

Classy Web Development with Ruby, Sinatra and Heroku

Get a site online in five minutes
or less.

But first...

serve

serve

Get a site online* in *5 seconds!*

serve

Get a site online* in *5 seconds!*

*localhost

```
rob$ serve
```

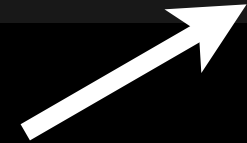
```
rob$ serve
```

```
[2010-03-23 11:35:38] INFO WEBrick 1.3.1
```

```
[2010-03-23 11:35:38] INFO ruby 1.8.7 (2008-08-11)
```

```
[2010-03-23 11:35:38] INFO Serve::Server#start: pid=63741 port=4000
```

```
rob$ serve  
[2010-03-23 11:35:38] INFO WEBrick 1.3.1  
[2010-03-23 11:35:38] INFO ruby 1.8.7 (2008-08-11)  
[2010-03-23 11:35:38] INFO Serve::Server#start: pid=63741 port=4000
```




```
rob$ serve  
[2010-03-23 11:35:38] INFO WEBrick 1.3.1  
[2010-03-23 11:35:38] INFO ruby 1.8.7 (2008-08-11)  
[2010-03-23 11:35:38] INFO Serve::Server#start: pid=63741 port=4000
```

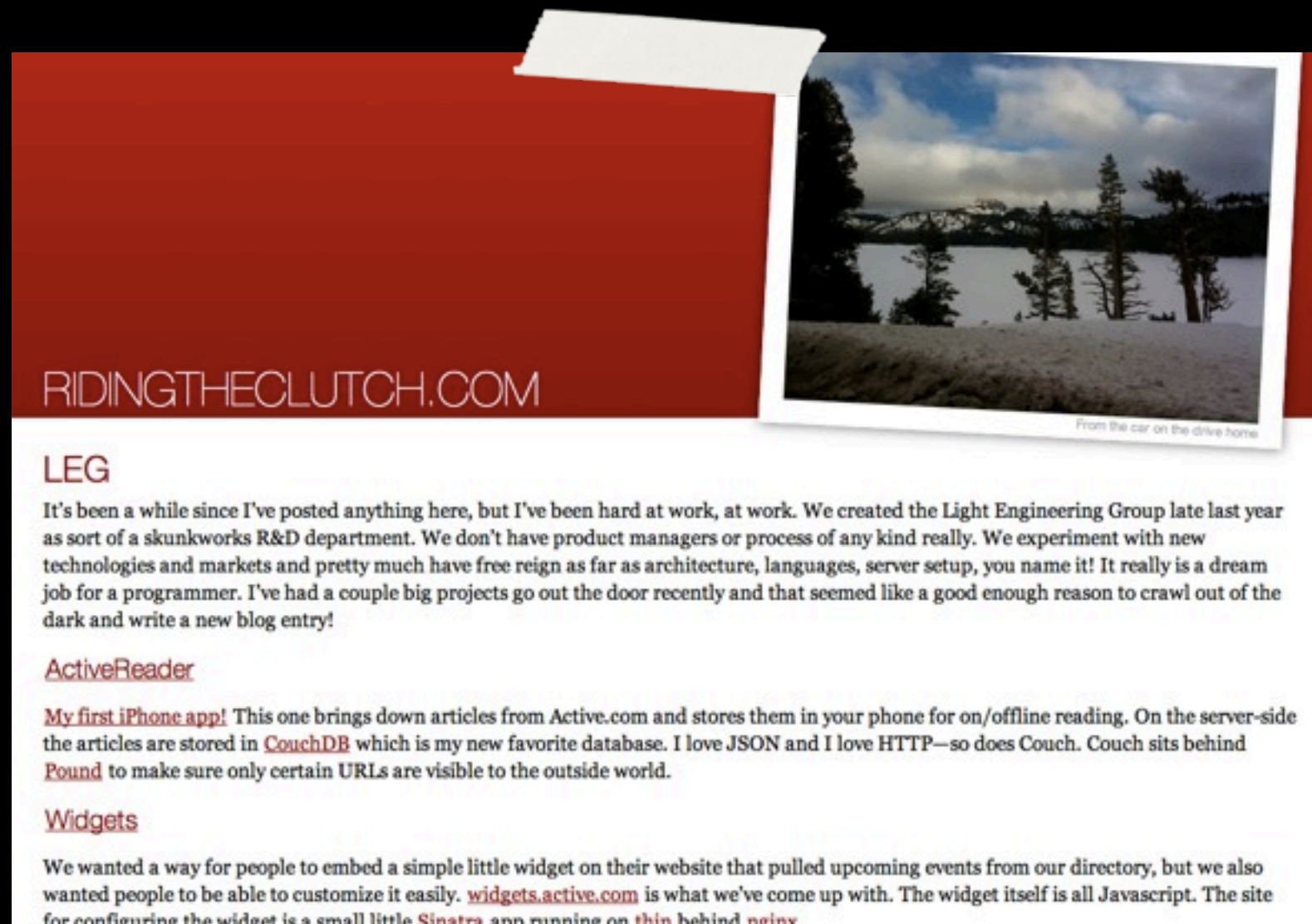


<http://localhost:4000>

```
rob$ serve
[2010-03-23 11:35:38] INFO WEBrick 1.3.1
[2010-03-23 11:35:38] INFO ruby 1.8.7 (2008-08-11)
[2010-03-23 11:35:38] INFO Serve::Server#start: pid=63741 port=4000
```



http://localhost:4000



```
rob$ serve
[2010-03-23 11:35:38] INFO WEBrick 1.3.1
[2010-03-23 11:35:38] INFO ruby 1.8.7 (2008-08-11)
[2010-03-23 11:35:38] INFO Serve::Server#start: pid=63741 port=4000
localhost - - [23/Mar/2010:11:54:40 PDT] "GET / HTTP/1.1" 200 0
- -> /
localhost - - [23/Mar/2010:11:54:40 PDT] "GET /stylesheets/reset-
fonts.css HTTP/1.1" 200 0
http://localhost:4000/ -> /stylesheets/reset-fonts.css
localhost - - [23/Mar/2010:11:54:40 PDT] "GET /stylesheets/
screen.css HTTP/1.1" 200 0
http://localhost:4000/ -> /stylesheets/screen.css
localhost - - [23/Mar/2010:11:54:40 PDT] "GET /stylesheets/
syntax.css HTTP/1.1" 200 0
http://localhost:4000/ -> /stylesheets/syntax.css
localhost - - [23/Mar/2010:11:54:40 PDT] "GET /javascripts/
prototype.js HTTP/1.1" 200 0
http://localhost:4000/ -> /javascripts/prototype.js
localhost - - [23/Mar/2010:11:54:40 PDT] "GET /javascripts/
effects.js HTTP/1.1" 200 0
http://localhost:4000/ -> /javascripts/effects.js
localhost - - [23/Mar/2010:11:54:40 PDT] "GET /javascripts/stack.js
HTTP/1.1" 200 0
http://localhost:4000/ -> /javascripts/stack.js
[2010-03-23 11:54:41] ERROR `/favicon.ico' not found.
localhost - - [23/Mar/2010:11:54:41 PDT] "GET /favicon.ico HTTP/1.1"
404 281
```

demo

serve

questions?

Sinatra



Sinatra

Put this in
your pipe

and smoke it

README
DOCUMENTATION
CONTRIBUTE
CODE
CREW
ABOUT

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

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

Fork me on GitHub

hi.rb

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

```
rob$ ruby hi.rb
```

```
rob$ ruby hi.rb
== Sinatra/0.9.4 has taken the stage on 4567 for development with
backup from Thin
>> Thin web server (v1.2.4 codename Flaming Astroboy)
>> Maximum connections set to 1024
>> Listening on 0.0.0.0:4567, CTRL+C to stop
```

```
rob$ ruby hi.rb
== Sinatra/0.9.4 has taken the stage on 4567 for development with
backup from Thin
>> Thin web server (v1.2.4 codename Flaming Astroboy)
>> Maximum connections set to 1024
>> Listening on 0.0.0.0:4567, CTRL+C to stop
```



```
rob$ ruby hi.rb
== Sinatra/0.9.4 has taken the stage on 4567 for development with
backup from Thin
>> Thin web server (v1.2.4 codename Flaming Astroboy)
>> Maximum connections set to 1024
>> Listening on 0.0.0.0:4567, CTRL+C to stop
```



<http://localhost:4567>

hi.rb

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

```
rob$ ruby hi.rb
== Sinatra/0.9.4 has taken the stage on 4567 for development with
backup from Thin
>> Thin web server (v1.2.4 codename Flaming Astroboy)
>> Maximum connections set to 1024
>> Listening on 0.0.0.0:4567, CTRL+C to stop
```




http://localhost:4567/hi

```
rob$ ruby hi.rb
== Sinatra/0.9.4 has taken the stage on 4567 for development with
backup from Thin
>> Thin web server (v1.2.4 codename Flaming Astroboy)
>> Maximum connections set to 1024
>> Listening on 0.0.0.0:4567, CTRL+C to stop
```



<http://localhost:4567/hi>



Hello World!

demo

sinatra/hi

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

```
require 'rubygems' ← Ruby package manager  
require 'sinatra'  
get '/hi' do  
  "Hello World!"  
end
```

```
require 'rubygems' ← Ruby package manager
require 'sinatra' ← Sinatra package (gem)
get '/hi' do
  "Hello World!"
end
```

```
require 'rubygems' ← Ruby package manager
require 'sinatra' ← Sinatra package (gem)
get '/hi' do ← GET request to /hi
  "Hello World!"
end
```

<code>require 'rubygems'</code>	←	Ruby package manager
<code>require 'sinatra'</code>	←	Sinatra package (gem)
<code>get '/hi' do</code>	←	GET request to /hi
<code> "Hello World!"</code>	←	Respond with text
<code>end</code>		

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

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

} “block”


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

} Whatever is returned from
the block is sent to the
browser

Where's the return?


In Ruby, if you don't say "return,"
it's implied.

In Ruby, if you don't say "return"
it's implied.

The last statement in a block is
always returned.

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

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



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

=

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

A slightly more complex example

beta.rb

```
require 'rubygems'
require 'sinatra'
```

```
get '/' do
  html = '<p>Welcome! Leave your email address for more info!</p>'
  html += '<form method="post" action="/submit">'
  html += '<input type="text" name="email" /><input type="submit" /></form>'
  return html
end
```

```
post '/submit' do
  html = '<p>Thank you for your submission!</p>'
  html += '<p>We will email ' + params['email']
  html += ' when we are ready to launch!</p>'
  return html
end
```

demo

sinatra/beta1

Views

```
`-- beta.rb
```

```
|-- beta.rb
```

```
`-- views
```

```
|-- index.erb
```

```
`-- submit.erb
```

erb

Embedded Ruby

beta.rb

```
require 'rubygems'  
require 'sinatra'
```

```
get '/' do  
  erb :index  
end
```

```
post '/submit' do  
  erb :submit  
end
```

beta.rb

```
require 'rubygems'
require 'sinatra'
```

```
get '/' do
  erb :index
end
```

```
post '/submit' do
  erb :submit
end
```

views/index.erb

```
<p>Welcome! Leave your email address for more info!</p>
<form method="post" action="/submit">
  <input type="text" name="email" />
  <input type="submit" />
</form>
```

beta.rb

```
require 'rubygems'
require 'sinatra'
```

```
get '/' do
  erb :index
end
```

```
post '/submit' do
  erb :submit
end
```

views/index.erb

```
<p>Welcome! Leave your email address for more info!</p>
<form method="post" action="/submit">
  <input type="text" name="email" />
  <input type="submit" />
</form>
```

views/submit.erb

```
<p>Thank you for your submission!</p>
<p>We will email <%= params['email'] %> when we are ready to launch!</p>
```


demo

sinatra/beta2

Layouts

```
|-- beta.rb
`-- views
    |-- index.erb
    |-- layout.erb
    `-- submit.erb
```

views/layout.rb

```
<!DOCTYPE html>
<html>
<head>
  <title>Beta Info</title>
</head>
<body>
  <%= yield %>
</body>
</html>
```

views/layout.rb

```
-----  
<!DOCTYPE html>  
<html>  
<head>  
  <title>Beta Info</title>  
</head>  
<body>  
  <%= yield %>  
</body>  
</html>
```

views/index.erb

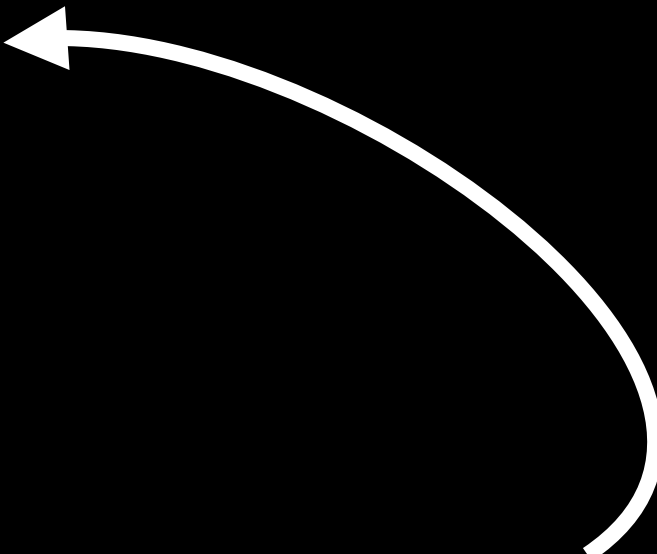
```
-----  
<p>Welcome! Leave your email address for more info!</p>  
<form method="post" action="/submit">  
  <input type="text" name="email" />  
  <input type="submit" />  
</form>
```

views/layout.rb

```
-----  
<!DOCTYPE html>  
<html>  
<head>  
  <title>Beta Info</title>  
</head>  
<body>  
  <%= yield %>  
</body>  
</html>
```

views/index.erb

```
-----  
<p>Welcome! Leave your email address for more info!</p>  
<form method="post" action="/submit">  
  <input type="text" name="email" />  
  <input type="submit" />  
</form>
```



demo

sinatra/beta3

Static Content

Images, stylesheets and Javascript


```
|-- beta.rb
`-- views
    |-- index.erb
    |-- layout.erb
    `-- submit.erb
```

```
|-- beta.rb
|-- public
|-- views
    |-- index.erb
    |-- layout.erb
    |-- submit.erb
```

```
|-- beta.rb
|-- public
|   |-- stylesheets
|       |-- shared.css
|-- views
|   |-- index.erb
|   |-- layout.erb
|   |-- submit.erb
```

demo

sinatra/beta4

So

So

1. Every URL you want to respond to goes in your .rb file

So

1. Every URL you want to respond to goes in your .rb file
2. Pages themselves go in /views as .erb files

So

1. Every URL you want to respond to goes in your .rb file
2. Pages themselves go in /views as .erb files
3. Wrapping layout goes in /views/layout.erb

So

1. Every URL you want to respond to goes in your .rb file
2. Pages themselves go in /views as .erb files
3. Wrapping layout goes in /views/layout.erb
4. Static files go in /public

But

But

If all you have is static web pages, no need to
add every path to your sinatra .rb file and
copy the code to /views

But

If all you have is static web pages, no need to
add every path to your sinatra .rb file and
copy the code to /views

Just put them in /public

But

If all you have is static web pages, no need to add every path to your sinatra .rb file and copy the code to /views

Just put them in /public

The .rb file is more for dynamic URLs or gathering form fields for use later

Dynamic URLs?

dynamic.rb

```
require 'rubygems'  
require 'sinatra'
```

```
get '/' do  
  'This is the homepage'  
end
```

```
get '/:city' do  
  'This is the ' + params['city'] + ' homepage'  
end
```

demo

sinatra/dynamic

questions?

Heroku



[How it Works](#) | [Pricing](#) | [Add-ons](#) | [Docs](#) | [Support](#) [Login](#)

create

instant deployment

Create Heroku apps instantly:

```
$ sudo gem install heroku
$ heroku create sushi
Created http://sushi.heroku.com/
git@heroku.com:sushi.git
```

deploy

pure git workflow

work

complete API

rock-solid ruby platform

Fast, frictionless, and maintenance free.

Deploy and scale with confidence.

53,957

apps running right now

It's free to get started and sign up is instant.

Deploy your first app within minutes.

[Sign Up](#) ▶



Add-ons

Build Powerful Apps Easily
Extend Heroku, add features,
and connect services instantly.
Everything in one place.



New Relic RPM™

Performance Management
Seamlessly integrated into your
app in one-click. Use RPM
Bronze for free on Heroku.

Cardinal
Blue



10,000 requests per minute
Facebook app uses Heroku for
scale, operational experience,
and pricing model



shopify

powered by Heroku
Shopify.com depends on
continuous, reliable, secure
service with no headaches

Kind of like EC2 but specific to
Ruby-based web applications

Run your application/website
for free

Heroku relies on Git.

Git

Like subversion, but more powerful (and complex)

```
rob$ git init .
```



```
rob$ git init .  
Initialized empty Git repository in /Users/rob/Dropbox/Active  
University/sinatra/heroku1/.git/
```

```
rob$ git init .  
Initialized empty Git repository in /Users/rob/Dropbox/Active  
University/sinatra/heroku1/.git/
```

```
rob$ git add .
```

```
rob$ git init .  
Initialized empty Git repository in /Users/rob/Dropbox/Active  
University/sinatra/heroku1/.git/
```

```
rob$ git add .
```

```
rob$ git commit -m 'First commit'
```

```
rob$ git init .  
Initialized empty Git repository in /Users/rob/Dropbox/Active  
University/sinatra/heroku1/.git/
```

```
rob$ git add .
```

```
rob$ git commit -m 'First commit'  
[master (root-commit) d2f7469] First commit  
5 files changed, 31 insertions(+), 0 deletions(-)  
create mode 100644 beta.rb  
create mode 100644 public/stylesheets/shared.css  
create mode 100644 views/index.erb  
create mode 100644 views/layout.erb  
create mode 100644 views/submit.erb
```

Our app is version controlled

```
rob$ heroku create
```

```
rob$ heroku create
Creating blooming-sunrise-28... done!
Created http://blooming-sunrise-28.herokuapp.com/ |
git@heroku.com:blooming-sunrise-28.git
Git remote heroku added
```

```
rob$ heroku create
Creating blooming-sunrise-28... done!
Created http://blooming-sunrise-28.herokuapp.com/ |
git@heroku.com:blooming-sunrise-28.git
Git remote heroku added
```

```
rob$ git push heroku master
```



```
rob$ heroku create
Creating blooming-sunrise-28... done!
Created http://blooming-sunrise-28.herokuapp.com/ |
git@heroku.com:blooming-sunrise-28.git
Git remote heroku added
```

```
rob$ git push heroku master
Counting objects: 10, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (8/8), done.
Writing objects: 100% (10/10), 1009 bytes, done.
Total 10 (delta 0), reused 0 (delta 0)
```

```
-----> Heroku receiving push
!       Heroku push rejected, no Rails or Rack app detected.
```

```
error: hooks/pre-receive exited with error code 1
To git@heroku.com:blooming-sunrise-28.git
! [remote rejected] master -> master (pre-receive hook declined)
error: failed to push some refs to 'git@heroku.com:blooming-sunrise-28.git'
```

demo

sinatra/heroku1

One thing missing

One thing missing

config.ru

```
| -- beta.rb
| -- config.ru
| -- public
|   `-- stylesheets
|       `-- shared.css
`-- views
    |-- index.erb
    |-- layout.erb
    `-- submit.erb
```

config.ru

require 'beta'

run Sinatra::Application

```
rob$ git add .
```

```
rob$ git add .
```

```
rob$ git commit -m 'Added config.ru file for heroku'
```



```
rob$ git add .
```

```
rob$ git commit -m 'Added config.ru file for heroku'  
[master 7b8ab4f] Added config.ru file for heroku  
1 files changed, 2 insertions(+), 0 deletions(-)  
create mode 100644 config.ru
```

```
rob$ git add .
```

```
rob$ git commit -m 'Added config.ru file for heroku'
[master 7b8ab4f] Added config.ru file for heroku
1 files changed, 2 insertions(+), 0 deletions(-)
create mode 100644 config.ru
```

```
rob$ git push heroku master
```

```
rob$ git add .
```

```
rob$ git commit -m 'Added config.ru file for heroku'
[master 7b8ab4f] Added config.ru file for heroku
1 files changed, 2 insertions(+), 0 deletions(-)
create mode 100644 config.ru
```

```
rob$ git push heroku master
Counting objects: 13, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (10/10), done.
Writing objects: 100% (13/13), 1.27 KiB, done.
Total 13 (delta 1), reused 0 (delta 0)
```

```
-----> Heroku receiving push
-----> Sinatra app detected
        Compiled slug size is 4K
-----> Launching..... done
        http://blooming-sunrise-28.herokuapp.com deployed to Heroku
```

```
To git@heroku.com:blooming-sunrise-28.git
* [new branch]      master -> master
```

demo

sinatra/heroku2

So

So

1. Get your app working in development

So

1. Get your app working in development
2. Add a config.ru file

So

1. Get your app working in development
2. Add a config.ru file
3. `git init .` (only the first time)

So

1. Get your app working in development
2. Add a config.ru file
3. `git init .` (only the first time)
4. `git add .`

So

1. Get your app working in development
2. Add a config.ru file
3. `git init .` (only the first time)
4. `git add .`
5. `git commit -m 'message'`

So

1. Get your app working in development
2. Add a config.ru file
3. `git init .` (only the first time)
4. `git add .`
5. `git commit -m 'message'`
6. `heroku create` (only the first time)

So

1. Get your app working in development
2. Add a config.ru file
3. `git init .` (only the first time)
4. `git add .`
5. `git commit -m 'message'`
6. `heroku create` (only the first time)
7. `git push heroku master`

questions?

Installation

Ruby 1.8.6 or higher

<http://ruby-lang.org>

Rubygems 1.3.6

<http://rubyforge.org/projects/rubygems>

Get Serve, Sinatra and Heroku gems

```
gem install serve sinatra heroku
```

Git

<http://git-scm.com/download>

Resources

Ruby Homepage

<http://ruby-lang.org>

Sinatra Homepage

<http://www.sinatrarb.com>

Heroku Homepage

<http://heroku.com>

Git Homepage

<http://git-scm.com>

Example Code

<http://empty-journey-91.herokuapp.com/>

The End