

JAV100: Cursus Java Programmeren

Code: JAV100

Duur: 5 dagen

Prijs: € 2250

Doelgroep voor de Java Training

Developers die willen beginnen met Java Programmeren of personen die Java code willen begrijpen.

Voorkennis van Java Programmeren

Elementaire kennis van en ervaring met programmeren is vereist om aan deze cursus te kunnen deelnemen. De principes van Object Oriented Programming met Java worden toegelicht. Voorafgaande kennis hiervan is bevorderlijk voor de begripsvorming.

Uitvoering Opleiding Java Programmeren

De theorie wordt behandeld aan de hand van presentatie slides en wordt afgewisseld met oefeningen. Demos zorgen voor een verduidelijking van de theorie. Er wordt gewerkt met Java 8. De cursustijden zijn van 9.30 tot 16.30.

Officieel Certificaat Java Programmeren

De deelnemers krijgen na het goed doorlopen van de cursus Java een officieel certificaat Java Programmeren.

Inhoud Java Cursus

Deze cursus java behandelt de basis principes van het Java platform en versie 8 programmeertaal . Op basis van op elkaar volgende oefeningen in een case study, leren de deelnemers om te programmeren met de variabelen, data types, operatoren en control flow constructies van de taal. De cursus behandelt ook de object georiënteerde begrippen als class, object, inheritance en polymorphism. Er wordt aandacht besteed aan fouten en exception handling en er wordt besproken hoe Java software wordt opgedeeld in packages. De splitsing van een programma in verschillende subtaken door threads en de synchronisatie van deze threads is ook onderwerp van de cursus. Nieuw geïntroduceerde kenmerken in de taal, zoals geparametriseerde types, generics, en het gebruik daarvan in het Collection Framework maken ook deel uit van het cursusprogramma. Ook wordt ingegaan op de in versie 8 geïntroduceerde lambda's en functional interfaces. Optionele modules zijn de toegang tot databases met JDBC, Java Beans en Graphical User Interfaces, GUI's. Deze modules kunnen, indien de tijd het toelaat worden behandeld. Deze cursus is een goede voorbereiding op het Java 8 Programmer Exam Part I (1Z1-808) ook wel OCA genoemd. In combinatie met de [cursus Advanced Java Programming](#) bereidt deze cursus ook voor op het Java 8 Programmer examen deel II (1Z0-809) ook wel OCP genaamd.



Java Programming Fundamentals



Module 1 : Basic Concepts	Module 2 : Language Syntax	Module 3 : Classes and Objects
History of Java Java Overview Language Editions Java Platform Java Community Process Libraries Java Language Security Application Types Compiling Programs Running Programs Compiler and Interpreter Application Structure Garbage Collection	Java Comments Variables Types of Variables Primitive Data Types Block Statements Operator Precedence Flow Control if else Statements switch Statement for and while Loop do while Loop break and continue Arrays Enhanced for Loop Strings Formatted Output	Classes and Objects Class Definition Encapsulation Access Modifiers Constructors Creating Objects Fields and Methods Using Objects static Modifier static Blocks Object Initializers this Keyword Parameter Passing Method Overloading Variable Arguments Object References final Modifier Object Destruction Lambda Functions
Module 4 : Inheritance	Module 5 : Exception Handling	Module 6 : Packages

<p>Inheritance extends Keyword Overriding and Hiding Polymorphism Abstract Classes Interfaces Implementing Interfaces Type Casting Implicit Casting Explicit Casting Cloneable Interface Cloning Objects</p>	<p>Error Conditions Exceptions in Java Exception Handling Syntax Exception Hierarchy Multiple Catch Clauses Multi Catch Clause finally Clause try with Resources Exception Information Predefined Exceptions Common Exceptions Throwing Exceptions User Defined Exceptions Chained Exceptions Rethrowing Exceptions Stack Traces Assertions</p>	<p>Packages Inside Packages Standard Packages Creating Packages Importing Classes Using Packages CLASSPATH import static Visibility Packaging in JAR Runnable JARS</p>
Module 7 : Threads	Module 8 : Synchronization	Module 9 : Special Classes
<p>Multiple Threads Benefits and Drawbacks Thread Characteristics Java Thread Model Thread Class Runnable interface Extending Thread Implementing Runnable Daemon Threads Thread Life Cycle States Thread Alive States Thread Class Methods Sleeping and Yielding Control Using join and interrupt Thread Priorities Suspending and Resuming</p>	<p>Concurrent Method Activation Synchronization Blocking on a Monitor Mutual Exclusion in Java Synchronized Statement Locking and Statics Deadlock Condition Synchronization Using wait and notify while Loop and notifyall</p>	<p>Inner Classes Types of Inner Classes Anonymous Inner Classes Inner Class Advantages Enumerations Old Enumerations Issues Enum Types Declaring Enums Enums as Constant Objects Enums are Classes Enum Methods and Fields Advantages new Enums Functional Interfaces</p>
Module 10 : Utility Classes	Module 11 : Collection Framework	Module 12 : Generics
<p>Object Class Wrapper Classes Autoboxing and Unboxing Overriding equals Math Class new Date API Regular Expressions Scanner Class Process Class Runtime Class System Class Locale Class Localizing Dates Localizing Numbers Localizing Currencies Javadoc</p>	<p>Collection Framework Framework Branches Implementation Classes Legacy Collections Collection Interface Iterator Interface Concrete Collections List Interface ArrayList Class LinkedList Class Adding to LinkedList Set and SortedSet NavigableSet and Map Comparable Interface Comparator Interface Map Interface Optional Methods Views</p>	<p>Generics Explained Syntax Generic Class Need for Generics Benefits of Generics Generic Class Examples Generics in Collections Generic Characteristics Type Erasure Bounded Type Parameter Generics and Subtyping Inheritance Relationships Wildcards Wildcards Arguments Upper Bounded Wildcards Lower Bounded Wildcards Raw Types Generic Methods</p>
Module 13 : Stream I/O	Optional Module : JDBC	Optional Module : Java Beans
<p>I/O Basics I/O Classes Byte Stream Classes Character Stream Classes Standard I/O Streams Stream Types Data Sink Streams Processing Streams Buffered Streams Reading Stream from Web Data Conversion Streams Serialization Serializable Classes Object Streams</p>	<p>JDBC JDBC Overall Architecture JDBC Drivers JDBC URL's Making Connections Executing a Statement Retrieving Results JDBC-ODBC Bridge</p>	<p>Software Components Java Beans Beans Terminology Bean Component Model Bean Event Pattern Event Firing Event Class Event Listener Interface Dispatching Events</p>

Optional Module : GUI's

Abstract Window Toolkit (AWT)
Controls and Containers
Layout Managers
Event Listeners
Swing Library
Window Painting
Swing and Threads
Java FX
Scene Graph