

Ruby Explorations 0

Mark Keane...CSI...UCD

Mixed delivery



This Week's Goals

- To have all the software properly installed
...and running properly
- To agree a standard build (sort of)
- To have a (fairly) standard environment
- Understand difference between **ruby** and **irb**



Learning by Doing...

- Lectures are available on Moodle; work through at your own pace but with open laptop
- Practical Clinics (4 over the semester, for guidance, see timetable) with very open laptop
- Practical work may be self-paced but would advise you to stick to the recommended submission dates with submissions via moodle (do not fall behind)
- Practical is like a drop-in clinic; MTK and handsome pals are available for advice

Assessment: Exam

Examined by:

Formal Written Exam (100%)

Coursework:

There are 10 weekly practical pieces of work to do

Marked as pass/fail with feedback

The do **NOT** contribute to assessment but are there to guide and aid you in learning the language

A Word on Plagiarism

Plagiarism is NOT an option

Important to help each other and use the group for advice but that is not the same as copying

Remember, copyier and copyee are treated equally

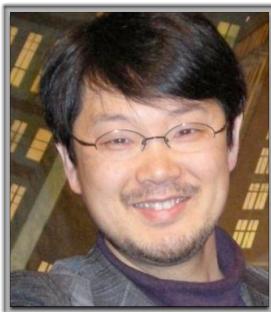
Outcomes vary, from zeroing of assignments up to expulsion from course

Ruby History

created by Yukihiro Matsumoto in 1993

a language balancing functional and imperative programming

Tubular Bells of programming...



Flanagan, D & Matsumoto, Y. (2008).
The Ruby Programming Language. O'Reilly.

Installation Summary

Summary Ruby



<http://www.ruby-lang.org/en/downloads/> for all directions for Windows, MacOSX (bundled or use macports, nb XCode) and Linux (see Thomas , 2009, chapter 1)

we will all use **Ruby 2.3.x**

basically, use a text editor to write files, a cshell to run files and navigate directories (or ruby2.3 if multiples)

ruby filename.rb [runs the file called *filename*]

irb [gets you the interactive version]

<http://www.ruby-doc.org/core/>

Now Let's Do it in Steps...

Ruby Downloads (PC)

<http://rubyforge.org/>

Please use 2.3 (for overall compatibility)

<http://www.ruby-lang.org/en/downloads/>
General directions for 2.3

<http://rubyinstaller.org/>

One-click installer for 2.3

NB: There are 32-bit and 64-bit versions

A screenshot of the Ruby website. The main navigation bar includes 'Downloads', 'Documentation', 'Libraries', 'Community', 'News', 'Security', and 'About Ruby'. Below the navigation, there's a section titled 'Ruby is...' with a brief description of Ruby as a dynamic, open source programming language. A code snippet shows how to iterate over an array. To the right, there's a 'Download Ruby' button and a 'Read More...' link. At the bottom, there's a 'RubyConf Taiwan 2015 Registration Online' link and a 'Get Started, it's easy!' section with links to 'Try Ruby! in your browser', 'Ruby in Twenty Minutes', and 'Ruby from Other Languages'.

The easy way to install Ruby on Windows

This is a self-contained Windows-based installer that includes the Ruby language, an execution environment, important documentation, and more.

Download **Add-ons**

Latest News

- RubyInstaller 2.1.9, 2.2.5, and 2.3.1 released
- RubyInstaller 2.0.0-p648, 2.1.8 and 2.2.4 released
- RubyInstaller 2.0.0-p645, 2.1.6 and 2.2.2 released

Extras

- Online Ruby Programming Course
- Online Rails Programming Course

Download Ruby

Here you can get the latest Ruby distributions in your favorite flavor. The current stable version is 2.3.1. Please be sure to read Ruby's License.

Ways of Installing Ruby

- On Linux/UNIX, you can use the package management system of your distribution or third-party tools (rbenv and RVM).
- On OS X machines, you can use third-party tools (rbenv and RVM).
- On Windows machines, you can use RubyInstaller.

See the Installation page for details on using package management systems or third-party tools.

Of course, you can also install Ruby from source on all major platforms.

Compiling Ruby - Source Code

Installing from the source code is a great solution for when you are comfortable enough with your platform and perhaps need specific settings for your environment. It's also a good solution in the event that there are no other packages available for your platform.

See the Installation page for details on building Ruby from source. If you have an issue compiling Ruby, consider using one of the third party tools mentioned above. They may help you.

Stable releases:

- Ruby 2.3.1
2016-07-26
ba7f738cb0232b44920fb8e3864dc5cfaeae9d89d8523ce0736945c...
97165433
- Ruby 2.2.5
2016-06-27
5a242a24...

Ruby Downloads (Mac)

<http://www.ruby-lang.org/en/downloads/>

Mac OSX Lions came with 1.8.7

Mac OSX Sierra came with 2.0.0

MUST use Ruby 2.3 (was last most stable...)

See Handout on details

NB; RAILS maps differently onto different RUBY versions

Mammy, what have I done ?

Checking Versions

Open a command-line window and check !

```
Last login: Tue Aug 28 15:13:36 on ttys000
Marks-MacBook-Pro:~ mkeane$ ruby -v
ruby 2.3.1p112 (2016-04-26 revision 54768) [x86_64-darwin15]
Marks-MacBook-Pro:~ mkeane$ ruby2.0 -v
ruby 2.0.0p481 (2013-06-27 revision 41674) [x86_64-darwin12]
*** if old version is still there ****
Marks-MacBook-Pro:~ mkeane$ gem -v
2.6.7
```

Downloads (Gems)

Ruby comes with libraries, called Ruby Gems

A standard gem package will come with most installs

Hence:

```
[Mouseking6-3:] markkean$ gem -v
2.6.7
```

You can look at your collection of gems using `gem list`

```
*** LOCAL GEMS ***
abstract (1.0.0)
actionmailer (3.1.0, 3.0.1)
actionpack (3.1.0, 3.0.1)
activemodel (3.1.0, 3.0.1)
active record (3.1.0, 3.0.1, 2.3.3)
activerecord (3.1.0, 3.0.1, 2.3.3)
addressable (2.2.6)
ansi (1.3.0)
archive-tar-minitar (0.5.2)
arcl (2.2.1, 1.0.1)
arcl (2.2.1, 1.0.1)
builder (3.0.0, 2.1.2)
bundler (1.0.18)
capistrano (2.9.0)
coffee-rails (3.1.1)
coffee-script (2.2.0)
coffee-script-source (1.1.2)
columnize (0.3.4)
distance_measures (0.0.6)
em-http-request (1.0.0)
em-socketify (0.1.0)
erubis (2.7.0, 2.6.6)
eventmachine (1.0.0.beta.4, 0.12.10)
execjs (0.9)
highline (1.6.2)
hike (1.2.1)
http_parser.rb (0.5.2)
i18n (0.6.0, 0.4.2)
jquery-rails (1.0.14)
json (1.1.6)
k (0.0.7)
linecache19 (0.5.12)
mail (2.3.0, 2.2.9)
mime-types (1.16)
minitest (2.6.0, 1.6.0)
multi_json (1.0.3)
net-scp (1.0.4)
net-ssh (2.0.5)
net-ssh (2.2.1)
```

do this...

```
Marks-MacBook-Pro:~ mkeane$ gem update
Updating installed gems
Updating actionmailer
ERROR: While executing gem ... (Gem::FilePermissionError)
  You don't have write permissions into the /opt/local/lib/
  ruby/1.9/gems/1.9.1 directory.
Marks-MacBook-Pro:~ mkeane$ sudo gem update
To proceed, enter your password, or type Ctrl-C to abort.

Password:
Updating installed gems
Updating actionmailer
Successfully installed activesupport-3.2.8
Successfully installed activemodel-3.2.8
...

```

defaults to
most recent

A World of Pain (Gems)

Gems libraries may be stored in different places

Mac OSX (unix) versions and gems may be stored in local directories or "higher" system applications dirs

Sometimes, you load a gem, you list it but then when your try to use it says "FU, Gem does not exist..."

Problem can often be gems stored in different places; can be very frustrating if occurs (avoid by clean installs)



will
cause
bugs

Geek Crap I: Search Paths

Move along, there is nothing to see here....

Ruby uses many environmental variables to store search paths and the like... under the bonnet

```
MacBook-Air-2:~ user$ ruby -e 'puts $LOAD_PATH'
/opt/local/lib/ruby2.3/gems/2.3.0/gems/did_you_mean-1.0.0/lib
/opt/local/lib/ruby2.3/site_ruby/2.3.0
/opt/local/lib/ruby2.3/site_ruby/2.3.0/x86_64-darwin15
/opt/local/lib/ruby2.3/vendor_ruby
/opt/local/lib/ruby2.3/vendor_ruby/2.3.0
/opt/local/lib/ruby2.3/vendor_ruby/2.3.0/x86_64-darwin15
/opt/local/lib/ruby2.3/vendor_ruby
/opt/local/lib/ruby2.3/2.3.0
/opt/local/lib/ruby2.3/2.3.0/x86_64-darwin15
```

Geek Crap II: Search Paths

BUT...if I check my other version of Ruby

```
Marks-MacBook-Pro:~ mkeane$ ruby1.8 -e 'puts $LOAD_PATH'
/Library/Ruby/Site/1.8
/Library/Ruby/Site/1.8/powerpc-darwin12.0
/Library/Ruby/Site/1.8/universal-darwin12.0
/Library/Ruby/Site
/System/Library/Frameworks/Ruby.framework/Versions/1.8/usr/lib/ruby/vendor_ruby/1.8
/System/Library/Frameworks/Ruby.framework/Versions/1.8/usr/lib/ruby/vendor_ruby/1.8/universal-darwin12.0
/System/Library/Frameworks/Ruby.framework/Versions/1.8/usr/lib/ruby/vendor_ruby
/System/Library/Frameworks/Ruby.framework/Versions/1.8/usr/lib/ruby/1.8
/System/Library/Frameworks/Ruby.framework/Versions/1.8/usr/lib/ruby/1.8/powerpc-darwin12.0
/System/Library/Frameworks/Ruby.framework/Versions/1.8/usr/lib/ruby/1.8/universal-darwin12.0
.
```

YOU...



Are now ready to go...

Set Up Your Environment

We support the ?bear/bare? minimum:

Command-line window terminal

Text editor of your choice (e.g., Textwrangler, PC?)

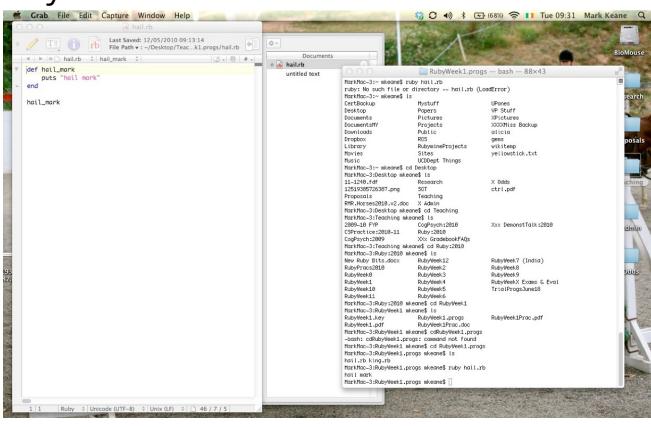
Fancier options exist:

Xcode, RubyMine for Macs

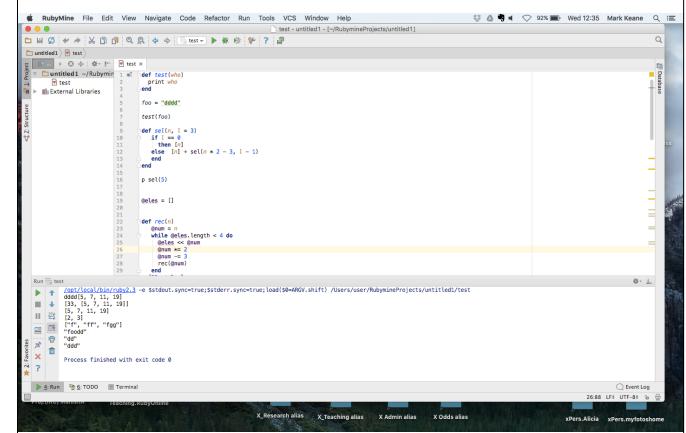
RDE Ruby IDE, RubyMine for PCs

But, they are your own business; start with basics

My Old Environment



My Environment (fancier)



Now, lets use it !

Open a file...

Write this in a simple text file, called **hail.rb**:

```
def hail_mark  
  puts "hail mark"  
end
```

hail_mark *hail.rb*

Open your Terminal...

Move to the directory in which the file sits...

```
markkean$ cd RubyWeek1  
  
markkean$ ls  
RubyWeek1.key      RubyWeek1.progs      RubyWeek1Prac.pdf  
RubyWeek1.pdf      RubyWeek1Prac.doc  
  
markkean$ cd RubyWeek1.progs  
  
markkean$ ls  
hail.rb      king.rb  
  
markkean$ more hail.rb  
def hail_mark  
  puts "hail mark"  
end  
  
hail_mark  
  
markkean$ ruby hail.rb  
hail mark  
  
markkean$
```

Well done !



Running Ruby Two Ways

Ruby can be used in two ways...

Invoke the ruby interpreter with **ruby** command

Invoke a ruby shell with **irb** command, to be interactive

Running a file with ***ruby*** (we have just seen)

```
markkean$ ruby hail.rb  
hail mark
```

output when file is executed by ruby
Interactively, you use ***irb*** and the type in expressions
(see next page)

Running Ruby The 2nd Way

Call the ruby shell using **irb**:

```
markkean$ irb
>>> "foo"
=> "foo"
>> puts "hail mark"
hail mark
=> nil
>>> def hail_mark
>> puts "hail mark"
>>> end
=> nil
>>> hail_mark
hail mark
=> nil
>>> quit
markkean$
```

every expression returns a value

awkward way to define a method

after a method does its thing, it returns nil



Resources I: <http://www.ruby-doc.org/core/>

Resources I: <http://www.ruby-doc.org/core/>

[Home](#) [Classes](#) [Methods](#)

In Files

- [complex.c](#)
- [pack.c](#)
- [rational.c](#)
- [string.c](#)
- [transcode.c](#)

Parent

Object

Methods

- [::new](#)
- [::try_convert](#)
- [%>](#)
- [#+](#)
- [#+@](#)
- [#<<](#)
- [#<>](#)
- [#<=](#)
- [#>=](#)

String

A `String` object holds and manipulates an arbitrary sequence of bytes, typically representing characters. String objects may be created using `String::new` or as literals.

Because of aliasing issues, users of strings should be aware of the methods that modify the contents of a `String` object. Typically, methods with names ending in `"!"` modify their receiver, while those without a `"!"` return a new `String`. However, there are exceptions, such as `String#[]`.

Public Class Methods

- [new\(str=nil\) → new_str](#)
- [new\(str="", encoding: enc\) → new_str](#)
- [new\(str:"", capacity: size\) → new_str](#)

Returns a new string object containing a copy of str.

The optional enc argument specifies the encoding of the new string. If not specified, the encoding of str (or ASCII-8BIT, if str is not specified) is used.

The optional size argument specifies the size of internal buffer. This may improve performance.

ZenHub
Get more done in GitHub.

Stop wasting hours in JIRA. Get epic, task boards, and more inside GitHub.

ada via Carbon

Resources II: <http://www.ruby-lang.org/>

Ruby A PROGRAMMER'S BEST FRIEND

[Downloads](#) [Documentation](#) [Libraries](#) [Community](#) [News](#) [Security](#) [About Ruby](#)

Ruby is...

A dynamic, open source programming language with a focus on simplicity and productivity. It has an elegant syntax that is natural to read and easy to write.

[Download Ruby](#) or [Read More...](#)

Ruby knows what you # mean, even if you # want to do math on # an entire Array cities = [WfLondon Oslo Paris Amsterdam Berlin] visited = [Berlin Oslo]

puts "I still need " + "to visit the " + "following cities:", cities - visited

[Get Started, It's easy!](#)

Resources III: csmoodle.ucd.ie/moodle/

CS Moodle 2018-19

type ! click !

Search courses: Ruby 2018 Go

- > LEVEL 1
- > LEVEL 2
- > LEVEL 3
- > LEVEL 4
- > LEVEL 5
- > MSc
- > PhD
- > Sri Lanka
- > BDIC

Resources III: csmoodle.ucd.ie/moodle/

CS Moodle 2018-19

Search results: 1 click !

COMP47530 Exploring Ruby (2018-19: Mixed Delivery)

Lecturer: Mark Keane Mixed delivery self-paced Ruby course.

Category: MSc

Search courses: Ruby 2018 Go

Resources III: csmoodle.ucd.ie/moodle/

CSMOODLE.UCD.IE

Log in

Username: mkeane
Password:
 Remember username

Forgotten your username or password?
Cookies must be enabled in your browser

Some courses may allow guest access

Resources III: csmoodle.ucd.ie/moodle/

CSMOODLE.UCD.IE

Log in

Username: mkeane
Password:
 Remember username

Forgotten your username or password?
Cookies must be enabled in your browser

Some courses may allow guest access

Enrollment Key:
ruby18

Resources III: That's a Bingo !

COMP47530 Exploring Ruby (2018-19: Mixed Delivery)

10 September - 16 September
17 September - 23 September
24 September - 30 September
1 October - 7 October

Announcements

10 September - 16 September

Welcome to the first week of the Exploring Ruby course. Below you will see all the materials and handouts for this first week. Note that there are pdfs of the slides, along with an mp4 of them with a voice-over of the lecture. You can run this voiced-lecture here in Moodle or (if you are on the move) you can connect to a dropbox version of this (via a provided url). Hopefully, this all works well. The main aim of this week is to get ruby installed on your machine and to be ready to start doing your practical work by next week. All the best and happy hunting.

Books



Thomas, D., Fowler, C. & Hunt, A. (2009).
Programming Ruby 1.9. Pragmatic Bookshelf.

Flanagan, D & Matsumoto, Y. (2008). mainly for reference !
The Ruby Programming Language. O'Reilly.

Black, D.A. (2006). *Ruby for Rails*. Manning.

My Ruby: March 2017

```
mkeane$ gem list
*** LOCAL GEMS ***
abstract (1.0.0)
actionmailer (3.2.8, 3.1.0, 3.0.1)
actionpack (3.2.8, 3.1.0, 3.0.1)
activemodel (3.2.8, 3.1.0, 3.0.1)
activecord (3.2.8, 3.1.0, 3.0.1, 2.3.3)
activeresource (3.2.8, 3.1.0, 3.0.1)
activesupport (3.2.8, 3.1.0, 3.0.1, 2.3.3)
addressable (2.3.2, 2.2.6)
ansi (1.4.3, 1.3.0)
archive-tar-minitar (0.5.2)
arel (3.0.2, 2.2.1, 1.0.1)
bcrypt-ruby (3.0.1)
builder (3.0.0, 2.1.2)
bundler (1.1.5, 1.0.18)
capistrano (2.13.3, 2.9.0)
coffee-rails (3.2.2, 3.1.1)
coffee-script (2.2.0)
coffee-script-source (1.3.3, 1.1.2)
columnize (0.3.6, 0.3.4)
distance_measures (0.0.6)
xecjs (1.4.0, 1.2.9)
highline (1.6.14, 1.6.2)
hike (1.2.1)
http_parserrb (0.5.2)
i18n (0.6.0, 0.4.2)
journey (1.0.4)
```

```
json (1.6.1)
k_means (0.0.7)
linecache19 (0.5.12)
mail (2.4.4, 2.3.0, 2.2.9)
mime-types (1.19, 1.16)
minitest (3.0.0, 2.6.0, 1.6.0)
multi_json (1.3.6, 1.0.3)
net-scp (1.0.4)
net-sftp (2.0.5)
net-ssh (2.5.2, 2.2.1)
net-ssh-gateway (1.1.0)
nokogiri (1.5.0)
polyglot (0.2.0, 0.2, 0.3)
rake (1.1.1, 1.1.0, 1.0.3)
rack (1.2, 1.1, 1.0.3)
rack-cache (1.2, 1.1, 1.0.3)
rack-mount (0.8.3, 0.6.13)
rack-test (0.6.1, 0.5.6)
rails (3.2.8, 3.1.0, 3.0.1)
raities (3.2.8, 3.1.0, 3.0.1)
rake (0.9.2.2, 0.9.2, 0.8.7)
rdoc (4.2.2, 3.10, 3.9.4, 2.5.8)
ruby-debug-base19 (0.11.25,
0.11.25.02)
ruby-debug-ide (0.4.16)
ruby_core_source (0.1.5)
sass (3.2.1, 3.1.10)
sass-rails (3.2.5, 3.1.4)
sprockets (2.1.3, 2.0.2, 2.0.0)
sqrite3 (1.3.4)
sinatra (1.3.3, 1.3.2)
therubyracer (0.10.0, 0.14.6, 0.14.4)
tilt (1.3.3)
treelib (1.4.10, 1.4.8)
turn (0.9.6, 0.8.3)
tzinfo (0.3.33, 0.3.30, 0.3.29, 0.3.23)
uglifier (1.2.7, 1.0.3)
```

Ruby 2.3.1 (p112)

Gem 2.6.7 (see list)

irb 0.9.6