

PRG500 : Ruby Programmeren

Code : PRG500

Duur : 3 dagen

Categorie : Scripting

Doelgroep :

Deze cursus is bedoeld voor ervaren developers die de scripttaal Ruby willen leren.

Voorkennis :

Kennis en ervaring met programmeren in een andere programmeertaal, zoals C, C++, C#, Visual Basic, Java of Perl is gewenst.

Uitvoering :

De theorie wordt behandeld aan de hand van presentatie slides en wordt afgewisseld met praktische oefeningen. Illustratieve demo's zorgen voor verdere verduidelijking van de concepten.



Ruby Programming



Inhoud :

In deze cursus leren de deelnemers programmeren in Ruby aan de hand van de syntax en taalconstructies van deze scripttaal. Na een introductie over de achtergronden van Ruby, de installatie en de wijze waarop Ruby code kan worden uitgevoerd, komen de variabelen, data types en control flow aan de orde. Ook is er aandacht voor methods en parameter overdracht in Ruby en voor de object geïntereerde aspecten zoals classes, objects, inheritance en polymorfisme. Daarnaast komt ook het afhandelen van fouten door middel van exception handling ter sprake. Vervolgens wordt ingegaan op het indelen van code in modules, modules als namespaces en mixins evenals op het gebruik van modules uit de standard library. Ook closures zoals blocks, lambda's en Procs komen aan de orde en er wordt ingegaan op Ruby Meta Programming met introspection, Open Classes en dynamic method invocation. Tenslotte komt het schrijven van Ruby Gems aan bod en wordt een overzicht gegeven van de mogelijkheden van de Ruby Standard library.

Module 1 : Ruby Intro

- What is Ruby?
- Ruby Timeline
- Ruby Object Orientation
- Ruby Installation
- Interactive Ruby
- Ruby Execution
- Loading Ruby Code
- Ruby Naming Conventions
- Executing External Programs
- Ruby Blocks
- Ruby Resources

Module 2 : Variables and Data Types

- Numbers and Big Numbers
- Strings and String Literals
- String Interpolation
- Arrays
- Hash
- Range
- RegExp
- Struct
- Types of Variables
- Ruby Naming Conventions
- Constants
- Local Variables
- Instance Variables
- Class Variables
- Global Variables
- Pre-defined Variables

Module 3 : Control Flow

- Statements
- Assignment
- Assignment operators
- Conditionals
- Multiple Selection
- while Loop
- until Loop
- for Loop
- each Iteration
- Aritmetic operators
- Comparison operators
- Ruby truth
- Equality
- Logical Operators

Module 4 : Methods and Parameters

- Method Definitions
- Invoking Methods
- Methods and Parenthesis
- Return values
- Default value argument
- Variable Argument List
- Array Argument
- Hash Argument
- Methods with Code Block
- Method with Bang
- Aliasing Methods

Module 5 : Classes and Objects

- Classes and Objects
- Example Class and Object
- Objects in Ruby
- Ruby Classes
- Class Syntax
- Creating Objects
- Object Initialization
- Getters and Setters
- Attribute Accessors
- Current Object
- Class Variables
- Class Methods
- Method Visibility
- Singleton Methods
- Inheritance
- Overriding
- Method Lookup
- Duck Typing
- Operator Overloading

Module 6 : Exception Handling

- Error Handling
- Exception Handling
- Raising Exceptions
- Handling Exceptions
- Exception Class
- Exception Class Hierarchy
- Typed Exception Handling
- Ensure Block
- Retry Command
- Exception Handling
- Throw and Catch
- Raising Exceptions
- User Defined Exceptions

Module 7 : Modules

- Modules
- Module Importing
- Files without Namespacing
- Modules for Namespacing
- Namespaces
- Mixins
- Mixin Example
- Include versus Extend
- Mixins and Inheritance Chain
- Modules versus Classes
- Comparable Module
- Enumerable Module

Module 8 : Closures

- What are Closures?
- Benefits of Closures
- Lambdas
- Lambdas with Parameters
- Procs
- Proc and Lambda
- Procs versus Lambdas
- Proc as Argument and Return
- Proc Objects as Closure
- What are Blocks?
- Representing Blocks
- Calling Blocks with Yield
- Passing Arguments
- Ampersand Operator
- From Proc to Block

Module 9 : Meta Programming

- What are Gems?
- About RubyGems
- Creating Gems
- Hello World Gem
- gemspec file
- Installing and Using the Gem
- Publish the Gem
- Grabbing the Gem
- Adding Code
- Multiple Files
- Rake
- Rake Demo
- Writing Tests
- Simple Test
- Documenting Code
- Using Rdoc
- What is Bundler?
- Using Bundler

Module 10 : Ruby Gems

- What are Gems?
- About RubyGems
- Creating Gems
- Hello World Gem
- gemspec file
- Installing and Using the Gem
- Publish the Gem
- Grabbing the Gem
- Adding Code
- Multiple Files
- Rake
- Rake Demo
- Writing Tests
- Simple Test
- Documenting Code
- Using Rdoc
- What is Bundler?
- Using Bundler

Module 11 : Ruby Standard Library

- Standard Library Overview
- Files
- Accessing Files
- File Open Modes
- Reading and Writing
- Directories
- Date and Time
- XML Access
- DOM Parsing
- SAX Parsing
- MultiThreading