
espace

Introduction to Ruby

Yasmine M. Gaber

Why Ruby?

- Generic, interpreted, garbage collected
- Optimized for programmer productivity and happiness
- Concise but readable and clean syntax
- Pure OO - everything is an object (including primitives, classes, and nil)
- Supports procedural and functional programming styles
- Strong Dynamic typing / Duck Typing
- Open Source (GPL or “The Ruby License”)

Naming Conventions

- **Identifier Names**

  MyClass, MyModule

  MY_CONSTANT = 3.14

  my_method

  local_variable = 3.14

  @instance_variable

  @@class_variable

  $global_variable

- **Method Name suffixes**

  dangerous_method!

  query_method?

  setter_method=

Datatypes

- Numeric
- String
- Array
- Hash
- Symbol
- Range
- Regexp
- Struct

Control Structures

- Conditionals
  - *If, elsif, else, end*
  - *? operator*
- Loops
  - *For .. in*
  - *While and until*
- Exceptions
  - *raise .. rescue*

Variables and Constants

- Variable scope
  - global variables ($)
  - class variables (@@)
  - instance variables (@)
  - local variables
- Constants

Variables and Constants

- Globals
  - ARGV (command line args)
  - ENV (environment variables)
  - RUBY_VERSION
  - STDIN ($stdin)
  - STDOUT ($stdout)
  - $$ ($PROCESS_ID)
  - $? (exit status of process)
  - $: ($LOAD_PATH)

Boolean Expressions & Assignments

- Truth vs false
- nil
- Equality
  - Object#equal?
  - == operator
- Boolean operators

Boolean Expressions & Assignments

- Assignment idioms

```
a, b = b, a
a = 1; b = 1
a = b = 1
a += 1
a, b = [1, 2]
a = b || c
a ||= b
```

Classes and Objects

- Class inheritance
- Getter and setter methods
- *attr_accessor*
- The current object (*self*)
- Duck typing
- Namespaces
- Modules vs classes

Classes and Objects

- Method definitions
- Method visibility
- Singleton methods
- Class methods

Blocks and Proc Objects

- Blocks
- Proc objects
- Lambdas

Reflection and Meta Programming

- Introspection
  - class, superclass, ancestors, instance_of?, is_a?
  - methods, instance_methods, respond_to?
- Dynamic method dispatch
- Evaluating code
- Defining methods dynamically

Resources

- http://ruby-doc.org/
- https://github.com/peter/ruby_basic

Thank You