



Startel

Dé IT-opleider

Java Advanced

In deze training worden de geavanceerde mogelijkheden van de programmeertaal: "Java" behandeld. Deze training is een mooie aanvulling als je de basiskennis en -vaardigheden in deze programmeertaal hebt opgedaan en jij je verder wilt verdiepen.

Algemene omschrijving

Java is een van de veelzijdigste en krachtigste programmeertalen die wordt gebruikt voor een groot aantal toepassingen, van webontwikkeling tot mobiele apps en bedrijfssoftware. De training Java Advanced is speciaal ontworpen voor softwareontwikkelaars die de basis van Java al beheersen en klaar zijn om hun kennis en vaardigheden naar een hoger niveau te tillen. In de training Java Advanced leer je de geavanceerde principes en technieken die nodig zijn om complexe en efficiënte Java-toepassingen te ontwikkelen.

De training Java Advanced biedt jou de mogelijkheid om diepgaande kennis te verkrijgen van de geavanceerde functies van Java. Dit omvat onderwerpen zoals de "final" keyword, try-with-resources, en het gebruik van Mockito voor unit testing. Verkrijg een uitgebreide en gedetailleerde verkenning van geavanceerde principes binnen Java. Je leert over onderwerpen zoals Navigable Set, Queue en ArrayDeque, evenals datastructuren zoals binaire bomen en binaire zoekalgoritmen.

Daarnaast verdiep jij je in diverse design patterns zoals Builder, Strategy, Iterator, Adapter, Singleton, Template en Visitor. Ook komen onderwerpen als recursie, algoritmen, threads, Runnable en synchronisatie aan bod. Bovendien bevat de training Java Advanced praktische oefeningen om jou helpen de opgedane kennis direct in praktijk te brengen.

Doelgroep

De training Java Advanced is met name geschikt voor softwareontwikkelaars met programmeerkennis en -vaardigheden in Java en zich verder willen bekwamen.

Leerdoelen

Door de training Java Advanced te volgen, zul je de volgende kennis en vaardigheden verkrijgen:

- Diepgaand inzicht in het gebruik van het keyword "final" in Java.
- Vaardigheid in het implementeren van try-with-resources voor effectief middelenbeheer.
- Bekwaamheid in het gebruik van Mockito voor unit testing en het maken van mock-objecten.

- Inzicht in Navigable Set en het gebruik van Queue en ArrayDeque voor efficiënte dataopslag en -beheer.
- Inzicht in datastructuren, zoals binaire bomen en binaire zoekalgoritmen.
- Het toepassen van diverse design patterns voor het ontwerpen van flexibele en herbruikbare code, waaronder:
 - Builder.
 - Strategy.
 - Iterator.
 - Adapter.
 - Singleton.
 - Template.
 - Visitor.
- Vaardigheid in het gebruik van recursie voor het oplossen van complexe problemen.
- Inzicht in geavanceerde algoritmen.
- Expertise in multithreading, inclusief het gebruik van threads, Runnable en synchronisatie voor het verbeteren van de prestaties en betrouwbaarheid van applicaties.
- Vaardigheid in het werken met regular expressions voor het verwerken en valideren van tekst.
- De Copy On Write ArrayList gebruiken voor threadveilige dataoperaties.
- Functioneel programmeren toepassen voor het schrijven van meer expressieve en onderhoudbare code.
- Diepgaand inzicht in input/output (IO)-operaties en een inleiding op NIO-2 voor efficiëntere data-invoer en -uitvoer.

Optioneel:

- Kennis van Spring Boot en het gebruik van Spring Boot-annotaties voor het ontwikkelen van robuuste en schaalbare applicaties.

Voorkennis

Voordat je deelneemt aan de training Java Advanced, is het van belang dat jij beschikt over basiskennis en -vaardigheden in de programmeertaal Java.

Als jij (nog) niet beschikt over kennis en vaardigheden in de programmeertaal Java, dan zou je kunnen overwegen om eerst de training Java Basic te gaan volgen.

Onderwerpen

- Keyword: "final".
- Try with Resources.
- Mockito.
- Navigable Set.
- Queue | ArrayDeque.
- Binary Trees.
- Binary Search.
- Design Patterns.
- Design Pattern: Builder.
- Design Pattern: Strategy.
- Design Pattern: Iterator.
- Design Pattern: Adapter.
- Design Pattern: Singleton.

- Design Pattern: Template.
- Design Pattern: Visitor.
- Recursion.
- Algorithms.
- Threads.
- Runnable.
- Synchronization.
- Regular Expressions.
- Copy On Write ArrayList.
- Functional Programming.
- IO.
- Introduction NIO-2.

Optioneel:

- Introduction to Spring Boot.
- Spring Boot annotations.