

Ruby là một ngôn ngữ lập trình hướng đối tượng, có khả năng phản ứng. Theo tác giả, Ruby chịu ảnh hưởng bởi Perl, Smalltalk, Eiffel, Ada và Lisp. Ruby là ngôn ngữ thông dịch và hướng đối tượng.[11]

Ruby cung cấp nhiều mẫu hình lập trình, bao gồm lập trình hàm, hướng đối tượng, mệnh lệnh, phản xạ. Nó sử dụng hệ thống kiểu biến động và tự động quản lý bộ nhớ tự động.

## Lịch sử

Ruby được tạo ra từ 24 tháng 2 năm 1993, bởi lập trình viên người Nhật Bản tên là Matsumoto Yukihiro (松本 行弘), biệt danh Matz. Phiên bản chính thức 0.95 (Đây là phiên bản công khai đầu tiên của Ruby được công bố tại Nhật Bản) vào ngày 21 tháng 12 năm 1995. Ở phiên bản Ruby 0.95, nó đã có nhiều tính năng quen thuộc trong các phiên bản sau của Ruby, bao gồm thiết kế hướng đối tượng, các lớp với kế thừa, mixin, vòng lặp, xử lý ngoại lệ và thu gom rác.

Sau khi phát hành Ruby 0.95 vào năm 1995, một số phiên bản ổn định của Ruby đã được phát hành trong những năm tiếp theo:

Ruby 1.0: ngày 25 tháng 12 năm 1996.

Ruby 1.2: tháng 12 năm 1998

Ruby 1.4: tháng 8 năm 1999.

Ruby 1.6: tháng 9 năm 2000.

Tháng 9 năm 2005, phiên bản ổn định mới nhất là 1.8.3. Ruby 1.9 (với một số thay đổi) cũng đang trong quá trình phát triển.

## Ruby 1.9

Ruby 1.9 được phát hành vào ngày Giáng sinh năm 2007. Hiệu quả với Ruby 1.9.3, phát hành ngày 31 tháng 10 năm 2011, Ruby chuyển từ việc được cấp phép kép theo Giấy phép Ruby và GPL để được cấp phép kép theo Giấy phép Ruby và giấy phép BSD hai điều khoản. Việc chuyển đổi sang phiên bản 1.9 từ 1.8 đã bị chậm lại do nhiều gems phổ biến cần được viết lại.

Ruby 1.9 giới thiệu nhiều thay đổi đáng kể so với các phiên bản 1.8. Ví dụ:

Giới hạn vùng tham chiếu của các biến cục bộ (các biến cục bộ chỉ được sử dụng ở trong các block mà chúng được khai báo)

Bổ sung cú pháp lambda:

```
f = ->(a,b) { puts a + b }
```

Bổ sung cú pháp Hash, sử dụng dấu hai chấm cho các khóa biểu tượng (symbol key):

```
{symbol_key: "value"} == {:symbol_key => "value"}
```

Hỗ trợ mã hóa ký tự theo chuỗi.

API socket mới (hỗ trợ IPv6)

require\_relative nhập khẩu an ninh

**Ruby 1.9 đã lỗi thời kể từ ngày 23 tháng 2 năm 2015, [32] và nó sẽ không còn nhận được bản sửa lỗi và bảo mật nữa. Người dùng nên nâng cấp lên phiên bản mới hơn.**

## Cái tên Ruby

Nguồn gốc của cái tên "Ruby" là từ một phiên chat online giữa Matsumoto và Ishitsuka Keiji vào ngày 24 tháng 2 năm 1993, trước khi bắt kỳ đoạn mã nào được viết cho ngôn ngữ này. Ban đầu, "Coral" và "Ruby" là hai cái tên được đề xuất. Matsumoto chọn tên sau trong email gửi đến Ishitsuka. Sau này Matz cũng bất ngờ khi phát hiện ra Pearl là viên đá quý tượng trưng cho những người sinh tháng 6, còn Ruby thì tượng trưng cho những người sinh tháng 7. Anh cho rằng cái tên Ruby như thế là phù hợp vì Ruby kế thừa và phát triển nhiều đặc tính từ Perl [12][13].

## Quan điểm

Quan điểm chính trong việc thiết kế của Matz là nhằm giảm thiểu các công việc nhàm chán mà họ, các nhà lập trình, buộc phải làm; tiếp đến là nguyên tắc thiết kế giao diện người dùng (user interface) hiệu quả.[14] Ông nhấn mạnh rằng việc thiết kế hệ thống cần phải tập trung vào con người, hơn là vào máy tính [15]:

Often people, especially computer engineers, focus on the machines. They think, "By doing this, the machine will run faster. By doing this, the machine will run more effectively. By doing this, the machine will something something something." They are focusing on machines. But in fact we need to focus on humans, on how humans care about doing programming or operating the application of the machines. We are the masters. They are the slaves.

Ngôn ngữ Ruby được thiết kế nhằm theo nguyên tắc ít gây ngạc nhiên nhất (principle of least surprise-POLS), nghĩa là ngôn ngữ hoạt động theo một cách trực quan hay ít nhất đó cũng là nhận xét mà các nhà lập trình đưa ra. Nguyên tắc này không xuất phát từ Matz và, nói chung, Ruby gần với suy nghĩ ít ngạc nhiên nhất của 'Matz' hơn.

## Ngữ nghĩa

Ruby là một ngôn ngữ hướng đối tượng: mỗi giá trị đều là một đối tượng (object), bao gồm các kiểu dữ liệu mà đối với các ngôn ngữ khác, chúng là kiểu cơ bản (primitive) như integer. Mỗi hàm (function) là một phương thức (method). Tên biến (variables) chính là tham chiếu (references) đến các đối tượng, bản thân nó không phải là đối tượng. Ruby hỗ trợ kế thừa (inheritance) với dynamic dispatch, mixin và singleton method (thuộc về, và để định nghĩa cho, một instance đơn hơn là định nghĩa dành cho lớp). Mặc dù Ruby không hỗ trợ đa kế thừa, các lớp vẫn có thể được đưa vào các module dưới dạng các mixins. Cú pháp dạng thủ tục (procedural syntax) vẫn còn được hỗ trợ, có vẻ như là ngoài tầm vực của mọi đối tượng, nhưng thực sự là thuộc một thể hiện của class Object tên là 'main'. Vì class này là cha của mọi class khác, nó trở nên ẩn đối với mọi lớp và đối tượng.

Ruby được xem là một ngôn ngữ lập trình đa mẫu hình (multi-paradigm programming language): nó cho phép bạn lập trình dạng thủ tục (tạo ra các hàm/biến nằm ngoài phạm vi của các lớp và biến chúng thành một phần của đối tượng gốc, 'self' Object), với khả năng hướng đối tượng (mọi thứ đều là đối tượng) hay hàm (nó có các hàm không có tên (anonymous functions), closures, và continuations; mọi câu lệnh đều có giá trị trả về, và các hàm đều trả về kết quả ước lượng cuối cùng). Nó hỗ trợ mạnh mẽ cho tự định kiểu (type introspection), reflection và meta-programming.

Theo Ruby FAQ, "Nếu bạn thích Perl, bạn sẽ thích Ruby và sẽ thấy thoải mái với cú pháp của nó. Nếu bạn thích Smalltalk, bạn sẽ thích Ruby và sẽ thấy thoải mái với ngữ nghĩa của nó (semantics). Nếu bạn thích Python, bạn có thể hoặc không thể dừng lại bởi sự khác biệt lớn trong triết lý hiện thực giữa Python và Ruby/Perl."

Trình thông dịch

Ruby có hai bản thông dịch chính: bộ thông dịch Ruby ban đầu (viết tắt là MRI), bản được dùng phổ biến nhất, và JRuby, bộ thông dịch dựa trên ngôn ngữ Java. Bộ thông dịch Ruby đã được cài đặt trên nhiều nền tảng khác nhau, bao gồm Unix, Microsoft Windows, DOS, Mac OS X, OS/2, Amiga và một số nền tảng khác. Bản chính thức của Ruby có kèm theo "IRB", là bộ thông dịch dạng dòng lệnh trực tiếp (interactive command-line interpreter) giúp cho việc kiểm tra code nhanh chóng.

Quy ước bản quyền

Ruby được phân phối tự do theo đăng ký tự do và mã nguồn mở GPL và Ruby License [1].

Tính năng

Hướng đối tượng

Xử lý ngoại lệ

Iterator và closures (dựa vào cách truyền một khối lệnh)

Native, biểu thức chính quy như Perl ở mức độ ngôn ngữ

Chồng toán tử (operator overloading)

Thu hồi rác tự động

Tính khả chuyển cao (portable)

Có thể chạy đa luồng (multi-threading) trên mọi platforms

Khả năng nạp động các DLL/thư viện chia sẻ trên hầu hết mọi platforms.

Tự kiểm tra (introspection), reflection và meta-programming

Rất nhiều thư viện chuẩn

Hỗ trợ dependency injection

Continuation và generator (xem ví dụ tại RubyGarden: continuations Lưu trữ 2005-11-26 tại Wayback Machine và generators Lưu trữ 2005-11-26 tại Wayback Machine)

Dùng khối lệnh (code block)

Ruby hiện tại vẫn chưa hỗ trợ Unicode, dù chỉ mới hỗ trợ tạm thời cho UTF-8.

[1] Click text view file

[2] Click text view file

[3] Click text view file