Document Generated: 11/12/2026

Learning Style: On Demand

Technology:

Difficulty: Intermediate

Course Duration: 5 Hours

Ruby Intermediate



About this course:

The Ruby: Intermediate is a programming language course which is the second course in the series of courses on Ruby programming language. The course provides hands on experience and builds on the topics taught in the fundamentals course. The course covers more advanced topics such as working with forms, databases, rails framework, validation techniques and styles. The course winds up by teaching the students to put all of the components together to create a

presentable application for end users.

This intermediate course provides a great learning experience for mid-career level programmers to understand the more advanced topics of Ruby programming language. The course enables the students to write a simple code in Ruby and develop the concepts that ultimately help in creating real-world applications using Ruby programming language.

Course Objective:

- Learn the core syntax of Ruby programming language
- Create fill in forms
- Integrate the forms with databases
- Learn data validation techniques
- Create presentable and user friendly applications using styles
- Customize user-specific applications by creating user accounts

Test and assure quality of the applications

Audience:

- Software developers working on Ruby programming language
- Individuals experienced with other programming languages such as C++, Java and Perl
- Experienced professionals in .NET framework
- QA engineers

Prerequisite:

- The course requires the students to have sound knowledge of working with Java.
- The students should be able to understand and demonstrate object oriented programming with Java 7 SE and Java 8.
- In addition to this, it is highly recommended that the students should undertake the Ruby: Fundamentals course before registering for this course.

Course Outline:

Chapter 01 - Conditionals and Blocks

- **Topic A: Standard Conditionals - Part 1**
- Standard Conditionals - Part 2
- Standard Conditionals - Part 3
- **Topic B: Compound Conditionals - Part 1**
- Compound Conditionals - Part 2
- Compound Conditionals - Part 3
- **Topic C: Operators - Part 1**
- Operators - Part 2
- Operators - Part 3

- **Topic D: Switch - Part 1**
- Switch - Part 2
- Switch - Part 3
- **Topic E: Guard - Part 1**
- Guard - Part 2
- Guard - Part 3
- **Topic F: What Are Blocks - Part 1**
- What Are Blocks - Part 2
- What Are Blocks - Part 3
- **Topic G: Yield - Part 1**
- Yield - Part 2
- Yield - Part 3
- **Topic H: Block Given - Part 1**
- Block Given - Part 2
- Block Given - Part 3
- **Topic I: File Class - Part 1**
- File Class - Part 2
- File Class - Part 3
- **Topic J: Yield in Rails - Part 1**
- Yield in Rails - Part 2
- Yield in Rails - Part 3
- **Topic K: Ampersand - Part 1**
- Ampersand - Part 2
- Ampersand - Part 3

Chapter 02 - Metaprogramming

- **Topic A: Proc - Part 1**
- Proc - Part 2
- Proc - Part 3
- **Topic B: Lambda - Part 1**
- Lambda - Part 2
- Lambda - Part 3
- **Topic C: Proc vs. Lambda - Part 1**
- Proc vs. Lambda - Part 2
- Proc vs. Lambda - Part 3
- **Topic D: Lambdas in Rails - Part 1**
- Lambdas in Rails - Part 2
- Lambdas in Rails - Part 3
- **Topic E: Metaprogramming - Part 1**
- Metaprogramming - Part 2
- Metaprogramming - Part 3
- **Topic F: Open Class - Part 1**
- Open Class - Part 2
- Open Class - Part 3
- **Topic G: Method Missing - Part 1**
- Method Missing - Part 2
- Method Missing - Part 3
- **Topic H: Define Method - Part 1**
- Define Method - Part 2

- Define Method - Part 3
- **Topic I: Rails Metaprogramming - Part 1**
- Rails Metaprogramming - Part 2
- Rails Metaprogramming - Part 3

Chapter 03 - Frameworks and Algorithms

- **Topic A: Error Handling and Syntax - Part 1**
- Error Handling and Syntax - Part 2
- Error Handling and Syntax - Part 3
- **Topic B: Antipatterns - Part 1**
- Antipatterns - Part 2
- Antipatterns - Part 3
- **Topic C: Error Logger - Part 1**
- Error Logger - Part 2
- Error Logger - Part 3
- **Topic D: Sinatra - Part 1**
- Sinatra - Part 2
- Sinatra - Part 3
- **Topic E: Rails - Part 1**
- Rails - Part 2
- Rails - Part 3
- **Topic F: Insertion Sort - Part 1**
- Insertion Sort - Part 2
- Insertion Sort - Part 3
- **Topic G: Quicksort - Part 1**
- Quicksort - Part 2
- Quicksort - Part 3
- **Topic H: Binary Search - Part 1**
- Binary Search - Part 2
- Binary Search - Part 3

Credly Badge:



Display your Completion Badge And Get The Recognition You Deserve.

Add a completion and readiness badge to your Linkedin profile, Facebook page, or Twitter account to validate your professional and technical expertise. With badges issued and validated by Credly, you can:

- Let anyone verify your completion and achievement by clicking on the badge
- Display your hard work and validate your expertise
- Display each badge's details about specific

skills you developed.

Badges are issued by QuickStart and verified through Credly.

Find Out More or See List Of Badges