

Name: \_\_\_\_\_

Matrikelnummer: \_\_\_\_\_

## Test 2 in Programmierparadigmen am 26. 1. 2026

Beispiele für **markierte** (= ausgewählte) Auswahlfelder:

Beispiele für **nicht markierte** (= nicht ausgewählte) Auswahlfelder:

Beurteilung: 1 Punkt pro Zeile mit gänzlich richtig markierten Auswahlfeldern

### Aufgabe 1 (10 Punkte)

A und B seien definiert durch `interface A<T> {}` und `interface B<R, S> extends A<S> {}`.  
Bitte markieren Sie jedes Auswahlfeld, bei dem der links stehende Typ ein Untertyp des darüber stehenden Typs ist. Es können keines, eines oder mehrere Felder pro Zeile auszuwählen sein.

|                                     | A<String>             | A<? super String>     | B<? extends String, ?> |
|-------------------------------------|-----------------------|-----------------------|------------------------|
| A<Object>                           | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| A<String>                           | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| B<String, Object>                   | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| B<Object, String>                   | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| B<?, ? extends String>              | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| B<? extends String, ?>              | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| B<? super String, ? extends String> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| B<? extends String, ? super String> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| B<A<String>, Object>                | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| B<A<Object>, String>                | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |

### Aufgabe 2 (10 Punkte)

Bitte markieren Sie in jeder Zeile das eine Auswahlfeld, bei dem die links stehende Aussage am ehesten eine Eigenschaft der darüber stehenden Parametrisierungsform in Java oder AspectJ ist.

Join-Points sind dynamisch identifizierbare Programmstellen  
kann die Werkzeuge zur Programmerzeugung beeinflussen  
„\*“ steht für beliebige Zeichen außer „.“ (Punkt)  
für binäre Methoden wird Rekursion benötigt  
zur Laufzeit sind Daten über Reflexion zugreifbar  
wird für Kernfunktionalität meist gemieden  
Lücken werden durch spezielle Parameter dargestellt  
`around()` ersetzt den spezifizierten Programmtext  
mit `@Retention` wird die Lebensdauer festgelegt  
Typschranken legen vorausgesetzte Funktionalität fest

| Generizität           | Annotationen          | Aspekte               |
|-----------------------|-----------------------|-----------------------|
| <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |

### Aufgabe 3 (10 Punkte)

Bitte markieren Sie jedes Auswahlfeld, bei dem der links stehende Typausdruck (mit Typnamen aus den Paketen `java.util.function` und `java.lang`) ein Typ des darüber stehenden Lambdas ist. Es können keines, eines oder mehrere Felder pro Zeile auszuwählen sein.

|   | $x \rightarrow y \rightarrow x+y$ | $(x, y) \rightarrow x+y$ | $x \rightarrow x+2$   | $x \rightarrow \{ \}$ | <code>Long::sum</code> |
|---|-----------------------------------|--------------------------|-----------------------|-----------------------|------------------------|
| <code>BinaryOperator&lt;String&gt;</code>                         | <input type="radio"/>             | <input type="radio"/>    | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| <code>LongBinaryOperator</code>                                   | <input type="radio"/>             | <input type="radio"/>    | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| <code>BiFunction&lt;Long, Long, Long&gt;</code>                   | <input type="radio"/>             | <input type="radio"/>    | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| <code>Function&lt;Long, Function&lt;Long, Long&gt;&gt;</code>     | <input type="radio"/>             | <input type="radio"/>    | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| <code>Function&lt;Long, LongUnaryOperator&gt;</code>              | <input type="radio"/>             | <input type="radio"/>    | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| <code>LongUnaryOperator</code>                                    | <input type="radio"/>             | <input type="radio"/>    | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| <code>Consumer&lt;String&gt;</code>                               | <input type="radio"/>             | <input type="radio"/>    | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| <code>Function&lt;Long, Function&lt;String, String&gt;&gt;</code> | <input type="radio"/>             | <input type="radio"/>    | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| <code>BiFunction&lt;Long, String, Long&gt;</code>                 | <input type="radio"/>             | <input type="radio"/>    | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |
| <code>BiFunction&lt;String, Long, String&gt;</code>               | <input type="radio"/>             | <input type="radio"/>    | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>  |

### Aufgabe 4 (10 Punkte)

Bitte markieren Sie jedes Auswahlfeld, bei dem die links stehende Kommandozeile (in `bash` ausgeführt) die darüber stehende Auswirkung hat. Es können keines, eines oder mehrere Felder pro Zeile auszuwählen sein.

|  | <code>cat &lt; a   wc &gt; b</code> | <code>cat a  &amp; wc &amp;&gt; b</code> | <code>cat a &amp;&gt; b &amp;&amp; wc b</code> | <code>cat a ; wc a</code> | <code>cat &lt; a  &amp; wc &amp;&gt; b &amp;</code> | <code>cat   wc</code> | <code>mehrere Prozesse laufen gleichzeitig</code> | <code>Prozesse laufen im Hintergrund</code> | <code>Prozesse über Pipeline verbunden</code> | <code>Standardeingabe von cat umgeleitet</code> | <code>Standardausgabe von cat umgeleitet</code> | <code>Fehlerausgabe von cat umgeleitet</code> |
|--|-------------------------------------|--|--|---------------------------|---|-----------------------|---|---|---|---|---|---|
| <code>for i in * ; do cat \$i   wc &gt; ./wc/\$i ; done</code>       | <input type="radio"/>               | <input type="radio"/>                    | <input type="radio"/>                          | <input type="radio"/>     | <input type="radio"/>                               | <input type="radio"/> | <input type="radio"/>                             | <input type="radio"/>                       | <input type="radio"/>                         | <input type="radio"/>                           | <input type="radio"/>                           | <input type="radio"/>                         |
| <code>for i in * ; do ( cat \$i &gt; ./bak/\$i &amp; ) ; done</code> | <input type="radio"/>               | <input type="radio"/>                    | <input type="radio"/>                          | <input type="radio"/>     | <input type="radio"/>                               | <input type="radio"/> | <input type="radio"/>                             | <input type="radio"/>                       | <input type="radio"/>                         | <input type="radio"/>                           | <input type="radio"/>                           | <input type="radio"/>                         |
| <code>if test 'cat a' = "a b c" ; then wc b ; else wc c ; fi</code>  | <input type="radio"/>               | <input type="radio"/>                    | <input type="radio"/>                          | <input type="radio"/>     | <input type="radio"/>                               | <input type="radio"/> | <input type="radio"/>                             | <input type="radio"/>                       | <input type="radio"/>                         | <input type="radio"/>                           | <input type="radio"/>                           | <input type="radio"/>                         |
| <code>cat a   wc</code>  | <input type="radio"/>               | <input type="radio"/>                    | <input type="radio"/>                          | <input type="radio"/>     | <input type="radio"/>                               | <input type="radio"/> | <input type="radio"/>                             | <input type="radio"/>                       | <input type="radio"/>                         | <input type="radio"/>                           | <input type="radio"/>                           | <input type="radio"/>                         |

### Aufgabe 5 (10 Punkte)

Bitte markieren Sie jedes Auswahlfeld, bei dem die links stehende Eigenschaft auf den darüber stehenden Methodenaufruf zutrifft. Es können keines, eines oder mehrere Felder pro Zeile auszuwählen sein.

- muss mit `InterruptedException` umgehen
- darf nur in einem Monitor verwendet werden
- soll nicht in einem Monitor verwendet werden
- gehört nicht zu Synchronisationsmechanismen
- suspendiert Ausführung des aktuellen Threads
- suspendiert Ausführung auf bestimmte Zeit
- weckt systemweit alle wartenden Threads auf
- weckt höchstens einen wartenden Threads auf
- weckt mindestens einen wartenden Thread auf
- gibt den Lock vorübergehend frei

|  | <code>Thread.sleep(5)</code> | <code>wait()</code>   | <code>notify()</code> | <code>notifyAll()</code> |
|--|------------------------------|-----------------------|-----------------------|--------------------------|
|  | <input type="radio"/>        | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>    |
|  | <input type="radio"/>        | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>    |
|  | <input type="radio"/>        | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>    |
|  | <input type="radio"/>        | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>    |
|  | <input type="radio"/>        | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>    |
|  | <input type="radio"/>        | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>    |
|  | <input type="radio"/>        | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>    |
|  | <input type="radio"/>        | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>    |
|  | <input type="radio"/>        | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>    |
|  | <input type="radio"/>        | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>    |
|  | <input type="radio"/>        | <input type="radio"/> | <input type="radio"/> | <input type="radio"/>    |

### Aufgabe 6 (10 Punkte)

Bitte markieren Sie jedes Auswahlfeld, bei dem die links stehende Aussage eine Eigenschaft des darüber stehenden Entwurfsmusters ist. Es können keines, eines oder mehrere Felder pro Zeile auszuwählen sein.

|   | Iterator              | Decorator             | Proxy                 | Prototype             | Factory-Method        |
|---|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| unterstützt Umkehrung der Abhängigkeiten  | <input type="radio"/> |
| führt zu vielen kleinen Objekten          | <input type="radio"/> |
| für oberflächliche Erweiterungen geeignet | <input type="radio"/> |
| oft große Anzahl an Unterklassen nötig    | <input type="radio"/> |
| ist Entwurfsmuster für Struktur           | <input type="radio"/> |
| ist erzeugendes Entwurfsmuster            | <input type="radio"/> |
| hilft große Zahl an Klassen zu vermeiden  | <input type="radio"/> |
| beruht auf Delegation                     | <input type="radio"/> |
| häufig als innere Klasse implementiert    | <input type="radio"/> |
| Objektidentität ist damit unzuverlässig   | <input type="radio"/> |

### Aufgabe 7 (10 Punkte)

Bitte markieren Sie in jeder Zeile das eine Auswahlfeld, bei dem die links stehende Aussage am ehesten eine Eigenschaft des darüber stehenden Entwurfsmusters ist.

|   | Iterator              | Decorator             | Proxy                 | Prototype             | Factory-Method        |
|---|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| zyklische Strukturen bereiten Probleme        | <input type="radio"/> |
| robuste Varianten werden bevorzugt            | <input type="radio"/> |
| Verantwortlichkeiten wieder entziehbar        | <input type="radio"/> |
| schlecht geeignet für umfangreiche Objekte    | <input type="radio"/> |
| mehrere gleichzeitige Abarbeitungen möglich   | <input type="radio"/> |
| kein Decorator, aber gleiche Struktur möglich | <input type="radio"/> |
| kann mit kovarianten Problemen umgehen        | <input type="radio"/> |
| es gibt externe und interne Varianten         | <input type="radio"/> |
| flache von tiefen Kopien unterschieden        | <input type="radio"/> |
| führt zu parallelen Klassenhierarchien        | <input type="radio"/> |

### Aufgabe 8 (10 Punkte)

Bitte markieren Sie in jeder Zeile das eine Auswahlfeld, bei dem die links stehende Aussage am ehesten eine Eigenschaft des darüber stehenden Entwurfsmusters ist.

|  | Iterator              | Decorator             | Proxy                 | Prototype             | Factory-Method        |
|--|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| <i>Component</i> ist ein Bestandteil           | <input type="radio"/> |
| <i>Product</i> ist ein Bestandteil             | <input type="radio"/> |
| <i>Subject</i> ist ein Bestandteil             | <input type="radio"/> |
| <i>Creator</i> ist ein Bestandteil             | <input type="radio"/> |
| <i>Aggregate</i> ist ein Bestandteil           | <input type="radio"/> |
| <i>Smart-Reference</i> ist eine Variante davon | <input type="radio"/> |
| wird auch <i>Wrapper</i> genannt               | <input type="radio"/> |
| wird auch <i>Surrogate</i> genannt             | <input type="radio"/> |
| wird auch <i>Virtual-Constructor</i> genannt   | <input type="radio"/> |
| wird auch <i>Cursor</i> genannt                | <input type="radio"/> |

### Aufgabe 9 (10 Punkte)

Bitte markieren Sie jedes Auswahlfeld, bei dem die links stehende Aussage eine Eigenschaft des darüber stehenden Entwurfsmusters ist. Es können keines, eines oder mehrere Felder pro Zeile auszuwählen sein.

|   | Visitor               | Singleton             | Template-Method       | Factory-Method        |
|---|-----------------------|-----------------------|-----------------------|-----------------------|
| verwaltet verwandte Operationen zentral | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| für Lazy-Initialization geeignet        | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| gibt Grundgerüst von Algorithmus vor    | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Anzahl erzeugter Objekte kontrollierbar | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| kann zu vielen Klassen führen           | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| kann zu vielen Methoden führen          | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| verwendet häufig Hooks                  | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| kann zu vielen Objekten führen          | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| mehrere Arten primitiver Operationen    | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| beruht auf dem Hollywood-Prinzip        | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |

### Aufgabe 10 (10 Punkte)

Bitte markieren Sie in jeder Zeile das eine Auswahlfeld, bei dem die links stehende Aussage am ehesten eine Eigenschaft des darüber stehenden Entwurfsmusters ist.

|   | Visitor               | Singleton             | Template-Method       | Factory-Method        |
|---|-----------------------|-----------------------|-----------------------|-----------------------|
| <i>accept</i> hat einen Parameter         | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| <i>instance</i> ist meist static          | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| <i>Element</i> ist ein Bestandteil        | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| hilft Duplikate im Code zu vermeiden      | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| kann als Anti-Pattern gesehen werden      | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Lambdas können Schreibaufwand reduzieren  | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| eine Methode ist häufig final             | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| globale Variable ist Alternative dazu     | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| unterstützt direkte Code-Wiederverwendung | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Umgang mit Unterklassen problematisch     | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |