
Frage 1 Was ist ein Vorteil von herkömmlichen Schleifen in Java?

- A) Mehr Kontrolle über den Ablauf der Iteration
 - B) Lambda-Ausdrücke können verwendet werden
 - C) Automatische parallele Verarbeitung
 - D) Sie vermeiden Seiteneffekte
-

Frage 2 Wann sind Lambda-Streams besonders nützlich?

- A) Wenn der Iterationsprozess präzise gesteuert werden muss
 - B) Bei der Verarbeitung großer Datenmengen mit paralleler Ausführung
 - C) Wenn die Geschäftslogik sehr komplex ist
 - D) Wenn Seiteneffekte notwendig sind
-

Frage 3 Ein Nachteil von Lambda-Streams ist:

- A) Sie erlauben keine parallele Verarbeitung
 - B) Komplexe Geschäftslogik kann schwer zu verstehen sein
 - C) Es ist nicht möglich, mehrere Filteroperationen zu kombinieren
 - D) Streams können keine Listen verarbeiten
-

Frage 4 Wie helfen Lambda-Streams bei der Übersichtlichkeit des Codes?

- A) Sie machen den Code kürzer und klarer
 - B) Sie erhöhen die Anzahl der Zeilen im Code
 - C) Sie machen den Code schwieriger zu lesen
 - D) Sie bieten mehr Kontrolle über den Iterationsprozess
-

Frage 5 Was ist ein Vorteil der Verwendung von herkömmlichen Schleifen?

- A) Sie sind immer schneller als Lambda-Streams
 - B) Es ist einfacher, den Code Schritt für Schritt zu debuggen
 - C) Sie verhindern Seiteneffekte in der Geschäftslogik
 - D) Sie sind für die parallele Verarbeitung optimiert
-



Frage 6 Was ist ein häufiges Problem bei herkömmlichen Schleifen?

- A) Es ist nicht möglich, Listen zu iterieren
 - B) Schleifen verhindern den Energieverbrauch
 - C) Sie können schnell unübersichtlich und fehleranfällig werden
 - D) Sie sind immer langsamer als Lambda-Streams
-

Frage 7 Wann sollte man herkömmliche Schleifen anstelle von Lambda-Streams verwenden?

- A) Wenn eine komplexe Geschäftslogik umgesetzt werden muss
 - B) Wenn eine einfache Filteroperation benötigt wird
 - C) Wenn Daten parallel verarbeitet werden sollen
 - D) Wenn die Anzahl der Schleifeniterationsschritte keine Rolle spielt
-

Frage 8 Welcher Vorteil gilt sowohl für herkömmliche Schleifen als auch für Lambda-Streams?

- A) Beide verwenden funktionale Programmierung
 - B) Beide verhindern Fehler in der Geschäftslogik
 - C) Beide können mit Listen und Maps in Java verwendet werden
 - D) Beide arbeiten immer parallel
-

Frage 9 Welches Merkmal beschreibt Lambda-Streams am besten?

- A) Sie sind deklarativ und funktional
 - B) Sie sind verfahrensorientiert und bieten vollständige Kontrolle über die Iteration
 - C) Sie benötigen mehr Codezeilen als herkömmliche Schleifen
 - D) Sie machen den Code schwieriger zu lesen
-

Frage 10 Was ist ein Vorteil der parallelen Verarbeitung mit Lambda-Streams?

- A) Sie reduziert immer den Speicherverbrauch
 - B) Sie erhöht die Komplexität der Iterationskontrolle
 - C) Sie verbessert die Performance bei großen Datenmengen
 - D) Sie verhindert die Verwendung von Lambda-Ausdrücken
-



Frage 11 Welche der folgenden Aussagen sind Vorteile von Lambda-Streams?

- A) Sie machen den Code kürzer und übersichtlicher
 - B) Sie geben dem Entwickler vollständige Kontrolle über den Iterationsprozess
 - C) Sie unterstützen parallele Verarbeitung
 - D) Sie verhindern alle Arten von Fehlern im Code
-

Frage 12 Wann können herkömmliche Schleifen in Java nützlich sein?

- A) Wenn der Code komplexe Geschäftslogik enthält
 - B) Wenn die Daten nur in einem Schritt verarbeitet werden müssen
 - C) Wenn der Iterationsprozess präzise gesteuert werden soll
 - D) Wenn der Code immer kürzer sein soll
-

Frage 13 Welche Nachteile können bei der Verwendung von Lambda-Streams auftreten?

- A) Schwierigeres Debugging
 - B) Komplexität bei komplizierten Geschäftslogiken
 - C) Sie erlauben keine parallele Verarbeitung
 - D) Sie sind immer schneller als herkömmliche Schleifen
-

Frage 14 Welche der folgenden Merkmale treffen auf herkömmliche Schleifen zu?

- A) Sie bieten mehr Kontrolle über den Ablauf
 - B) Es ist einfacher, den Code Schritt für Schritt zu debuggen
 - C) Sie sind immer effizienter als Lambda-Streams
 - D) Sie erlauben keine Verarbeitung von Listen und Maps
-

Frage 15 Welche Vorteile bieten Lambda-Streams gegenüber herkömmlichen Schleifen?

- A) Sie unterstützen die funktionale Programmierung
 - B) Sie machen den Code oft kürzer und klarer
 - C) Sie geben dem Entwickler mehr Kontrolle über Schleifenvariablen
 - D) Sie sind immer die effizienteste Lösung, unabhängig von der Datenmenge
-

