

CS50 For MBA's: Programming with Ruby

Robert Krabek
Spring 2016

What is Ruby?

- First created in 1995 by Yukihiro Matsumoto
- Object-oriented, general purpose programming language
- Designed for programmer happiness
- Follows principles of good UI design

Why program in Ruby?

- It's very easy to learn, about as close to English as you're going to get
- Literally designed for programmer happiness
- You need it for Rails, the go to tool for rapid web development
- Slow compared to compiled languages but getting faster, is now a more mature language with an established community supporting it
- Completely open source, unlike .NET allowing for more collaboration
- Tons of libraries (Ruby gems) to leverage

Getting started

Check installed version of Ruby

```
$ ruby -v
```

Else see <https://www.ruby-lang.org/en/documentation/installation/> for instructions

IRB (Interactive RuBy shell) because all ruby files end with .rb

```
$ irb
```

This will take you into a ruby programming environment or execute a ruby file with:

```
$ ruby <filename>.rb
```

Ruby Syntax

Declare variables with no type, no semicolons (yay!)

No parentheses, put 'end' after a block is finished

Only += and -= no ++ or --

require 'some_library' instead of #include or <script src=>

Variable sized arrays, you can append with <<

No chars, just 1 letter strings

File I/O in Ruby

Similar to C or Python

```
fname = "description.txt"
```

```
file = File.open(fname, "w")
```

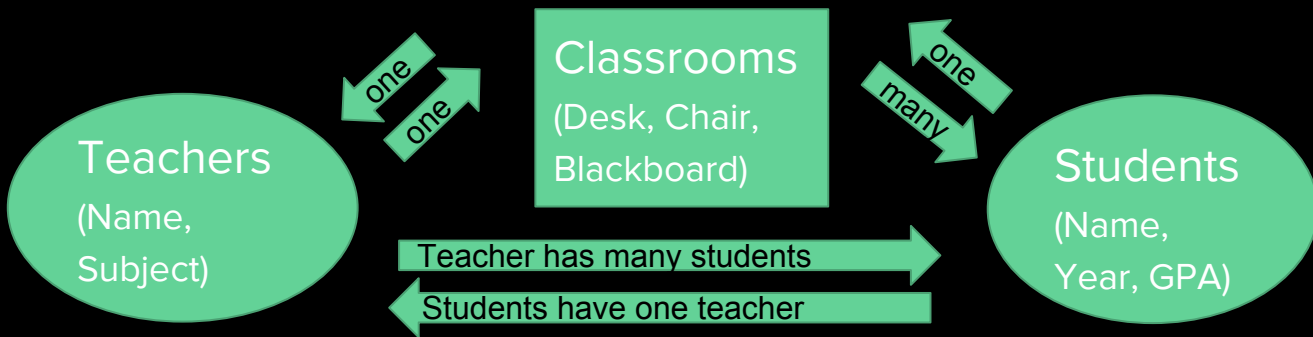
```
file.puts description
```

```
file.close
```

Can also use csv and json libraries to output csv and json files

Object-oriented Ruby

- Designed as genuine object-oriented language
- Everything is an object, lone functions are actually methods of the Object class
- Objects are defined with properties and relationships to other objects
- Example of OOP paradigm: design an elementary school



- A bit about Inheritance: what if you wanted to design a private elementary school?

Is OOP the Only Option?

- No! Although Ruby is specifically developed with an OOP design it can be used in other programming paradigms as well although it is not designed to be.
- Functional programming generally defined by:
 - First class functions (functions can be passed as parameters, returned from other functions etc.)
 - Immutable data (can't modify existing data, only return new data)
 - But most importantly, NO SIDE EFFECTS

Ruby on Rails

- Ruby work for non-rails web development is an edge case (maybe Sinatra)
- Framework (what does that even mean?)
 - Default structure for database, web service, web pages
- Convention over Configuration
 - There's a lot of "magic" in rails. Stick to the structure and build Twitter in a weekend... otherwise you may find yourself in trouble.

Sites using RoR

- AirBnB
- GitHub
- Groupon
- Twitter (At first)
- Bloomberg
- Hulu
- Kickstarter
- Codecademy

To Ruby or not to Ruby?

Yes

- Quick development for web applications
- Quick to learn and get established in codebase
- Programmer happiness
- Lively rails community

No

- Need memory management capabilities
- Need super fast performance or large scaling
- Need to develop on Windows
- If you're trying to be 'cool'

Resources

Rails Documentation - <http://rubyonrails.org/documentation/>

RoR Seminar - https://manual.cs50.net/seminars/#ruby_on_rails

Ruby Documentation - <http://ruby-doc.org/>

Codecademy & HackerRank

Gem Collection - <https://www.ruby-toolbox.com/>