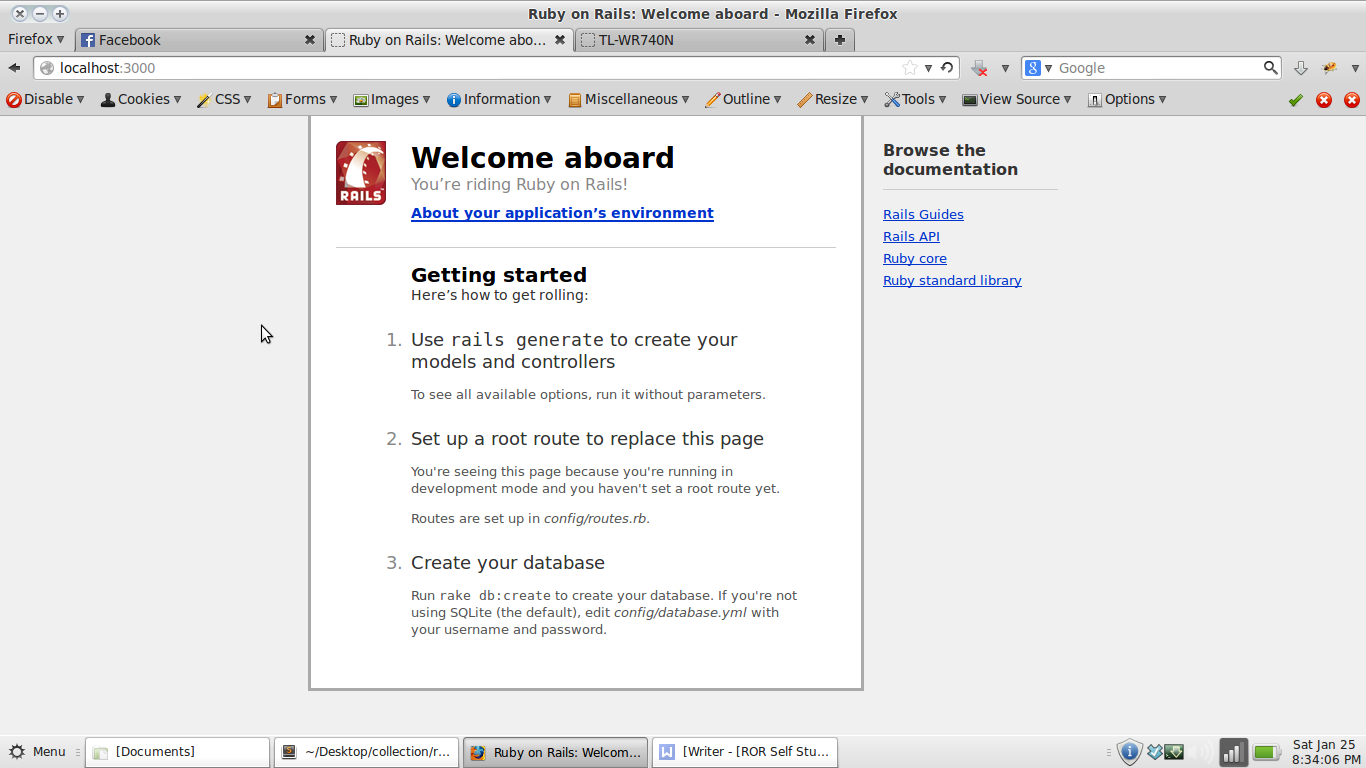
ROR Self Study

Try following on Command line

1. rails --help #to see avilable rails command
2. $ rails new <project name> #create new project with unit test
3. $ rails new <project name> -T #create new project with test case
4. $ bundle #run bundle to install dependencies

Now let’s start rails server

1. $ rails s #will start rails server then you can browse the application using http://localhost:3000/
2. $ rails s -p 5000 # will sepcifiy the port to 5000, then the url to access the project will be <http://localhost:5000/>



Your app will look some thing like this

Data base configuration

Configure config/database.yml as necessary

*development:*

*adapter: mysql2*

*encoding: utf8*

*database: crud\_dev*

*pool: 5*

*username: root*

*password:*

*test:*

*adapter: mysql2*

*encoding: utf8*

*database: crud\_test*

*pool: 5*

*username: root*

*password:*

*production:*

*adapter: mysql2*

*encoding: utf8*

*database: crud\_production*

*pool: 5*

*username: root*

*password:*

If you are using mysql2 adapter then, you will have to update your Gemfile with

*gem ‘mysql2’*

$ bundle # on every update on Gemfile you need to run bundle

If you had added some package via apt-get then it would probaly need to rehash using

$ rbenv rehash

**Command : rake**

1. rake -T #listing all possible rake command
2. rake db:create # crreate database as configured on config/database.yml
3. $ bundle exec rails g # list generators
4. $ rails g controller category new create list delete # generate category controller with action new, create, list and delete
5. $ rails destroy controller category # destory generate controller (i.e category controller)
6. While destorying controller you might have to manually remove routes form config/routes.rb

**Default landing page**

Open config/routes.rb and add

*root "controller\_name#action\_name"*

For example

*root "category#list"*

**Templating on ROR i.e Using simple html + css template as ROR template**

.

├── **app**

│   ├── **assets**

│   │   ├── **images**

│   │   ├── **javascripts**

│   │   │   └── application.js

│   │   └── **stylesheets**

│   │   └── application.css

│   ├── controllers

│   │   ├── application\_controller.rb

│   │   └── concerns

│   ├── helpers

│   │   └── application\_helper.rb

│   ├── mailers

│   ├── models

│   │   ├── concerns

│   │   └── user.rb

│   └── views

│   └── layouts

│   └── **application.html.erb**

├── bin

│   ├── bundle

│   ├── rails

│   └── rake

├── config

│   ├── application.rb

│   ├── boot.rb

│   ├── database.yml

│   ├── environment.rb

│   ├── environments

│   │   ├── development.rb

│   │   ├── production.rb

│   │   └── test.rb

│   ├── initializers

│   │   ├── backtrace\_silencers.rb

│   │   ├── filter\_parameter\_logging.rb

│   │   ├── inflections.rb

│   │   ├── mime\_types.rb

│   │   ├── secret\_token.rb

│   │   ├── session\_store.rb

│   │   └── wrap\_parameters.rb

│   ├── locales

│   │   └── en.yml

│   └── routes.rb

├── config.ru

├── db

│   ├── development.sqlite3

│   ├── migrate

│   │   └── 20130926143431\_devise\_create\_users.rb

│   └── seeds.rb

├── Gemfile

├── Gemfile.lock

├── lib

│   ├── assets

│   └── tasks

├── log

│   └── development.log

├── public

│   ├── 404.html

│   ├── 422.html

│   ├── 500.html

│   ├── favicon.ico

│   └── robots.txt

├── Rakefile

├── README.md

├── README.rdoc

├── tmp

│   ├── cache

│   │   └── assets

│   ├── pids

│   ├── sessions

│   └── sockets

└── vendor

└── assets

├── javascripts

└── stylesheets

**Steps**

1. Copy all the images files to **project\_name/app/assests/images/**
2. Copy all the sytlesheets files to **project\_name/app/assests/stylesheets**
3. Copy all the javascripts files to **project\_name/app/assests/javascripts**
4. Update **project\_name/app/views/layout/application.html.erb**  as necessary

For instance application.html.erb will look some thing like this before updating

<!DOCTYPE html>

<html>

<head>

<title>Gallery</title>

<%= stylesheet\_link\_tag "application", media: "all", "data-turbolinks-track" => true %>

<%= javascript\_include\_tag "application", "data-turbolinks-track" => true %>

<%= csrf\_meta\_tags %>

</head>

<body>

<%= yield %>

</body>

</html>

As per as your templates HTML structure you can update it .

**Creating Shared/partials layout on ROR**

**Steps**

1. Create **Project\_name/app/views/shared (where shared is new folder under views folder)**
2. Create partial or shared views prefixed by underscore for instance

**Project\_name/app/views/shared/\_header.html.erb**

**Project\_name/app/views/shared/\_footer.html.erb**

**Project\_name/app/views/shared/\_sidebar.html.erb**

1. Update view or layout file in following way to use partial or shared views. Sample project\_name/app/views/layout/application.html.erb with partial or shared views

<%= render 'shared/header' %>

<%= render 'shared/leftsidebar' %>

<section id="main" class="column">

<article class="module width\_full">

<!--<header><h3>Stats</h3></header>-->

<div class="module\_content">

<%= yield %>

<div class="clear"></div>

</div>

</article><!-- end of stats article -->

<div class="spacer"></div>

</section>

<%= render 'shared/footer' %>