

Ruby Programming



Ruby was named after the precious gem.

Ruby is an interpreted, object-oriented programming language. Its creator, Yukihiro Matsumoto, a.k.a “Matz,” released it to the public in 1995. Its history is covered here. Its many features are listed here.

The book is currently broken down into several sections and is intended to be read sequentially. **Getting started** will show how to install and get started with Ruby in your environment. **Basic Ruby** demonstrates the main features of the language syntax. The **Ruby language** section is organized like a reference to the language. **Available modules** covers some of the standard library. **Intermediate Ruby** covers a selection of slightly more advanced topics. Each section is designed to be self contained.

1 Table of Contents

1.1 Getting started

Overview
Installing Ruby
Ruby editors
Notation conventions
Interactive Ruby
Mailing List FAQ

1.2 Basic Ruby

Hello world
Strings
Alternate quotes
Here documents
ASCII
Encoding
Introduction to objects
Ruby basics
Data types — numbers, strings, hashes and arrays
Writing methods
Classes and objects
Exceptions

1.3 Ruby Semantic reference

- Syntax
 - Lexicology
 - Identifiers
 - Comments
 - Embedded Documentation
 - Reserved Words
 - Expressions
 - Variables and Constants
 - Local Variables
 - Instance Variables
 - Class Variables
 - Global Variables
 - Constants
 - Pseudo Variables
 - Pre-defined Variables
 - Literals
 - Numerics
 - Strings
 - Interpolation
 - Backslash Notation
 - The % Notation
 - Command Expansion
 - Regular Expressions

- Arrays
- Hashes
- Ranges
- Symbols
- Operators
 - Assignment
 - Self Assignment
 - Multiple Assignment
 - Conditional Assignment
 - Scope
 - and
 - or
 - not
- Control Structures
 - Conditional Branches
 - if
 - if modifier
 - unless
 - unless modifier
 - case
 - Loops
 - while
 - while modifier
 - until
 - until modifier
 - for
 - for ... in
 - break
 - redo
 - next
 - retry
 - Exception Handling
 - raise
 - begin
 - rescue modifier
 - Miscellanea
 - return
 - returning
- Methods
 - super
 - Iterators
 - yield
- Classes
 - Class Definition
 - Instance Variables
 - Class Variables
 - Class Methods
 - Instantiation
 - Declaring Visibility

- Private
- Public
- Protected
- Instance Variables
- Inheritance
- Mixing in Modules
- Ruby Class Meta-Model
- Ruby Hooks add the ability to know when new methods are defined et al.

See also some `rdoc` documentation on the various keywords.

1.4 Built in Classes

This is a list of classes that are available to you by default in Ruby. They are pre-defined in “core.”

- Built-in Functions
- Predefined Variables (globals)
- Predefined Classes
 - Object
 - Array
 - Class
 - Comparable
 - Encoding
 - Enumerable
 - Enumerable::Enumerator
 - Enumerator
 - Exception
 - FalseClass
 - Fiber
 - IO
 - File
 - File::Stat
 - GC
 - GC::Profiler
 - Marshal
 - Method
 - Math
 - Module
 - Class
 - NilClass
 - Numeric
 - Integer
 - Bignum
 - Fixnum
 - Float
 - Range
 - Regexp

- [RubyVM](#)
- [String](#)
- [Struct](#)
 - [Struct::Tms](#)
- [Symbol](#)
- [Time](#)
- [Thread](#)
- [TrueClass](#)

1.5 Available Standard Library Modules

These are parts of Ruby that you have available (in the standard library, or via installation as a gem). To use them you typically have to require some filename, for example require 'tracer' would make accessible to you the Tracer class.

You can see a list of basically all the (std lib ruby) modules available in the [ruby source](#) and [lib readme](#). There are a several more modules available in the std lib, which are C based extensions. You can see their list [here](#).

- [BigDecimal](#) gives you a way to have arbitrary precision Decimal style numbers. Never succumb to rounding errors again!
- [Debugger](#) gives you a way to step through debug your Ruby code.
- [Distributed Ruby](#) (DRb) gives you a way to make remote procedure calls against objects in a different VM.
- [mkmf](#) is a utility used to generate makefiles for ruby extensions.
- [Mutex](#) gives you a way to control thread concurrency.
- [Net::HTTP](#) gives you a way to download web pages.
- [Open3](#) gives you a way to run a sub-process and have easy access to its I/O.
- [OpenSSL](#) is a wrapper to the OpenSSL (C) library, giving you access to secure socket connections.
- [Pathname](#) gives you an easy way to manipulate filenames and create/remove files.
- [Profiler](#) gives you a way to profile what is taking up the most time in your code.
- [OpenURI](#) gives you a way to download files using ruby.
- [REXML](#) is a way to parse XML in pure Ruby.
- [Ripper](#) gives you a way to parse pure Ruby code into an AST.
- [Socket](#) gives you access to Network connectivity.

- [Tracer](#) gives you a way to see which lines of your code are being executed and in what order.
- [Win32::Registry](#) gives you a way to query and edit the windows registry.
- [Win32API](#) gives you a way to call into specific windows core methods easily.
- [WIN32OLE](#) gives you a way to use Windows OLE.

1.6 Other Libraries

- [Database Interface Modules](#)

1.6.1 GUI Libraries

- [GUI Toolkit Modules](#) gives a run down of various options for ruby GUI programming.

Here is info on some specifically:

- [Tk](#)
- [GTK2](#) Notes on the GTK/Gnome bindings.
- [Qt4](#)

2 Intermediate Ruby

Here are some more in depth tutorials of certain aspects of Ruby.

[Unit testing](#)
[RubyDoc](#)
[Rake](#)
[RubyGems](#)
[Running Multiple Processes](#)
[Using Network Sockets](#)
[Building C Extensions](#)
[Rails](#)
[Embedding Ruby within a separate C program](#)

3 External links

- [Ruby homepage](#)
- [Access to various Ruby mailing lists](#)
- [Ruby Talk FAQ](#)
- [RubyForge: The Repository for Open-source Ruby Projects](#)
- [a place where people vote on things they like most per category](#)

3.1 Documentation

3.1.1 Core Docs

- [Ruby Documentation Homepage](#)
- [ruby-doc.org](#) various ruby documentations and tutorials, as well as information on how to update ruby's core docs should you so desire.
- [webri](#) ruby core api as a webri

3.1.2 gem docs

- “[rdoc.info](#)”:<http://www.rdoc.info> yard rdoc's for gems hosted on github
- “[ruby toolbox](#)”:<http://ruby-toolbox.com> list of gems by popularity

3.2 Learning Ruby

- [Try Ruby](#) - learn the basics of the Ruby language in your browser.
- [LearnStreet](#) - Free online Ruby tutorials & practice exercises
- [A Ruby Tutorial that Anyone can Edit](#) (broken)
- [Learning Ruby](#) A free tool to find and learn Ruby concepts

3.3 Books

Print

- [The Ruby Programming Language](#) by David Flanagan, Yukihiro Matsumoto aka “Matz,” the creator of Ruby. Also covers 1.9
- [Programming Ruby 3](#) (aka “Pickaxe”) — this 2009 version covers Ruby 1.9
- [Ruby by Example](#)

Online

- [Programming Ruby](#) (a.k.a. “Pickaxe”) 1st edition online version
- [Ruby Study Notes](#)
- [Why's \(Poignant\) Guide To Ruby](#)
- [Humble Little Ruby Book](#)
- [Ruby Hacker's Guide](#) is a guide to the guts of (mostly 1.8) ruby, translated from its original Japanese.

3.3.1 Quick References

- [Ruby Quick Reference](#) (some of more obscure expressions are explained)
- [Ruby Cheat Sheets](#) (a list of some different Ruby cheat sheets)

4 Text and image sources, contributors, and licenses

4.1 Text

- **Ruby Programming** *Source:* https://en.wikibooks.org/wiki/Ruby_Programming?oldid=3140655 *Contributors:* Dysprosia~enwikibooks, Ramir, Yath~enwikibooks, Mkn, Sjc~enwikibooks, Panic2k4, Krischik, Keagan, Iamunknown, Derbeth, Dfrankow, Darklama, Withinfocus, Hyad~enwikibooks, Orderud, Pavan, Yuuki Mayuki~enwikibooks, Jguk, Rsperberg, Marburg, Valters, Effejetsanders, Briand~enwikibooks, Supriya kunjeer, OinkOink~enwikibooks, Herraotic, Ravichandar84, EvanCarroll, Damien Karras, Wantless, Dallas1278, Elizabeth Barnwell, JackPotte, Whym, Mehryar, QuiteUnusual, Jk33, Eisel98, Adrignola, Shirock, Dirk Hünninger, Abd, Rogerdpack, Mjbmrbot, Sebleclerc, Linuxios, Evotopid, Jdaudier, Computers1000 and Anonymous: 55

4.2 Images

- **File:00_percents.svg** *Source:* https://upload.wikimedia.org/wikipedia/commons/6/60/00_percent.svg *License:* CC0 *Contributors:* File:00%.svg *Original artist:* Siebrand
- **File:100_percents.svg** *Source:* https://upload.wikimedia.org/wikipedia/commons/2/24/100_percent.svg *License:* CC0 *Contributors:* File:100%.svg *Original artist:* Siebrand
- **File:25_percents.svg** *Source:* https://upload.wikimedia.org/wikipedia/commons/c/ce/25_percent.svg *License:* CC0 *Contributors:* File:25%.svg *Original artist:* Ftiercel
- **File:50_percents.svg** *Source:* https://upload.wikimedia.org/wikipedia/commons/6/62/50_percents.svg *License:* CC0 *Contributors:* File:50%.svg *Original artist:* Ftiercel
- **File:75_percents.svg** *Source:* https://upload.wikimedia.org/wikipedia/commons/6/62/75_percent.svg *License:* CC0 *Contributors:* File:75%.svg *Original artist:* Ftiercel
- **File:Gnome-mime-application-pdf.svg** *Source:* <https://upload.wikimedia.org/wikipedia/commons/b/b6/Gnome-mime-application-pdf.svg> *License:* LGPL *Contributors:* <http://ftp.gnome.org/pub/GNOME/sources/gnome-themes-extras/0.9/gnome-themes-extras-0.9.0.tar.gz> *Original artist:* David Vignoni
- **File:Printer.svg** *Source:* <https://upload.wikimedia.org/wikipedia/commons/2/23/Printer.svg> *License:* Public domain *Contributors:* The Tango! Desktop Project *Original artist:* The people from the Tango! project
- **File:Ruby_logo.png** *Source:* https://upload.wikimedia.org/wikipedia/commons/f/f1/Ruby_logo.png *License:* CC BY-SA 2.5 *Contributors:* <http://rubyidentity.org/> (archive) *Original artist:* Yukihiro Matsumoto, Ruby Visual Identity Team
- **File:Wikipedia-logo.png** *Source:* <https://upload.wikimedia.org/wikipedia/commons/6/63/Wikipedia-logo.png> *License:* GFDL *Contributors:* based on the first version of the Wikipedia logo, by Nohat. *Original artist:* version 1 by Nohat (concept by Paullusmagnus);

4.3 Content license

- Creative Commons Attribution-Share Alike 3.0