

# Devise 줌 customizing

( 회원 가입시에 이미지 받아서, 페이스북처럼 글  
쓰면 그 옆에 사용자 이미지 뜨게 해보기)




디렉토리는 빨간색으로 표시  
bash 창 명령어는 파란색으로 표시  
일반 코드들은 검정색으로 표시

완성된 결과화면 ->

[Edit Profile](#) [Logout](#)



## Posts

Title Content		
	1 1	<a href="#">Show</a> <a href="#">Edit</a> <a href="#">Destroy</a>
	2 2	<a href="#">Show</a> <a href="#">Edit</a> <a href="#">Destroy</a>
	3 3	<a href="#">Show</a> <a href="#">Edit</a> <a href="#">Destroy</a>

[New Post](#)

**rails new any\_name**

any\_name 디렉토리에 들어가서

**rails g scaffold post title:text content:text**

### ***Gemfile***

```
gem 'devise'  
gem 'paperclip', '~> 4.1'
```

**bundle install**

**rails generate devise:install**

**rails generate devise user**

**rails g devise:views**

***routes.rb*** 설정

root 'posts#index'

```
<%= link_to 'Login', new_user_session_path %>
```

위 코드 부분을 아래 코드로 교체!

```
<body>
```

```
<% if current_user %>
```

```
  <%= link_to 'Edit Profile', edit_user_registration_path %>
```

```
  <%= link_to 'Logout', destroy_user_session_path, method: :delete %>
```

```
<% else %>
```

```
  <%= link_to 'Login', new_user_session_path %>
```

```
<% end %>
```

 이 부분 추가!

```
<%= yield %>
```

```
</body>
```

```
</html>
```

**rails generate migration add\_avatars\_to\_users**

*db/migrate/~*

```
class AddAvatarsToUsers < ActiveRecord::Migration[5.2]
  def self.up
    change_table :users do |t|
      t.attachment :avatar
    end
  end

  def self.down
    drop_attached_file :users, :avatar
  end
end
```

기존 내용 지우고  
이 부분 추가!

=> rake db:migrate

## *Application\_controller.rb*

```
class ApplicationController < ActionController::Base
  # Prevent CSRF attacks by raising an exception.
  # For APIs, you may want to use :null_session instead.
  protect_from_forgery with: :exception
  #before_filter 라고 하면 안됨
  before_action :configure_permitted_parameters, if: :devise_controller?

  protected

  def configure_permitted_parameters
    devise_parameter_sanitizer.permit(:sign_up, keys: [:email, :password, :avatar])
    devise_parameter_sanitizer.permit(:account_update, keys:
[:email, :password, :current_password, :avatar])
  end
end
```

*views>devise>registrations>new.html.erb*

*views>devise>registrations>edit.html.erb*

```
<div class="form-group">  
  <%= f.label :avatar, class: 'col-sm-2 control-label' %>  
  <div class="col-sm-6">  
    <%= f.file_field :avatar %>  
  </div>  
</div>
```

위의 코드 추가!

Post에 글 쓸 때 user id를 받아올 수 있도록 칼럼 추가!

**rails g migration add\_userid\_to\_posts user\_id:integer**

**rake db:migrate**

## *user.rb*

```
class User < ActiveRecord::Base
  # existing code
  has_many :posts
```

```
  has_attached_file :avatar, :styles => { :medium => "300x300", :thumb =>
    "100x100#" }, :default_url => "/images/:style/missing.png"
  validates_attachment_content_type :avatar, :content_type => /\wAimage\w/.*\wZ/
end
```

사용자가 사진을  
등록하지 않을 시  
기본 이미지 사용

300X300을  
넘을 수 없음

잘라서  
100X100 으로  
만들

## *post.rb*

```
  belongs_to :user
```

추가

*public/images/thumb* 과  
*public/images/medium* 에  
missing.png 추가

(※주의: 원하는 사이즈로 미리  
편집해 두고 올려야 함)



## *application.html.erb*

```
<% if current_user %>
```

```
  <%= link_to 'Edit Profile', edit_user_registration_path %>
```

```
  <%= link_to 'Logout', destroy_user_session_path, method: :delete %>
```

```
<% else %>
```

위의 if, else 문 사이에 아래의 코드 추가

```
<%= image_tag(current_user.avatar.url(:thumb)) %>
```

## *posts\_controller.rb*

@post = Post.new(post\_params) 이 코드를

⇒ @post = current\_user.posts.new(post\_params)

로 교체

## *index.html.erb*

```
<td> <%= image_tag(User.find(post.user_id).avatar.url(:thumb)) %> </td>
```

# 완성!

페이스북처럼 댓글에도 응용할 수 있을 것 같습니다

참고 링크: <http://www.peoplecancode.com/tutorials/users-avatars-uploading-images-using-paperclip>

캐리어웨이브를 이용한 방법도 있다고 하니  
캐리어웨이브를 이용하고 싶으신 분들은  
그 방법으로 찾아서 해보시면 될 것 같습니다!

[Edit Profile](#) [Logout](#)



## Posts

Title Content



1 1 [Show](#) [Edit](#) [Destroy](#)



2 2 [Show](#) [Edit](#) [Destroy](#)



3 3 [Show](#) [Edit](#) [Destroy](#)

[New Post](#)

감사합니다!