

Ruby Programming

Audience Course Ruby Programming

The course Ruby Programming is intended for experienced developers who want to learn the Ruby scripting language.

Prerequisites Course Ruby Programming

Knowledge and experience with programming in another programming language, such as C, C++, C#, Visual Basic, Java or Perl is required.

Realization Training Ruby Programming

The theory is discussed on the basis of presentation slides and is interspersed with practical exercises. Illustrative demos provide further clarification of the concepts. The course material is in English.

Certification Ruby Programming

Participants receive an official certificate Ruby Programming after successful completion of the course.

Duration: 3 days

Price: € 1850

[Open Schedule](#)



Ruby Programming



Content Course Ruby Programming

In the course Ruby Programming participants learn to program in Ruby using the syntax and language constructs of that language.

Ruby Syntax

After an introduction about the background of Ruby, the installation and the way in which code can run, the variables, data types and control flow are discussed.

Classes en Objects

Attention is also paid to methods and parameter passing in Ruby and to object oriented aspects such as classes, objects, inheritance and polymorphism.

Exception Handling

The handling of errors through exception handling is discussed as well.

Modules and Mixins

The course continues with a discussion of partitioning code into modules, modules as namespaces and mixins as the use of modules from the standard library.

Blocks and Procs

Also closures such as blocks, lambdas and Procs are discussed and Ruby meta programming is treated with introspection, Open Classes and dynamic method invocation.

Ruby Gems

Finally writing RubyGems is on the course schedule and an overview of the capabilities of the Standard Library is given.

Modules Course Ruby Programming

| Module 1 : Ruby Intro | Module 2 : Variables and Data Types | Module 3 : Control Flow |
|---|---|---|
| What is Ruby? Ruby Timeline Object Orientation Installation Interactive Ruby Ruby Execution Loading Ruby Code Naming Conventions Executing External Programs Ruby Blocks Resources | Numbers and Big Numbers Strings and String Literals String Interpolation Arrays Hash Range Struct Types of Variables Naming Conventions Constants Pre-defined Variables | Statements Assignment operators Conditionals Multiple Selection while and until Loop for Loop each Iteration Arithmetic operators Comparison operators Ruby truth Logical Operators |
| Module 4 : Methods and Parameters | Module 5 : Classes and Objects | Module 6 : Exception Handling |
| 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 | Classes and Objects in Ruby Object Initialization Attribute Accessors Current Object Class Variables and Methods Method Visibility Singleton Methods Inheritance Overriding Method Lookup Duck Typing | Error Handling Exception Handling Raising Exceptions Handling Exceptions Exception Class Hierarchy Typed Exception Handling Ensure Block Retry Command Throw and Catch Raising Exceptions User Defined Exceptions |
| Module 7 : Modules | Module 8 : Closures | Module 9 : Meta Programming |
| 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 | Benefits of Closures Lambdas and Procs Lambdas with Parameters 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 | What is Introspection? Introspection Code Classes are Open Class Definition are Executable Receivers Classes are Objects Dynamic Method Invocation Method Missing Invoking method_missing define_method Evaluating Code Dynamic Typing |
| Module 10 : Ruby Gems | Module 11 : Ruby Standard Library | |
| What are Gems? Creating Gems gemspec file Installing and Using the Gem Publish the Gem Grabbing the Gem Rake Writing Tests Documenting Code Using Rdoc Using Bundler | Standard Library Overview Files Accessing Files File Open Modes Reading and Writing Directories Date and Time XML Access DOM Parsing SAX Parsing MultiThreading | |