

Groovy Programmeren

Doelgroep Cursus Groovy Programmeren

De cursus Groovy Programmeren is bestemd voor Java developers die willen leren programmeren met de script taal Groovy.

Voorkennis Cursus Groovy Programmeren

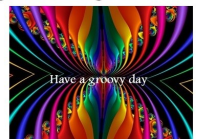
Om aan deze cursus te kunnen deelnemen is kennis van en ervaring met programmeren in Java vereist.

Uitvoering Training Groovy Programmeren

De theorie wordt behandeld aan de hand van presentatie slides en wordt afgewisseld met oefeningen. Illustratieve demo's zorgen voor een verdere verduidelijking van de behandelde begrippen. De cursustijden zijn van 9.30 tot 16.30.

Certificering Groovy Programmeren

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

Duur: 3 dagen**Prijs: € 2299**[Open Rooster](#)**Groovy Programming**

Inhoud Groovy Programmeren

In de cursus Groovy Programmeren leren de deelnemers to programmeren in de dynamische scripttaal Groovy die ook wel wordt beschouwd als Java++. Groovy is een JVM taal hetgeen betekent dat Groovy scripts interoperabel zijn met Java code en daardoor makkelijk samen met Java in dezelfde applicatie kunnen worden gebruikt. Groovy biedt een natuurlijke uitbreiding op de Java syntax en biedt onder andere een eenvoudiger syntax voor lists, ranges, maps en reguliere expressies.

Cursus Groovy Programmeren Intro

De cursus Groovy Programmeren begint met een introductie van de basiskennmerken van Groovy. Hierin wordt ingegaan op dynamic typing en type inference in Groovy. Ook wordt besproken hoe Groovy verschilt van Java, met veel minder boilerplate code en uitgebreide ondersteuning voor scripting.

Language Syntax en Data Structures

Vervolgens wordt de syntax van de Groovy taal behandeld, inclusief variabelen, datatypes, conditionals, operators en loops. Ook worden de data structures van Groovy besproken, zoals lists, maps en sets, en hoe Groovy gebruikmaakt van het Java Collection Framework.

Cursus Groovy Programmeren: Methods en Closures

Daarnaast komen Groovy methods, parameters passing en closures aan bod. Groovy closures zijn anonieme codeblokken die als functieobjecten met de scope waarin ze zijn gecreëerd, worden opgeslagen om op een later tijdstip, al dan niet met extra parameters, te worden uitgevoerd.

Classes en Traits

Groovy is, net als Java, een objectgeoriënteerde taal. Er wordt aandacht besteed aan hoe je in Groovy gebruikmaakt van classes, constructors, fields, interfaces en traits.

XML en JSON

Ook file I/O staat op het programma. In dit verband wordt ingegaan op de veelgebruikte dataformaten XML en JSON en hoe Groovy hiermee omgaat.

Meta Programming in Groovy

Tenslotte wordt de cursus afgesloten met de behandeling van meta programming in Groovy. Hiermee is het mogelijk om methods dynamisch aan te roepen en classes en methods on the fly te creëren. Voor meer informatie over Groovy, bezoek de [officiële Groovy website](#).

Deze cursus is ideaal voor ontwikkelaars die willen leren hoe ze Groovy kunnen gebruiken in hun projecten. Voor meer cursussen, zoals [Reactive Programmeren met RxJS](#), kun je ook bij ons terecht.

Modules Groovy Programmeren

Module 1 : Groovy Intro	Module 2 : Language Syntax	Module 3 : Data Structures
What is Groovy? Groovy Features HelloWorld in Java HelloWorld in Groovy Removing Noise Removing Boilerplate Dynamic Types Variable Interpolation POJOs on Steroids Script Support Differences with Java Running Groovy Scripts Groovy Development Kit	Groovy Variables and Data Types Wrapper Types Number Methods Groovy Strings Groovy Conditionals Switch Statement Groovy Operators Other Operators Elvis Operator Range Operator Groovy Loops For in Loop Each Loop	Groovy Lists List Methods List Manipulation Groovy Maps Map Methods Map Manipulation Java Collection Interface Concrete Collections List Interface ArrayList Class LinkedList Class Set and SortedSet Map Interface
Module 4 : Methods and Closures	Module 5 : File I/O	Module 6 : Classes and Traits
Groovy Methods Method Parameters Return Values Locals versus Globals What are Closures? Benefits of Closures Groovy Closures Closure Parameters Closure Variables Closures in Methods Closures and Collections Find and FindAll Any and Every Collect	IO Basics Java Stream Classes Stream Types Nested Streams Groovy IO Reading Files Writing Text Data Conversion Streams DataStreams Java DataStreams Groovy Copy and Delete Directory Methods Traversing File Trees Executing External Processes	Groovy Classes Normal Class Duck Typing Property Support Inner Classes Inner Class Usage Anonymous Inner Class Abstract Classes Interfaces Constructors Fields Comparator Traits Meaning of this
Module 7 : XML and JSON Handling	Module 8 : Meta Programming	
XML Processing Options DOM Tree SAX Callbacks XML Support in Groovy MarkupBuilder JSON Object Representation JSON Array Representation JSON with Array and Objects Groovy Json Classes Producing JSON from Map Produce JSON From Pogo JSON with JSONSlurper	Groovy is Dynamic Dynamic Methods Adding Properties Meta Object Protocol Invoker Operation invokeMethod GroovyObject interface Groovy Interceptable Metaclass Intercepting Methods Expando GPath	