

Rails from the Ground Up!

with Jared Richardson
RoleModel Software

Sinatra

A Lightweight Server

Sinatra

[README](#)
[DOCUMENTATION](#)
[CONTRIBUTE](#)
[CODE](#)
[CREW](#)
[ABOUT](#)

Put this in
your pipe

```
require 'sinatra'  
  
get '/hi' do  
  "Hello World!"  
end
```

and smoke it

```
$ gem install sinatra
$ ruby -rubygems hi.rb
== Sinatra has taken the stage ...
>> Listening on 0.0.0.0:4567
```

Sinatra

Put this in
your pipe

and smoke it

README
DOCUMENTATION
CONTRIBUTE
CODE
CREW
ABOUT

require 'sinatra'

```
get '/hi' do
  "Hello World!"
end
```

```
$ gem install sinatra
$ ruby -rubygems hi.rb
== Sinatra has taken the stage ...
>> Listening on 0.0.0.0:4567
```



A Quick Aside

HTTP Methods

get

post

put

delete

Basic Application

```
# myapp.rb

require 'sinatra'

get '/' do
  'Hello world!'
end
```

Or . . .

```
# myapp.rb  
require 'sinatra'  
put '/' do  
  'Hello world!'  
end
```

Or Another

```
# myapp.rb  
require 'sinatra'  
post '/' do  
  'Hello world!'  
end
```

Yet Another

```
# myapp.rb  
require 'sinatra'  
delete '/' do  
  'Hello world!'  
end
```

Or Even

```
# myapp.rb

require 'sinatra'

get '/' do
  'Get the world!'
end

put '/' do
  'Put the world!'
end

post '/' do
  'Post the world!'
end

delete '/' do
  'Delete the
world!'
end
```

Chrome Plugins

DevHttpClient

Request Maker

But That Won't Work for You

gem install sinatra
but you did that yesterday :)

A Quick Aside

Ruby Gems

Package management

rubygems.org

Basic Commands

gem install <gem name>

gem install <gem name> -v <version>

gem search -r <gem name>

gem uninstall <gem>

gem update

gem update --system

RubyGems.org
your community gem host

Build and Share

gem build your_gem.gemspec

gem push your_gem-1.0.0.gem

<http://guides.rubygems.org/make-your-own-gem>

To Use a Gem

`require "<gem name>"`

Back to Sinatra

gem install sinatra

Basic Application

```
# myapp.rb

require 'sinatra'

get '/' do
  'Hello world!'
end
```

Sinatra Routes

```
get '/' do
  # show
end

post '/' do
  # create
end

put '/' do
  # replace
end

patch '/' do
  # modify
end

delete '/' do
  # remove
end

options '/' do
  # appease?
end
```

Route Order

First come/first serve

Params via the params variable

params

```
# to match "GET /hello/jay"  
# and "GET /hello/matt"  
# params[:name] is 'jay' or 'matt'  
  
get '/hello/:name' do  
  "Hello #{params[:name]}!"  
end
```

Your Turn

Write a time server ($t = \text{Time.now}$)

Re-use your tests

“Raw” time on /raw

HTML page on /

Month on /month

Next...

Moving to Rails...