ Ja, es könnte effizienter sein, auf die Serialisierung einiger Attribute zu verzichten, wenn diese nicht persistiert oder versendet werden müssen, weil bspw. die Gegenseite diese Attribute nicht benötigt.

**Typparameterangaben erfolgen in spitzen Klammern hinter dem Klassennamen.**

- ☐ Richtig
- ☐ Falsch

**Als Typparameter sind Referenztypen und primitive Datentypen erlaubt.**

- ☐ Falsch
- ☐ Richtig

**Typparameter müssen mit T bezeichnet werden.**

- ☐ Richtig
- ☐ Falsch

**Wofür sind Generics gut? Bitte markieren Sie alle zutreffenden Aussagen! (3)**

- ☐ Um durch Downcasts den richtigen Typen beim Abruf von Daten aus der Liste zu bekommen.
- ☐ Um auf casts zu verzichten, wenn Daten aus einer Liste abgerufen werden.
- ☐ Um zu vermeiden, dass Objekte unterschiedlicher Datentypen aus Versehen zusammen in einer Liste gespeichert werden.
- ☐ Um mögliche Fehler beim Einfügen in eine Liste zur Laufzeit zu erkennen.
- ☐ Damit die gleichen Methoden oder ganze Klassen nicht mehrfach für unterschiedliche Datentypen implementiert werden müssen.

**Die Klasse ObjectOutputStream stellt welche der folgende Methoden zur Verfügung. Bitte markieren Sie alle zutreffenden Antworten! (2)**

- ☐ writeData(Object data)
- ☐ writeConstructor(Constructor data)
- ☐ writeClass(Class data)
- ☐ writeObject(Object data)
- ☐ writeInterface(Interface data)

**Bitte markieren Sie alle zutreffenden Aussagen! Das Interface Serializable... (2)**

- ☐ ist leer.
- ☐ enthält abstrakte Methoden
- ☐ dient nur zur Markierung
- ☐ enthält Attribute

**Die Rekonstruktion der Datenfelder eines Objektes beim Lesen aus einem Bytestrom übernimmt die Klasse**

- ☐ ByteArrayInputStream
- ☐ ObjectInputStream
- ☐ FileInputStream
- ☐ DataInputStream

**Falls beim Serialisieren eines Objektes ein Attribut nicht serialisierbar ist, wird eine Ausnahme geworfen vom Typ...**

- ☐ ArrayIndexOutOfBoundsException
- ☐ OutOfMemoryError
- ☐ ArithmeticException
- ☐ NotSerializableException

