Ruby

Learn ruby basics

Agenda

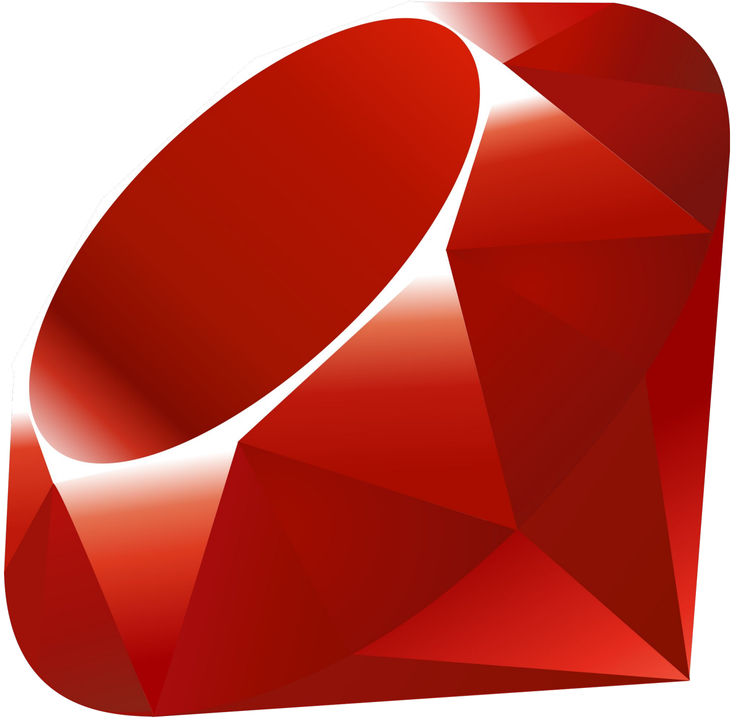
Intro

Advantages and disadvantages

Exercise

Go further

Questions



Intro

Dynamic	`x = 3; x = ‘3’`
Reflective	https://goo.gl/vDhFmK
Object-oriented	`class Klass; end`
General-purpose	
Open source	
Perl, Smalltalk, Eiffel, Ada, and Lisp	

Advantages

Easy to read and write (happy programmer)

Flexible

Everything is an object (`5.times { print "We *love* Ruby" }`)

Metaprogramming

Mixins

More info at https://www.ruby-lang.org/en/about

Disadvantages

Performance

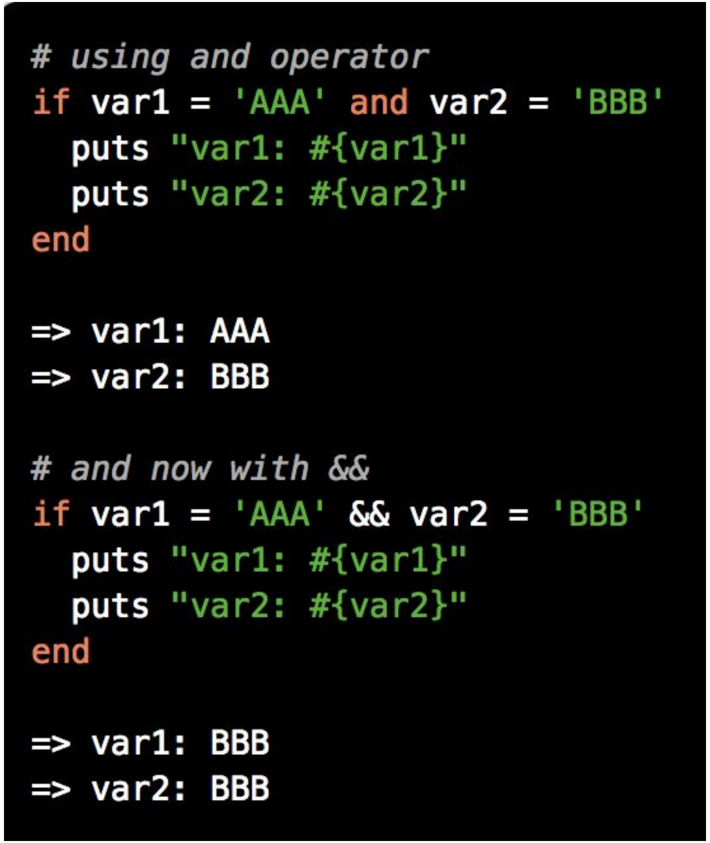
No max limit in gems use

Redefine core methods

Syntactic sugar

Threads

Global variables



```
# using and operator
if var1 = 'AAA' and var2 = 'BBB'
  puts "var1: #{var1}"
  puts "var2: #{var2}"
end

=> var1: AAA
=> var2: BBB

# and now with &&
if var1 = 'AAA' && var2 = 'BBB'
  puts "var1: #{var1}"
  puts "var2: #{var2}"
end

=> var1: BBB
=> var2: BBB
```

Exercise

Analyze https://pastebin.com/uxraMA7r

```
$ brew install sdl2

$ gem install gosu

$ git clone https://github.com/yovasx2/cars
```

Complete EnemyGenerator class in “exercise” branch

Gosu example https://goo.gl/s4yxil

Go further

- Have a quick reference https://goo.gl/xhd2gp | https://goo.gl/ptc59q
- Try ruby at http://tryruby.org
- Define your style https://github.com/airbnb/ruby
- Look the API http://ruby-doc.org
- Use and enhance already developed libraries https://rubygems.org
- Read a book http://ruby-doc.com/docs/ProgrammingRuby
- Take a course https://www.codeschool.com/learn/ruby
- Be supported by community http://rubyconferences.org