

Week 10

its the final countdown

Week 10 Topics

- Merb
- ruby debugger
- rails debugger

Merb

merb-tastic

Merb Sales Pitch

- Built For Speed
- Lightweight
- Powerful

Built For Speed

- ORM agnostic
- JavaScript library agnostic
- Template Language agnostic
- Thread safe

Lightweight

- Simple
- Well Organized
- Easy to Extend

Powerful

- No wheel reinvention
- Plugin architecture
- Uses RubyGems for plugins

Installation

```
sudo gem install merb
```

look at all those gems!

merb-gen app my_app

watch that stuff scroll!

```
cd my_app; merb
```

guess the port! (its 4000)

Directory Layout

```
.
|-- app
|   |-- controllers
|   |-- helpers
|   `-- views
|       |-- exceptions
|       `-- layout
|-- autotest
|-- config
|   `-- environments
|-- log
|-- public
|   |-- images
|   `-- stylesheets
`-- spec
```

14 directories

What is Missing?

```
.
|-- app
|   |-- controllers
|   |-- helpers
|   `-- views
|       |-- exceptions
|       `-- layout
|-- autotest
|-- config
|   `-- environments
|-- log
|-- public
|   |-- images
|   `-- stylesheets
`-- spec
```

14 directories

Controllers

control it.

merb-gen controller products

more files, yay!

merb-gen controller products

```
[master] [aaron@Jordan my_application]$ merb-gen controller products
exists    app
exists    app/controllers
exists    app/helpers
exists    app/views
create    app/views/products
create    app/controllers/products.rb
create    app/helpers/products_helper.rb
create    app/views/products/index.html.erb
dependency merb_controller_test
create    spec/controllers/
create    spec/helpers/
exists    spec
exists    spec/controllers
exists    spec/helpers
create    spec/controllers/products_spec.rb
create    spec/helpers/products_helper_spec.rb
```

Generated Controller

```
class Products < Application  
  
  def index  
    render  
  end  
  
end
```

View

- Defaults to ERb
- But Plugins may be added

Testing

make sure it works.

Merb

- defaults to rspec
- but plugins may be added

rake spec

Generated Test

```
require File.join(File.dirname(__FILE__), '..', 'spec_helper.rb')

describe Products, "index action" do
  before(:each) do
    dispatch_to(Products, :index)
  end
end
```

Adding an ORM

activerecord ftw!

```
sudo gem install merb_activerecord
```

change config/init.rb

```
# Merb doesn't come with database support by default. You need  
# an ORM plugin. Install one, and uncomment one of the following  
lines,  
# if you need a database.
```

```
# Uncomment for DataMapper ORM  
# use_orm :datamapper
```

```
# Uncomment for ActiveRecord ORM  
use_orm :activerecord
```

```
# Uncomment for Sequel ORM  
# use_orm :sequel
```

run merb

it will complain

merb-gen model product

merb-gen model product

```
[master][aaron@Jordan my_application]$ merb-gen model product
  exists  app
  create  app/models
  create  app/models/product.rb
dependency migration
  create  schema/migrations
  exists  schema
  exists  schema/migrations
  create  schema/migrations/001_product_migration.rb
dependency merb_model_test
  exists  spec
  create  spec/models
  create  spec/models/product_spec.rb
[master][aaron@Jordan my_application]$
```

Update the Migration

```
class ProductMigration < ActiveRecord::Migration
  def self.up
    create_table :products do |t|
      t.string :name
      t.string :description

      t.timestamps
    end
  end

  def self.down
    drop_table :products
  end
end
```

```
rake db:migrate
```

Model Testing

Pull it Together!

The Ruby Debugger

bugs! ack!

Ruby Debugger

- set breakpoints
- step over functions
- step into functions
- examine variables
- watch variables
- move in the stack

Running the Debugger

```
ruby -r debug some_program.rb
```

Breakpoints

- line
- method
- class

Setting a Breakpoint

```
b[reak] [file:|class:]<line|method>  
b[reak] [class.]<line|method>
```

Getting To your Breakpoint

```
c[ont] run until program ends or hit breakpoint
```

Stepping Over/In to A Line

```
s[tep][ nnn]  step (into methods) one line or till line nnn  
n[ext][ nnn] go over one line or till line nnn
```

List Your Variables

<code>v[ar]</code>	<code>g[lobal]</code>	<code>show global variables</code>
<code>v[ar]</code>	<code>l[ocal]</code>	<code>show local variables</code>
<code>v[ar]</code>	<code>i[nstance]</code> <code><object></code>	<code>show instance variables of object</code>
<code>v[ar]</code>	<code>c[onst]</code> <code><object></code>	<code>show constants of object</code>

Viewing Your Context

```
(rdb:1) list
[82, 91] in /Library/Ruby/Gems/1.8/gems/mechanize-0.7.5/lib/www/
mechanize.rb
  82     def initialize
  83         # attr_accessors
  84         @cookie_jar      = CookieJar.new
  85         @log              = nil
  86         @open_timeout    = nil
=> 87         @read_timeout   = nil
  88         @user_agent      = AGENT_ALIASES['Mechanize']
  89         @watch_for_set   = nil
  90         @ca_file         = nil # OpenSSL server certificate file
  91
(rdb:1)
```

The Rails Debugger

more bugs! ack!

```
gem install ruby-debug
```

add 'debugger' for
breakpoints