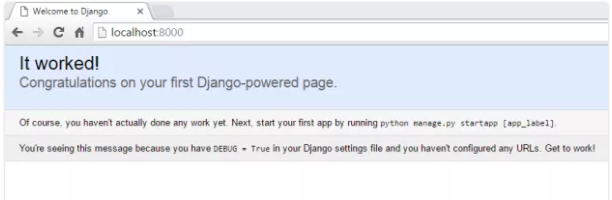
1. Tạo 1 Project



1. Kiểm tra server



* Kết quả:

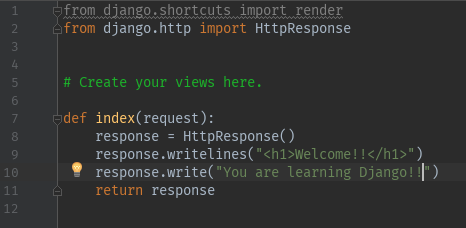


1. Tạo 1 Web App và xử lí yêu cầu khi người dùng yêu cầu truy nhập

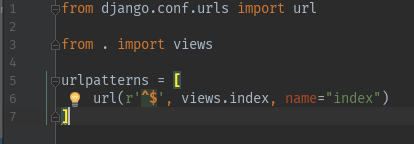
* Ví dụ để tạo một app có tên là home ta sử dụng:



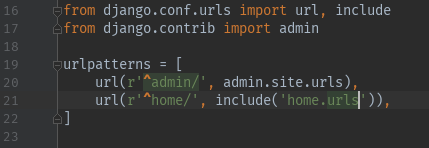
* Trong file view.py thực hiện import thư viện HttpResponse và viết hàm index():



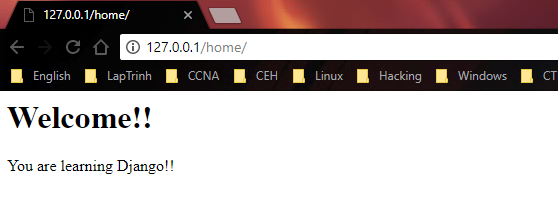
* Tại home tạo urls.py: home/urls.py để tạo đường dẫn liên kết đến views



* Tại Project2/urls.py:



* Chạy server ta được kết quả:

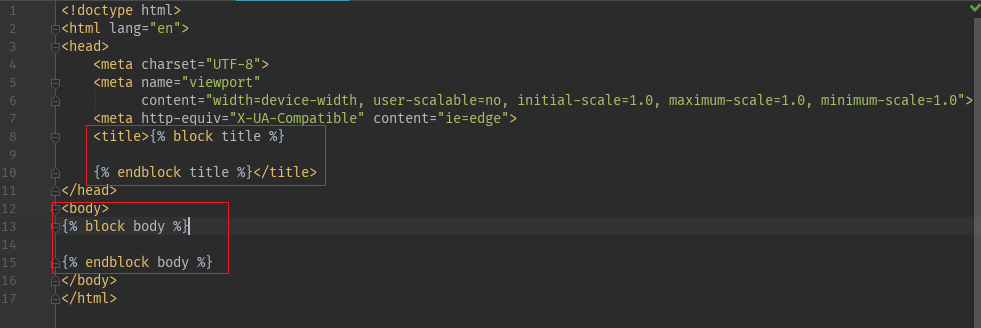


1. Sử dụng Template và Jinja

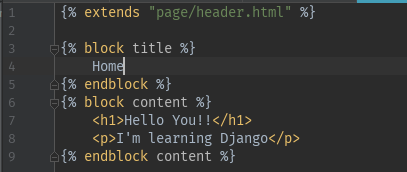
* Tại setting.py thêm app ‘home’ vào INSTALLED\_APPS



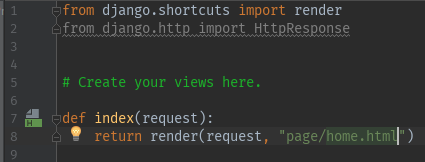
* Tại home/templates/page/header.html



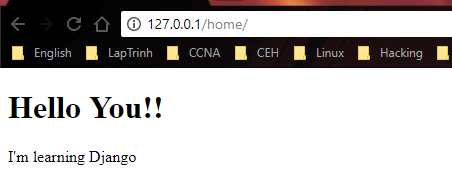
* Tại home/templates/page/home.html



* Khi đó tại home/views.py ta sửa lại:

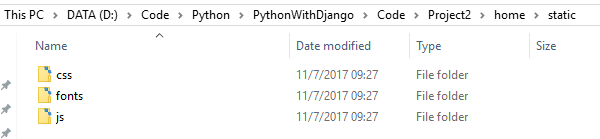


* Chạy server:

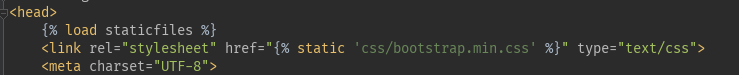


1. File tĩnh và thiết kế web bằng Bootstrap

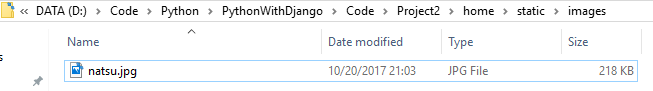
* Tải Bootstrap tại: <http://getbootstrap.com.vn/getting-started/#download>
* Tại home tại folder static và giải nén file Bootstrap vừa tải về vào đó.



* Tại header.html:



* Tại thư mục static tạo thêm folder images và copy 1 ảnh bất kì vào đó:



* Phần body của file header.htm được viết lại như sau:

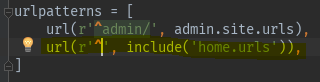


* Chạy server:

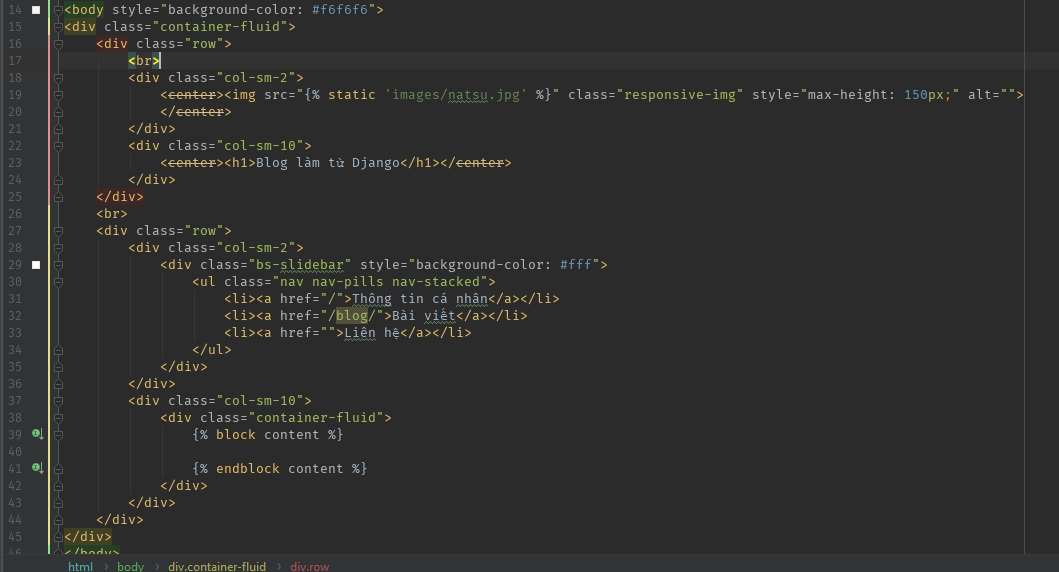


1. Thiết kế blog hoàn chỉnh

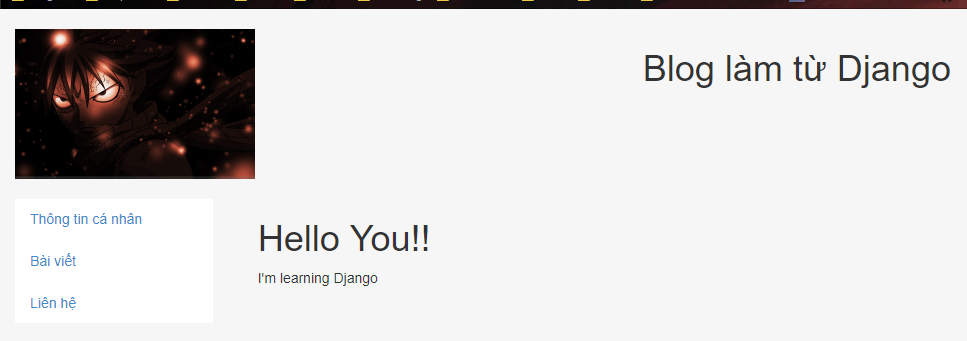
* Tại Project2/urls.py sửa lại:



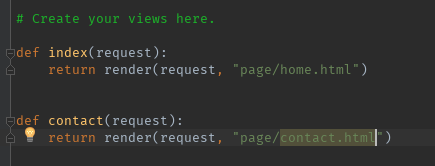
* File header.htm sau khi được viết lại:



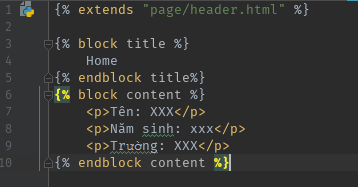
* Chạy server:



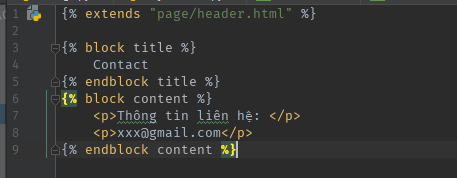
* Tại home/views.py viết thêm hàm contact:



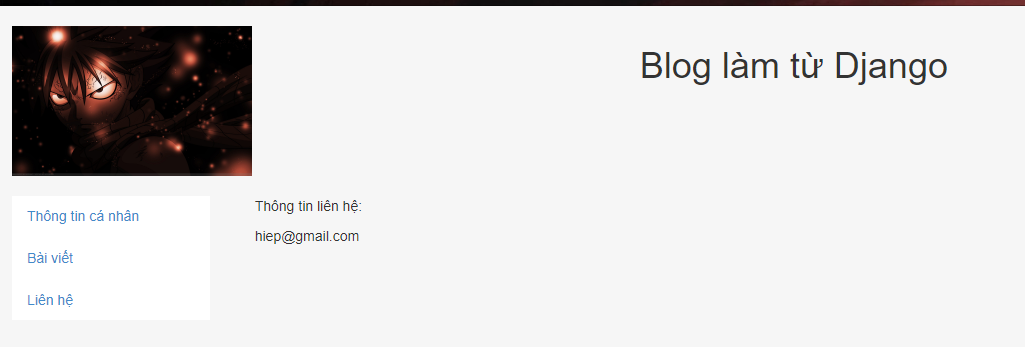
* Sửa home.html:



* Tạo liên kết cho phần Liên hệ: Tại home/templates/page/contact.html:



* Chạy server:

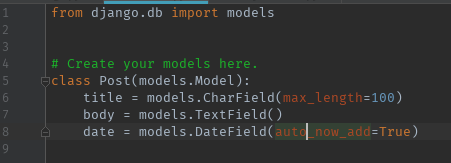


1. Dùng Model tạo database

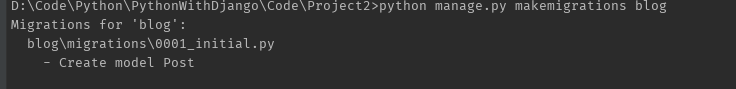
* Tạo một app tên là bolg:



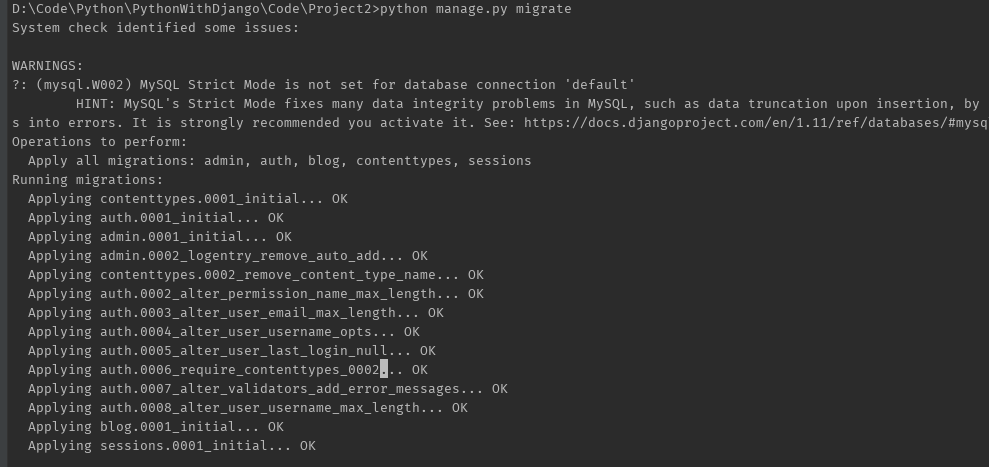
* Nhớ khai báo app trong setting.py
* Trong blog/models.py:



* Khai báo cho project có sự thay đổi trong database:

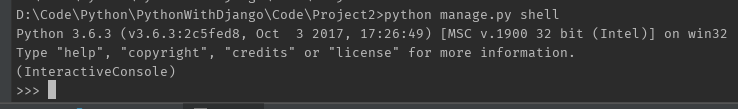


* Tạo ra CSDL:

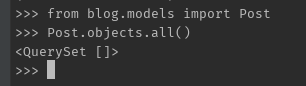


1. Tương tác database bằng python

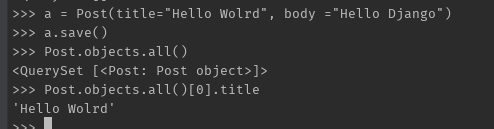
* Dùng shell:



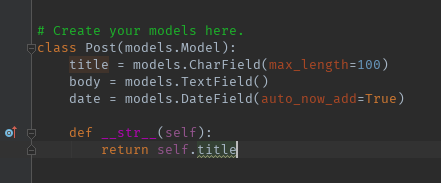
* Hiển thị toàn bộ đối tượng trong database:



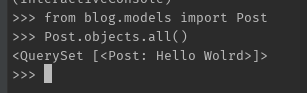
* Thêm một đối tượng Post:



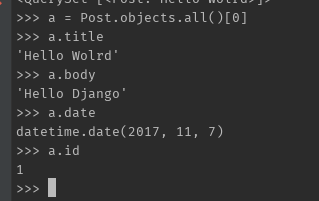
* Sửa lại models.py:



* Chạy lại shell để hiện thị luôn title của đối tượng ra:

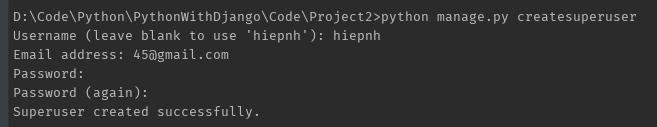


* Một số thao tác khác:

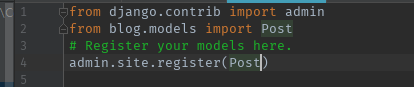


1. Hệ thống Admin

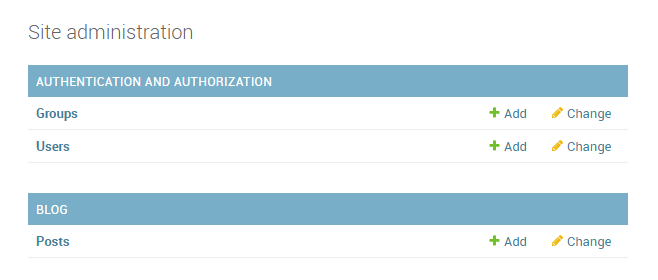
* Tạo 1Admin:

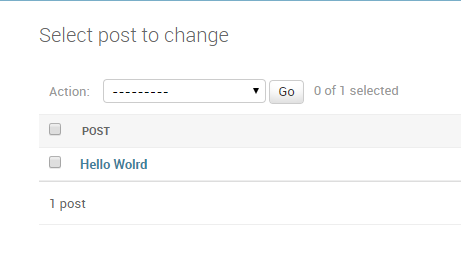


* Truy cập trang admin: <http://127.0.0.1/admin/>
* Đăng kí đối tượng với admin tại blog/admin.py:



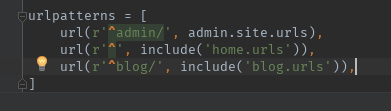
* Chạy server:



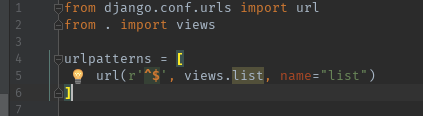


1. Liệt kê danh sách bài viết

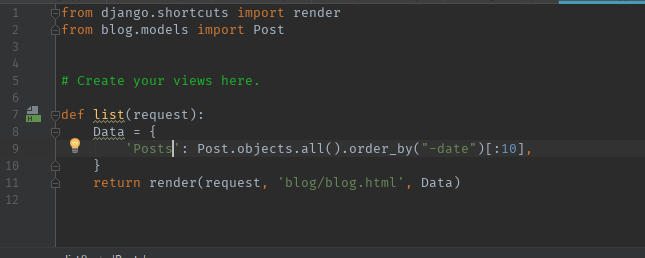
* Tại Project2/urls.py thêm url cho blog đồng thời tại folder tạo file urls.py:



* Tại blog/urls.py:



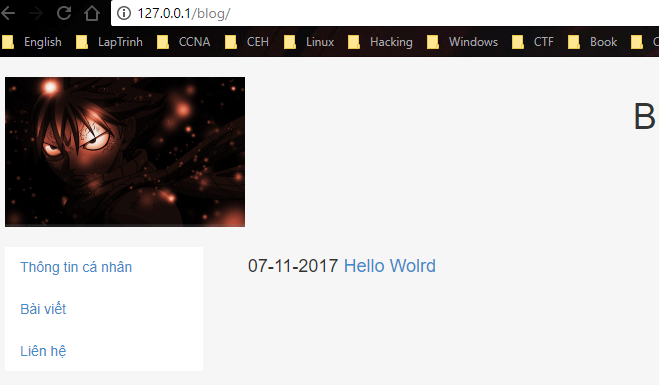
* Tại blog/views.py tạo hàm list:



* Tạo blog/templates/blog/blog.html:



* Chạy server:

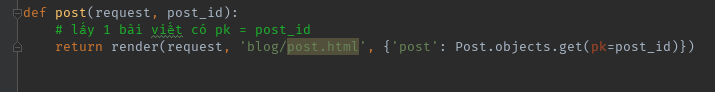


1. Hiển thị thông tin bài viết

* Tại blog/urls.py viết thêm:



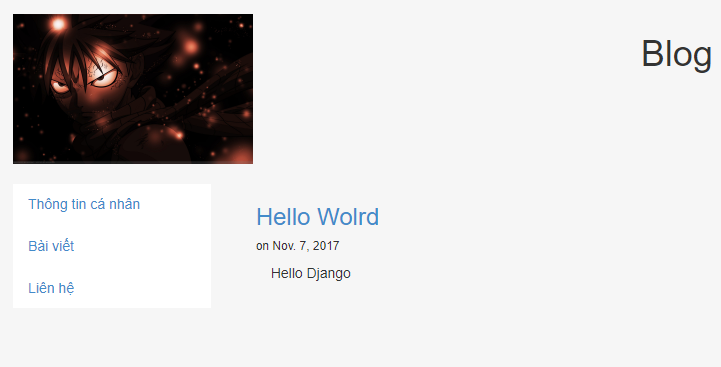
* Tại blog/views.py viết hàm post:



* Tại blog/templates/blog/post/html:

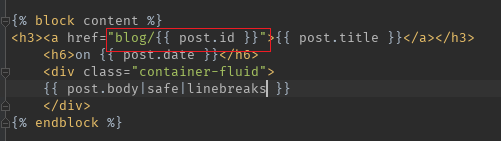


* Chạy server:

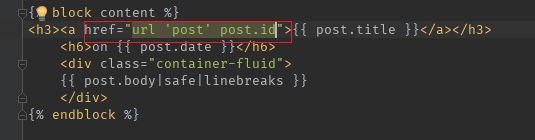


1. Loại bỏ Hardcoded URL

* Vùng khoanh đỏ được coi là Hardcode và rất bất tiện khi Blog của bạn phát triển thêm nhiều nội dung khác, vì vậy chúng ta cần loại bỏ nó.

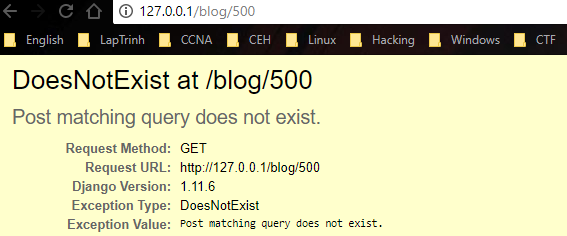


* Tại post.htm ta sửa lại như sau:



1. Xử lí lỗi 404

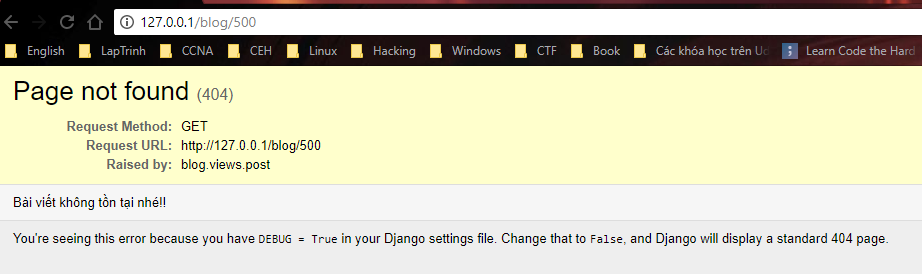
* Ví dụ blog của chúng ta có 10 bài viết nhưng khi ta nhập post\_id vượt quá 10 thì sẽ báo lỗi:



* Tại blog/views.py ta import thêm thư viện Http404. Hàm post của views.py được sửa lại như sau:



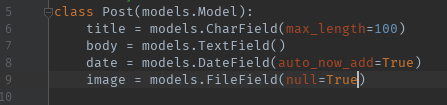
* Chạy server:



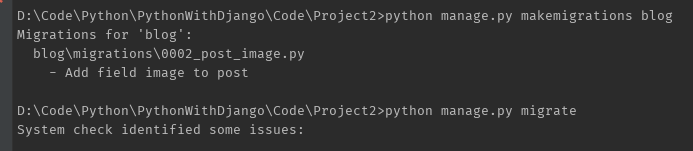
* Ngoài ra ta có thể tự tạo trang báo lỗi 404 cho mình cũng được hoặc gặp lỗi thì tự động chuyển về trang chủ.

1. Upload File cho Blog

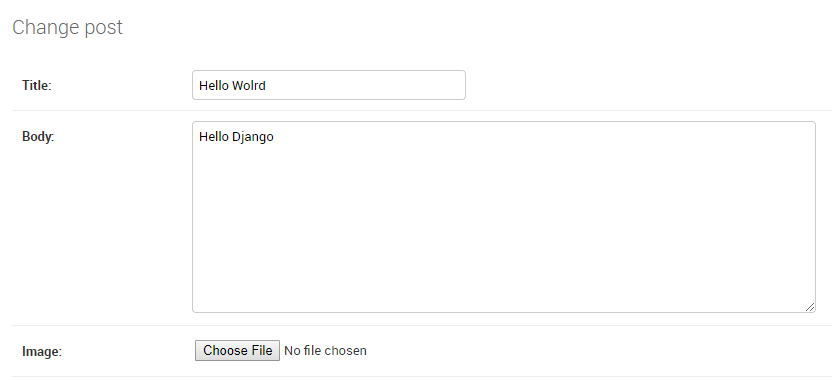
* Trong blog/models.py thêm biến nữa gọi là image



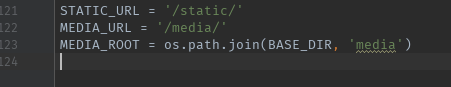
* Sau đó update lại database với 2 câu lệnh sau:



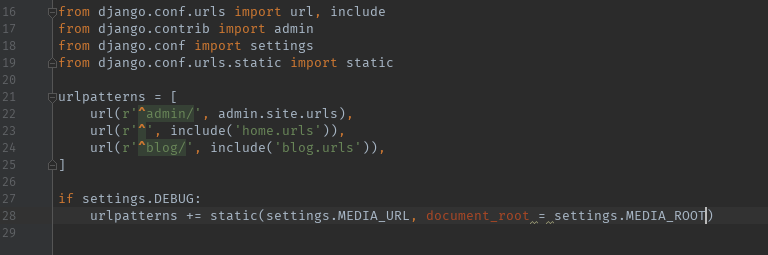
* Chạy server vào trang Admin:



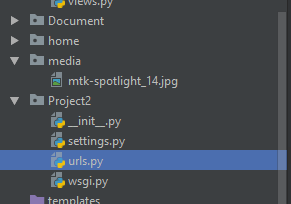
* File được upload sẽ lưu tại Project2, vì vậy ta cần quản lí việc upload file. Do đó, ta cần tạo một file để chứa file được upload. Tại file setting.py ta tạo thêm như sau:



* Tại Project2/urls.py:



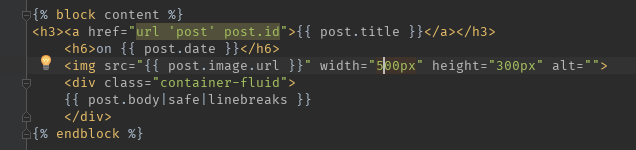
* Chạy server và upload file, ta sẽ thấy trong project có file media chứa file ảnh được upload:



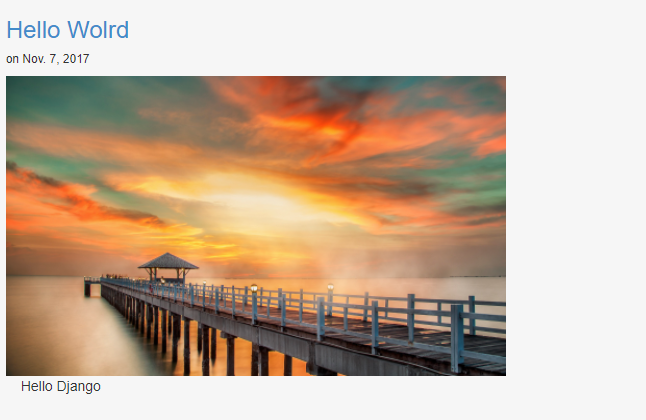
* Tiếp theo ta sẽ cho hiển thị ảnh upload ra cùng bài viết. Tại blog.html ta viết thêm:



* Đổng thời tại post.html cũng thêm dòng:

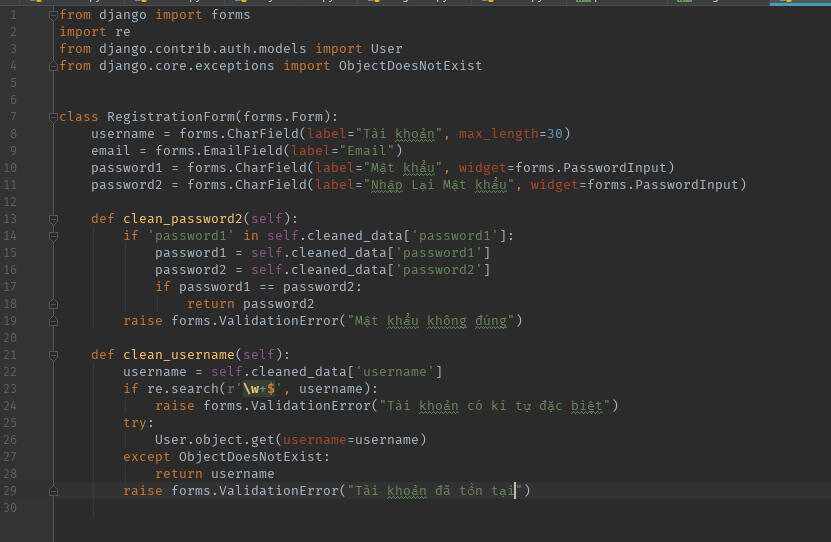


* Chạy server:



1. Tạo form đăng kí tài khoản

* Tại home tạo forms.py:



* Tại home/views.py viết hàm xử lí đăng kí register():



* Home/template/page/register.html:



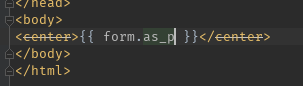
* Tại home/urls.py ta thêm urls cho register:



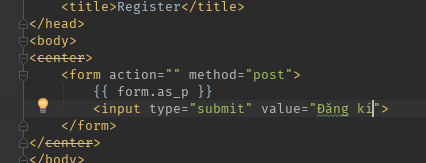
* Chạy server: <http://127.0.0.1/register/>



* Để các form này xuống dòng ta sửa lại ở register.html:



* Tiếp tục xử lí form trong file register.html:



* Để hoàn thành việc đăng kí user vào database:

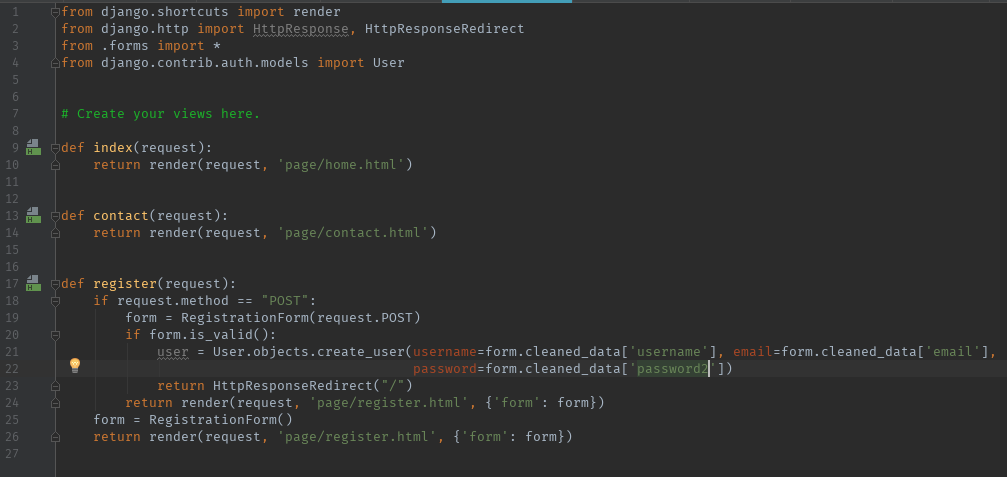
+ forms.py:



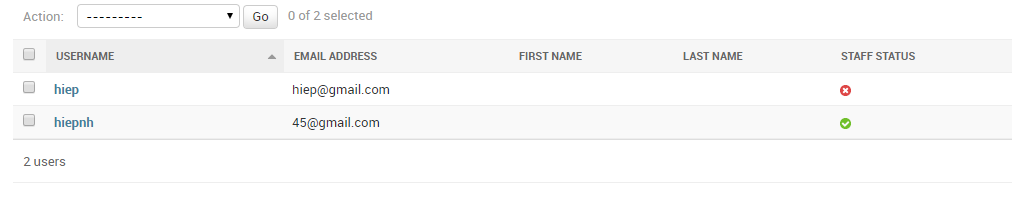
+ register.html:



+ home/views.py:



* Chạy server và vào trang admin ta thấy user đã được đăng kí:

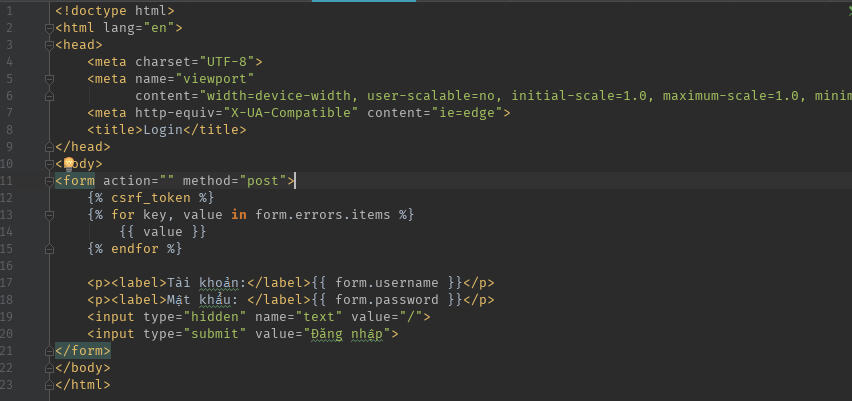


1. Xử lí login và logout

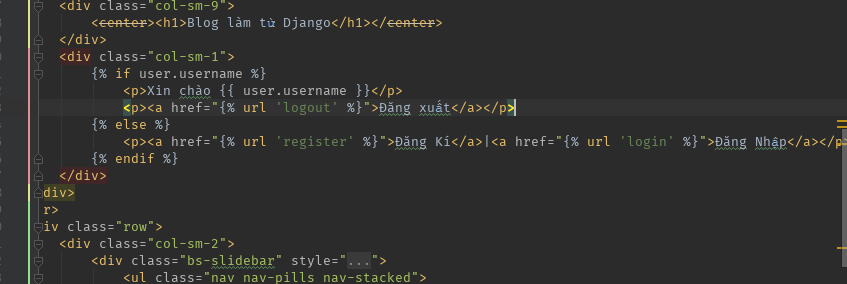
* Tại home/urls.py:



* Tại home/template/page/login.html:



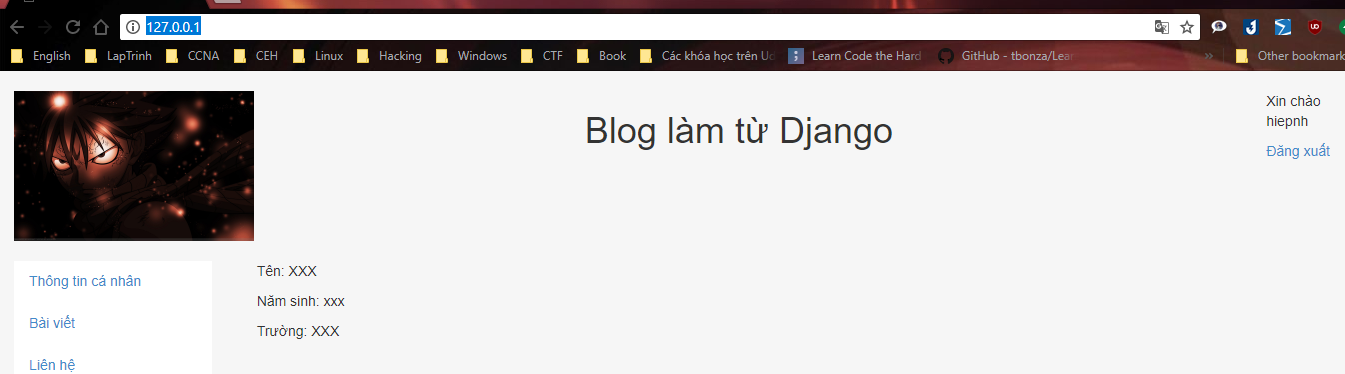
* Tại header.html sửa lại:



* Tại home/urls.py:



* Chạy server: <http://127.0.0.1/>

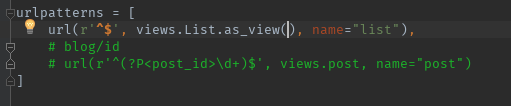


1. Sử dụng Generic view

* Tại blog/views.py ta viết lại thành:



* Tại blog/urls.py:

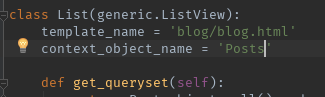


* Tại blog.html sửa lại thành:



* Chạy server: OK
* Ngoài ra nếu không muốn để mặc định là object\_list ta làm như sau:

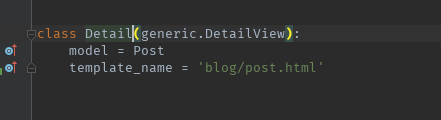
+ Tại blog/views.py:



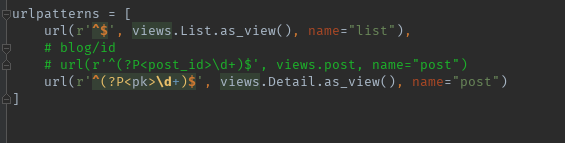
+ Bên blog.html ta giữ nguyên là Post như ban đầu.

* Tiếp tục xử lí in ra nội dung bài viết:

+ Tại blog/views.py tạo class Detail:



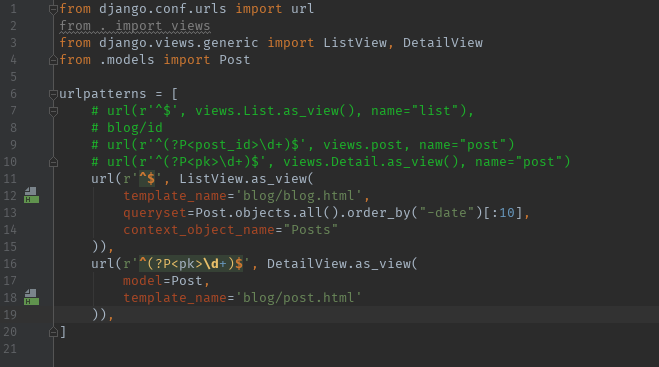
+ sang urls.py ta sửa lại:



* Chạy server: Ok
* Sử dụng generic có thể rút gọn code hơn nữa:

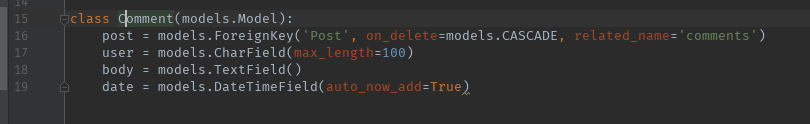
+ Xóa bỏ code ở views.py đi.

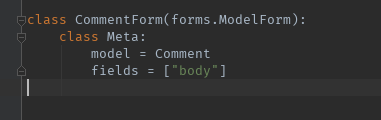
+ Ta viết thẳng code trong urls.py



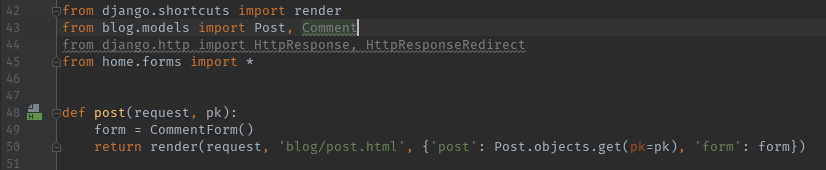
1. Tạo bình luận trong blog

* Trong blog/models.py tạo thêm comment:

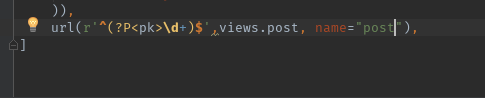
- Trong forms.py tạo thêm class:



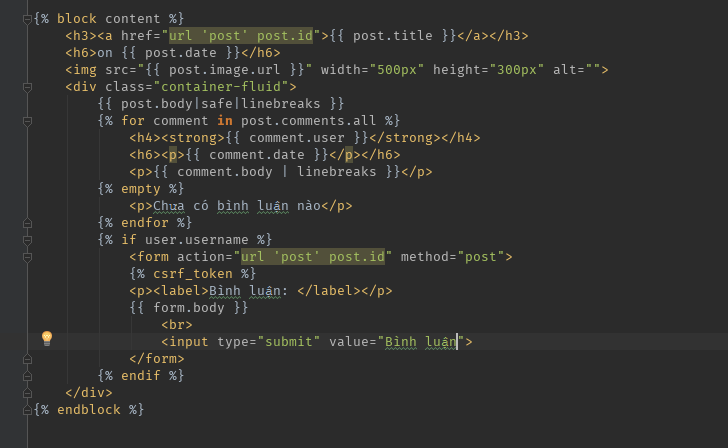
* Sau đó update database với 2 câu lệnh makemigrations và migrate
* Vào views.py tạo hàm xử lí comment



* Sửa lại urls.py:



* Đến file post.html:



* Đến file views.py:

