

## **Aufgaben und Fragen zur Wiederholung**

1. Erklären Sie die Begriffe Objekt und Klassen mit ihren Zusammenhängen.
2. Erklären Sie, was Objektvariablen, Parameter und lokale Variablen sind. Wann werden Objektvariablen, wann lokale Variablen genutzt?
3. Was ist der Default-Konstruktor, wann existiert er, wann nicht?
4. Welche Varianten gibt es, damit Objektvariablen bei der Objekterzeugung Werte ungleich null haben?
5. Was sind get- und set-Methoden, wozu werden sie eingesetzt?
6. Java arbeitet mit Objektreferenzen, was bedeutet das? Erklären Sie dies an einem kleinen Beispiel.
7. Was ist ein Debugger, wie wird er typischerweise eingesetzt?
8. Welche graphischen Möglichkeiten zur Ablaufdarstellung gibt es in Aktivitätsdiagrammen (der in der Veranstaltung genutzten sequentiellen Version)?
9. Was ist „this“? Zeigen Sie, wie sich ein Objekt selbst bei einem Methodenaufruf übergeben kann.
10. Wie werden Sichtbarkeiten typischerweise in Java genutzt, warum gibt es sie?
11. Erklären Sie anhand eines Beispiels, welche Möglichkeiten es in einem switch-case-Statement gibt.
12. Was ist ein Algorithmus, durch welche Eigenschaften kann er charakterisiert werden?
13. Erklären Sie anhand eines eigenen Beispiels, was ein Iterator ist.
14. Geben Sie vier verschiedene Varianten an, wie man durch eine Sammlung iterieren kann.
15. Erklären Sie die grundsätzliche Funktionsweise von JUnit, gehen Sie dabei darauf ein, wie @Before, @After und @Test genutzt werden.
16. Was versteht man unter iterativ-inkrementeller Entwicklung? Warum ist sie sinnvoll?
17. Erklären Sie das Grundprinzip der Vererbung.
18. Was versteht man unter dem Überschreiben von Methoden?
19. Erklären Sie anhand eines eigenen Beispiels den Begriff „statische Polymorphie“.
20. Erklären Sie anhand eines eigenen Beispiels den Begriff „dynamische Polymorphie“. Welche Rolle spielen die Begriffe „Compile-Zeit“ und „Laufzeit“?
21. Was ist bei der Vererbung, bei Erstellung und der Nutzung von Konstruktoren zu beachten?
22. Was versteht man unter „casten“, was ist dabei zu beachten?
23. Welche Bedeutung hat die Klasse Object in Java?
24. Was versteht man unter Identität und inhaltlicher Gleichheit? Wie wird beides in Java umgesetzt?
25. Was versteht man unter „Clonen von Objekten“, wozu kann man das einsetzen?
26. Was ist eine abstrakte Klasse, wie werden sie wozu verwendet?

27. Was ist ein Interface, wie werden sie wozu verwendet?
28. Was versteht man unter „Design by Contract“, warum ist es für Praxis sehr relevant?
29. Wozu gibt es JavaDoc, wie wird es typischerweise eingesetzt?
30. Was ist Mehrfachvererbung, wie ist sie in Java umgesetzt?
31. Was ist ein Array, welche Gemeinsamkeiten und Unterschiede mit einer ArrayList gibt es?
32. Was ist ein mehrdimensionaler Array, wie kann er gefüllt und gelesen werden?
33. Wie werden Java-Programme typischerweise gestartet?
34. Wie kann man Java-Programme von der Konsole aus kompilieren und starten?
35. Was versteht man unter der lokalen Behandlung von Exceptions?
36. Beschreiben Sie wie ein try-catch-finally-Programmfragment abgearbeitet wird.
37. Wie kann man Exceptions weiterreichen, wann ist dies sinnvoll?
38. Wie erstellt man eigene Exceptions?
39. Wann sollten Exceptions, wann einfache if-Befehle genutzt werden?
40. Wie testen Sie, dass eine erwartete Exception auftritt?
41. Wie testen Sie, dass eine mögliche Exception nicht auftritt?
42. Beschreiben Sie den Zusammenhang zwischen hashCode() und equals().
43. Geben Sie einen Überblick, welche bisher bekannten Sammlungen unter welchen Voraussetzungen genutzt werden sollen.
44. Erklären Sie den Unterschied zwischen Objekt- und Klassenvariablen. Wann werden sie wie sinnvoll genutzt?
45. Was sind Klassenmethoden, auf welche Arten von Variablen kann in ihnen zugreifen?
46. Wie kann man Konstanten definieren, die in allen Klassen nutzbar sind?
47. Wozu gibt es Pakete?
48. Erklären Sie den Boundary-Control-Entity-Ansatz.
49. Erklären Sie die Unterschiede von int und Integer.
50. Was versteht man unter Autoboxing?
51. Was versteht man unter der Kurzschlussauswertung bei Booleschen Ausdrücken?
52. Wie kann man in Java Aufzählungen realisieren?
53. Wie verhalten sich int-Werte, wenn sie durch 0 (Zahl null) geteilt und wenn die Werte sehr groß (gegen unendlich und minus unendlich) werden?
54. Wie verhalten sich double-Werte, wenn sie durch 0 (Zahl null) geteilt und wenn die Werte sehr groß (gegen unendlich und minus unendlich) werden?