**About Ruby**

**Ruby** is a [dynamic](https://en.wikipedia.org/wiki/Dynamic_programming_language), [reflective](https://en.wikipedia.org/wiki/Reflection_%28computer_science%29), [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming_language), [general-purpose programming language](https://en.wikipedia.org/wiki/General-purpose_programming_language). It was designed and developed in the mid-1990s by [Yukihiro "Matz" Matsumoto](https://en.wikipedia.org/wiki/Yukihiro_Matsumoto) in [Japan](https://en.wikipedia.org/wiki/Japan).

|  |
| --- |
| Ruby is a simple programming language:   * **Chef uses Ruby as its reference language to define the patterns that are found in resources, recipes, and cookbooks** * **Use these patterns to configure, deploy, and manage nodes across the network**   Ruby is also a powerful and complete programming language:   * Use the Ruby programming language to make decisions about what should happen to specific resources and recipes * Extend Chef in any manner that your organization requires |

According to its creator, Ruby was influenced by [Perl](https://en.wikipedia.org/wiki/Perl), [Smalltalk](https://en.wikipedia.org/wiki/Smalltalk), [Eiffel](https://en.wikipedia.org/wiki/Eiffel_%28programming_language%29), [Ada](https://en.wikipedia.org/wiki/Ada_%28programming_language%29), and [Lisp](https://en.wikipedia.org/wiki/Lisp_%28programming_language%29).[[11]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-11) It supports multiple [programming paradigms](https://en.wikipedia.org/wiki/Programming_paradigm), including [functional](https://en.wikipedia.org/wiki/Functional_programming), [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming), and [imperative](https://en.wikipedia.org/wiki/Imperative_programming). It also has a [dynamic type](https://en.wikipedia.org/wiki/Dynamic_type) system and automatic [memory management](https://en.wikipedia.org/wiki/Memory_management).

**Table of versions**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Latest teeny version** | **Initial release date** | **End of support phase** | **End of security maintenance phase** |
| 1.8 | 1.8.7-p375 | 2003-08-04[[45]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-45) | 2012-06[[46]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-46) | 2014-07-01[[47]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-47) |
| 1.9 | 1.9.3-p551 | 2007-12-25[[49]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-49) | 2014-02-23 | 2015-02-23[[51]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-51) |
| 2.0 | 2.0.0-p648 | 2013-02-24[[53]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-53) | 2015-02-24[[52]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-Ruby_2.0.0-p648_Released-52) | 2016-02-24[[52]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-Ruby_2.0.0-p648_Released-52) |
| 2.1 | 2.1.10[[54]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-54) | 2013-12-25[[55]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-55) | 2016-03-30[[56]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-56)[[57]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-57) | 2017-03-30[[58]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-58) |
| 2.2 | 2.2.6[[59]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-59) | 2014-12-25[[60]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-60) | TBA | TBA |
| 2.3 | 2.3.3[[61]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-61) | 2015-12-25[[62]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-62) | TBA | TBA |
| **2.4** | 2.4.0[[43]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-:0-43) | 2016-12-25[[43]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-:0-43) | TBA | TBA |
| 3.0 |  | TBA[[63]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-63) |  |  |
|  | | | | |

**Features**

 Thoroughly [object-oriented](https://en.wikipedia.org/wiki/Object-oriented) with [inheritance](https://en.wikipedia.org/wiki/Inheritance_%28computer_science%29), [mixins](https://en.wikipedia.org/wiki/Mixin) and [metaclasses](https://en.wikipedia.org/wiki/Metaclass)[[68]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-stewart-68)

 [Dynamic typing](https://en.wikipedia.org/wiki/Dynamic_typing" \o "Dynamic typing) and [duck typing](https://en.wikipedia.org/wiki/Duck_typing)

 Everything is an [expression](https://en.wikipedia.org/wiki/Expression_%28programming%29) (even [statements](https://en.wikipedia.org/wiki/Statement_%28programming%29)) and everything is executed [imperatively](https://en.wikipedia.org/wiki/Imperative_programming) (even [declarations](https://en.wikipedia.org/wiki/Declaration_%28computer_science%29))

 Succinct and flexible syntax[[69]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-venners-productivity-69) that minimizes [syntactic noise](https://en.wikipedia.org/w/index.php?title=Syntactic_noise&action=edit&redlink=1) and serves as a foundation for [domain-specific languages](https://en.wikipedia.org/wiki/Domain-specific_languages)[[70]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-fowler-dsl-70)

 Dynamic [reflection](https://en.wikipedia.org/wiki/Reflection_%28computer_science%29) and [alteration](https://en.wikipedia.org/wiki/Dynamic_programming_language#Object_runtime_alteration) of objects to facilitate [metaprogramming](https://en.wikipedia.org/wiki/Metaprogramming)[[71]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-codeproject-dynamic-71)

 [Lexical closures](https://en.wikipedia.org/wiki/Closure_%28computer_science%29), [iterators](https://en.wikipedia.org/wiki/Iterator) and [generators](https://en.wikipedia.org/wiki/Generator_%28computer_science%29), with a unique [block syntax](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#Blocks_and_iterators)[[72]](https://en.wikipedia.org/wiki/Ruby_%28programming_language%29#cite_note-venners-closures-72)

 Literal notation for [arrays](https://en.wikipedia.org/wiki/Dynamic_array), [hashes](https://en.wikipedia.org/wiki/Associative_array), [regular expressions](https://en.wikipedia.org/wiki/Regular_expression) and [symbols](https://en.wikipedia.org/wiki/Symbol_%28Lisp%29)

 Embedding code in strings ([interpolation](https://en.wikipedia.org/wiki/Variable_interpolation))

 [Default arguments](https://en.wikipedia.org/wiki/Default_argument" \o "Default argument)

 Four levels of variable scope ([global](https://en.wikipedia.org/wiki/Global_variable), [class](https://en.wikipedia.org/wiki/Class_variable), [instance](https://en.wikipedia.org/wiki/Instance_variable), and [local](https://en.wikipedia.org/wiki/Local_variable)) denoted by [sigils](https://en.wikipedia.org/wiki/Sigil_%28computer_programming%29) or the lack thereof

 [Garbage collection](https://en.wikipedia.org/wiki/Garbage_collection_%28computer_science%29" \o "Garbage collection (computer science))

 [First-class continuations](https://en.wikipedia.org/wiki/First-class_continuation" \o "First-class continuation)

 Strict boolean [coercion](https://en.wikipedia.org/wiki/Implicit_type_conversion) rules (everything is *true* except false and [nil](https://en.wikipedia.org/wiki/Null_pointer))

 [Exception handling](https://en.wikipedia.org/wiki/Exception_handling" \o "Exception handling)

 [Operator overloading](https://en.wikipedia.org/wiki/Operator_overloading" \o "Operator overloading)

 Built-in support for [rational numbers](https://en.wikipedia.org/wiki/Rational_number), [complex numbers](https://en.wikipedia.org/wiki/Complex_number) and [arbitrary-precision arithmetic](https://en.wikipedia.org/wiki/Arbitrary-precision_arithmetic)

 Custom dispatch behavior (through method\_missing and const\_missing)

 Native [threads](https://en.wikipedia.org/wiki/Thread_%28computer_science%29) and cooperative [fibers](https://en.wikipedia.org/wiki/Fiber_%28computer_science%29) (fibers are a 1.9/[YARV](https://en.wikipedia.org/wiki/YARV) feature)