<http://launchwithcode.com/>

# Start First Django Project

本章内容包括

* Django的安装
* 创建django项目
* 本地服务器的启动
* 初始化数据库

**安装django**

pip install django=1.5.5

**创建django项目**

django-admin.py startproject lwc

**启动本地服务器**

python manage.py runserver

127.0.0.1:8000 第一个django项目

**初始化数据库**

创建表

python mange.py syncdb

创建表和超级用户

python mange.py createsuperuser

# First View using a Function Based View

基本的view功能包括

* 创建url
* 创建view

在url.py添加新的url

urlpatterns **=** patterns**(**''**,**

# Examples:

url**(**r'^$'**,** lwc**.**views**.**home', name='home'),

**)**

创建views.py

**from** django**.**shortcuts **import** render

**def** home**():**

context **=** **{}**

template **=** "home.html"

**return** render**(**request**,** template**,** context**)**

# Settings Setup Django Main Configuration

Django的主要配置

* 模板
* 调试开关

**模板**

创建lwc/templates & lwc/templates/home.html

配置settings.py

TEMPLATE\_DIRS **=** **(**

# Put strings here, like "/home/html/django\_templates" or "C:/www/django/templates".

# Always use forward slashes, even on Windows.

# Don't forget to use absolute paths, not relative paths.

os**.**path**.**join**(**BASE\_DIR**,**"templates"**)**

**调试开关**

DEBUG

TEMPLATE\_DEBUG

…..

# Implement Bootstrap Front End Framework to Django

本章将介绍如何在django中使用bootstraps

Step1 拷贝下面代码

<http://getbootstrap.com/examples/jumbotron/>

step2 网络相对地址转为绝对链接或者本地链接

<link rel="icon" href="../../favicon.ico">

<link rel="icon" href="http://getbootstrap.com/favicon.ico">

<!-- Bootstrap core CSS -->

<link href="../../dist/css/bootstrap.min.css" rel="stylesheet">

<link href="http://getbootstrap.com/dist/css/bootstrap.min.css" rel="stylesheet">

href="../../dist/css/bootstrap.min.css" = <http://getbootstrap.com/examples/jumbotron/>../../dist/css/bootstrap.min.css

<!-- IE10 viewport hack for Surface/desktop Windows 8 bug -->

<link href="../../assets/css/ie10-viewport-bug-workaround.css" rel="stylesheet">

<link href="http://getbootstrap.com/assets/css/ie10-viewport-bug-workaround.css" rel="stylesheet">

<!-- Custom styles for this template -->

<link href="jumbotron.css" rel="stylesheet">

<link href="http://getbootstrap.com/examples/jumbotron/jumbotron.css" rel="stylesheet">

<script src="../../assets/js/ie-emulation-modes-warning.js"></script>

<script src="http://getbootstrap.com/assets/js/ie-emulation-modes-warning.js"></script>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>

<script src="http://code.jquery.com/jquery-latest.js"></script>

<script>window.jQuery || document.write('<script src="../../assets/js/vendor/jquery.min.js"><\/script>')</script>

<script>window.jQuery || document.write('<script src=" http://getbootstrap.com/assets/js/vendor/jquery.min.js"><\/script>')</script>

<script src="../../dist/js/bootstrap.min.js"></script>

<script src="http://getbootstrap.com/dist/js/bootstrap.min.js"></script>

<script src="../../assets/js/ie10-viewport-bug-workaround.js"></script>

<script src="http://getbootstrap.com/assets/js/ie10-viewport-bug-workaround.js"></script>

# Start a Django

本节将介绍如果创建一个django应用，包括

* model & db
* admin
* form

**创建application**

python manage.py startapp joins

**创建model**

joins/models.py

# Create your models here.

**class** **Join(**models**.**Model**):**

email **=** models**.**EmailField**()**

timestamp **=** models**.**DateTimeField**(**auto\_now\_add **=** **True,** auto\_now **=** **False)**

updated **=** models**.**DateTimeField**(**auto\_now\_add **=** **False,** auto\_now **=** **True)**

**def** \_\_unicode\_\_**(**self**):**

**return** "%s " **%(**self**.**email**)**

**生成数据表**

python manage.py syncdb

将会创建joins\_join表

需执行这个命令，否则user auth里找不到对应的model

**创建admin接口**

Joins\admin.py

**from** django**.**contrib **import** admin

**from** **.**models **import** Join

**class** **JoinAdmin(**admin**.**ModelAdmin**):**

list\_display **=** **[**'\_\_unicode\_\_'**,** 'timestamp'**,** 'updated'**]**

**class** **Meta:**

model **=** Join

admin**.**site**.**register**(**Join**,** JoinAdmin**)**

<https://docs.djangoproject.com/en/1.6/ref/models/fields/> 已失效

<https://docs.djangoproject.com/en/1.10/ref/models/fields/>

# Using Django Forms

form的修改包括

* 创建新的form文件，并定义form
* View中对form进行处理，包括
  + Get
  + Post
* 模板中添加form

新建form文件lwc/forms.py，并定义form

**from** django **import** forms

**from** **.**models **import** Join

**class** **EmailForm(**forms**.**Form**):**

name **=** forms**.**CharField**(**required **=** **False)**

email **=** forms**.**EmailField**()**

**class** **JoinForm(**forms**.**ModelForm**):**

**class** **Meta:**

model **=** Join

添加对form的处理

* nonModel Form
* ModelForm

joins/views.py

**def** home**(**request**):**

form **=** EmailForm**(**request**.**POST **or** **None)**

**if** form**.**is\_valid**():**

email **=** form**.**cleaned\_data**[**'email'**]**

new\_join**,** created **=** Joins**.**objects**.**get\_or\_create**(**email**=**email**)**

**print** new\_join**,** created

form **=** JoinForm**(**request**.**POST **or** **None)**

**if** form**.**is\_valid**():**

new\_join **=** form**.**save**(**commit **=** **False)**

email **=** form**.**cleaned\_data**[**'email'**]**

new\_join\_old**,** created **=** Join**.**objects**.**get\_or\_create**(**email **=** email**)**

context **=** **{**"form"**:**form**}**

template **=** "home.html"

**return** render**(**request**,** template**,** context**)**

模板中添加form

templates/home.html

**{% extends "base.html" %}**

**{% block content %}**

<h1> **Welcome Home!** </h1>

<form method=**"POST"** action=**""**>**{% csrf\_token %}**

<input type=**"email"** name=**"email"** placeholder=**"Your email..."** />

**{{form.as\_p}}**

<input type=**"submit"** value=**"join"** class=**"btn"** />

</form>

**{% endblock %}**

method 这儿可以是GET或者POST

action 是url

**{% csrf\_token %}** 安全，防止密码泄露

form可以用non-Model form或者model form

# Make changes to Django Models with South

安装south

pip install south

INSTALLED\_APPS **=** **(**

'joins'**,**

'django.contrib.auth'**,**

'django.contrib.contenttypes'**,**

'django.contrib.sessions'**,**

'django.contrib.sites'**,**

'django.contrib.messages'**,**

'django.contrib.staticfiles'**,**

# Uncomment the next line to enable the admin:

'django.contrib.admin'**,**

# Uncomment the next line to enable admin documentation:

'django.contrib.admindocs'**,**

'south'

**)**

用south来控制joins的migration

python manage.py convert\_to\_south joins

这条命令会创建history和初始的migration

本地运行时，之前已有joins数据库，一个简单方法是直接删除joins/migrations,然后执行sync，将会从头执行

python manage.py syncdb

将会创建表 south\_migrationhistory, joins\_join

执行修改

python manage.py schemamigration joins –auto

migrate db

python manage.py migrate joins

<https://github.com/codingforentrepreneurs/Guides/blob/master/all/using_south_in_django.md>

1. Install south : pip install south, add south to settings.py in INSTALLED APPS
2. Ensure model is in sync in database
3. Covert the model to south with : python manage.py convert\_to\_south appname
4. Make changes to model (eg add new fields : ip\_address = model.CharField(max\_length=120
5. Run schemamigration : python manage.py schemamigration appname --auto
6. Run migrate : python mange.py migrate

# Get User IP Address from Requests

获取ip包括以下几个步骤

* Join添加ip字段
* 更新join form的显示，ip这个字段并不开放给用户
* 从request.META.get里获取ip
* 在form处理时把这个ip存进join

<https://github.com/codingforentrepreneurs/Guides/blob/master/all/using_south_in_django.md>

views.py

**def** get\_ip**(**request**):**

**try:**

x\_forward **=** request**.**META**.**get**(**"HTTP\_X\_FORWORDED\_FOR"**)**

**if** x\_forward**:**

ip **=** x\_forward**.**split**(**','**)[**0**]**

**else:**

ip **=** request**.**META**.**get**(**"REMOTE\_ADDR"**)**

**except:**

ip **=** ""

**return** ip

**def** home**(**request**):**

'''

form = EmailForm(request.POST or None)

if form.is\_valid():

email = form.cleaned\_data['email']

new\_join, created = Joins.objects.get\_or\_create(email=email)

print new\_join, created

'''

form **=** JoinForm**(**request**.**POST **or** **None)**

**if** form**.**is\_valid**():**

new\_join **=** form**.**save**(**commit **=** **False)**

email **=** form**.**cleaned\_data**[**'email'**]**

new\_join\_old**,** created **=** Join**.**objects**.**get\_or\_create**(**email **=** email**)**

**if** created**:**

new\_join\_old**.**ip\_address **=** get\_ip**(**request**)**

new\_join\_old**.**save**()**

context **=** **{**"form"**:**form**}**

template **=** "home.html"

**return** render**(**request**,** template**,** context**)**

**class** **Join(**models**.**Model**):**

email **=** models**.**EmailField**(**unique **=** **True)**

ip\_address **=** models**.**CharField**(**max\_length **=** 120**,** default **=** 'ABC'**)**

timestamp **=** models**.**DateTimeField**(**auto\_now\_add **=** **True,** auto\_now **=** **False)**

updated **=** models**.**DateTimeField**(**auto\_now\_add **=** **False,** auto\_now **=** **True)**

**def** \_\_unicode\_\_**(**self**):**

**return** "%s " **%(**self**.**email**)**

**class** **JoinForm(**forms**.**ModelForm**):**

**class** **Meta:**

model **=** Join

fields **=** **[**'email'**,]**

只显示email

# Create Custom Reference ID

Reference处理包括以下几个步骤

* 在join添加ref\_id字段
* 从uuid里面拿出值，并置成ref\_id
* 在form POST处理时把这个字段存进join

运行 python manage.py shell

>>> import uuid

>>> uuid.uuid4()

**class** **Join(**models**.**Model**):**

email **=** models**.**EmailField**(**unique **=** **True)**

ref\_id **=** models**.**CharField**(**max\_length**=**120**,** default**=**'ABC'**,** unique**=True)**

ip\_address **=** models**.**CharField**(**max\_length **=** 120**,** default **=** 'ABC'**)**

timestamp **=** models**.**DateTimeField**(**auto\_now\_add **=** **True,** auto\_now **=** **False)**

updated **=** models**.**DateTimeField**(**auto\_now\_add **=** **False,** auto\_now **=** **True)**

**def** \_\_unicode\_\_**(**self**):**

**return** "%s %s" **%(**self**.**email**,** self**.**ref\_id**)**

**class** **Meta:**

unique\_together **=** **(**"email"**,** "ref\_id"**,)**

**import** uuid

**def** get\_ref\_id**():**

ref\_id **=** str**(**uuid**.**uuid4**())[:**11**].**replace**(**'-'**,**''**).**lower**()**

**try:**

id\_exists **=** Join**.**objects**.**get**(**ref\_id**=**ref\_id**)**

get\_ref\_id**()**

**except:**

**return** ref\_id

**def** home**(**request**):**

form **=** JoinForm**(**request**.**POST **or** **None)**

**if** form**.**is\_valid**():**

new\_join **=** form**.**save**(**commit **=** **False)**

email **=** form**.**cleaned\_data**[**'email'**]**

new\_join\_old**,** created **=** Join**.**objects**.**get\_or\_create**(**email **=** email**)**

**if** created**:**

new\_join\_old**.**ref\_id **=** get\_ref\_id**()**

new\_join\_old**.**ip\_address **=** get\_ip**(**request**)**

new\_join\_old**.**save**()**

context **=** **{**"form"**:**form**}**

template **=** "home.html"

**return** render**(**request**,** template**,** context**)**

# Create a Social Sharing Page to Share

创建分享页

* 添加url
* 将email字段unique去掉
* 添加分享页view
* 添加分享页template
* 添加view中重定向到分享页的处理

urlpatterns **=** patterns**(**''**,**

url**(**r'^(?P<ref\_id>.\*)$'**,** 'joins.views.share'**,** name**=**'share'**),**

**)**

**from** django**.**shortcuts **import** render**,** HttpResponseRedirect

**def** home**(**request**):**

form **=** JoinForm**(**request**.**POST **or** **None)**

**if** form**.**is\_valid**():**

new\_join **=** form**.**save**(**commit **=** **False)**

email **=** form**.**cleaned\_data**[**'email'**]**

new\_join\_old**,** created **=** Join**.**objects**.**get\_or\_create**(**email **=** email**)**

**if** created**:**

new\_join\_old**.**ref\_id **=** get\_ref\_id**()**

new\_join\_old**.**ip\_address **=** get\_ip**(**request**)**

new\_join\_old**.**save**()**

**return** HttpResponseRedirect**(**"/%s" **%(**new\_join\_old**.**ref\_id**))**

context **=** **{**"form"**:**form**}**

template **=** "home.html"

**return** render**(**request**,** template**,** context**)**

**class** **Join(**models**.**Model**):**

#email = models.EmailField(unique = True)

email **=** models**.**EmailField**()**

friend **=** models**.**ForeignKey**(**"self"**,** related\_name**=**'referral'**,** null**=True,** blank**=True)**

ref\_id **=** models**.**CharField**(**max\_length**=**120**,** default**=**'ABC'**,** unique**=True)**

ip\_address **=** models**.**CharField**(**max\_length **=** 120**,** default **=** 'ABC'**)**

timestamp **=** models**.**DateTimeField**(**auto\_now\_add **=** **True,** auto\_now **=** **False)**

updated **=** models**.**DateTimeField**(**auto\_now\_add **=** **False,** auto\_now **=** **True)**

**def** \_\_unicode\_\_**(**self**):**

**return** "%s %s" **%(**self**.**email**,** self**.**ref\_id**)**

**class** **Meta:**

unique\_together **=** **(**"email"**,** "ref\_id"**,)**

unique改成False，否则Join的时候check不过

joins/views.py

**def** share**(**request**,** ref\_id**):**

join\_obj **=** Join**.**objects**.**get**(**ref\_id**=**ref\_id**)**

context **=** **{**"ref\_id"**:** join\_obj**.**ref\_id **}**

template **=** "share.html"

**return** render**(**request**,** template**,** context**)**

templates/share.html

**{% extends "base.html" %}**

**{% block content %}**

**{% include "navbar.html" %}**

**ref\_id : {{ref\_id}}**

**{% endblock %}**

# Use Custom Django Middleware to Track Shares

添加middleware处理分享页

* 添加middleware类
* 从middleware中获取reference值 (url中的内容)，并将这个值存入session
* 在view中将session中的值拿到，并根据这个值get到join

<https://docs.djangoproject.com/en/1.10/topics/http/middleware/>

MIDDLEWARE\_CLASSES **=** **(**

'django.middleware.common.CommonMiddleware'**,**

'django.contrib.sessions.middleware.SessionMiddleware'**,**

'django.middleware.csrf.CsrfViewMiddleware'**,**

'django.contrib.auth.middleware.AuthenticationMiddleware'**,**

'django.contrib.messages.middleware.MessageMiddleware'**,**

# Uncomment the next line for simple clickjacking protection:

# 'django.middleware.clickjacking.XFrameOptionsMiddleware',

'lwc.middleware.ReferMiddleware'**,**

**)**

创建lwc/middleware.py

**from** joins**.**models **import** Join

**class** **ReferMiddleware():**

**def** process\_request**(**self**,** request**):**

ref\_id **=** request**.**GET**.**get**(**"ref"**)**

**try:**

obj **=** Join**.**objects**.**get**(**ref\_id **=** ref\_id**)**

**except:**

obj **=** **None**

**if** obj**:**

request**.**session**[**'join\_id\_ref'**]** **=** obj**.**id

**def** home**(**request**):**

**try:**

join\_id **=** request**.**session**[**'join\_id\_ref'**]**

obj **=** Join**.**objects**.**get**(**id**=**join\_id**)**

**except:**

obj **=** **None**

form **=** JoinForm**(**request**.**POST **or** **None)**

**if** form**.**is\_valid**():**

new\_join **=** form**.**save**(**commit **=** **False)**

email **=** form**.**cleaned\_data**[**'email'**]**

new\_join\_old**,** created **=** Join**.**objects**.**get\_or\_create**(**email **=** email**)**

**if** created**:**

new\_join\_old**.**ref\_id **=** get\_ref\_id**()**

new\_join\_old**.**ip\_address **=** get\_ip**(**request**)**

new\_join\_old**.**save**()**

**return** HttpResponseRedirect**(**"/%s" **%(**new\_join\_old**.**ref\_id**))**

context **=** **{**"form"**:**form**}**

template **=** "home.html"

**return** render**(**request**,** template**,** context**)**

# Save Reference ID with Foreign Key Model Fields

添加Friend，用reference id作为索引

* 在join model中添加friend字段
* 在view中将join object存为friend

**class** **Join(**models**.**Model**):**

email **=** models**.**EmailField**()**

friend **=** models**.**ForeignKey**(**"self"**,** related\_name**=**'referral'**,** null**=True,** blank**=True)**

ref\_id **=** models**.**CharField**(**max\_length**=**120**,** default**=**'ABC'**,** unique**=True)**

ip\_address **=** models**.**CharField**(**max\_length **=** 120**,** default **=** 'ABC'**)**

timestamp **=** models**.**DateTimeField**(**auto\_now\_add **=** **True,** auto\_now **=** **False)**

updated **=** models**.**DateTimeField**(**auto\_now\_add **=** **False,** auto\_now **=** **True)**

**def** \_\_unicode\_\_**(**self**):**

**return** "%s %s" **%(**self**.**email**,** self**.**ref\_id**)**

**class** **Meta:**

unique\_together **=** **(**"email"**,** "ref\_id"**,)**

**def** home**(**request**):**

**try:**

join\_id **=** request**.**session**[**'join\_id\_ref'**]**

obj **=** Join**.**objects**.**get**(**id**=**join\_id**)**

**except:**

obj **=** **None**

form **=** JoinForm**(**request**.**POST **or** **None)**

**if** form**.**is\_valid**():**

new\_join **=** form**.**save**(**commit **=** **False)**

email **=** form**.**cleaned\_data**[**'email'**]**

new\_join\_old**,** created **=** Join**.**objects**.**get\_or\_create**(**email **=** email**)**

**if** created**:**

new\_join\_old**.**ref\_id **=** get\_ref\_id**()**

**if** **not** obj **==** **None:**

new\_join\_old**.**friend **=** obj

new\_join\_old**.**ip\_address **=** get\_ip**(**request**)**

new\_join\_old**.**save**()**

**return** HttpResponseRedirect**(**"/%s" **%(**new\_join\_old**.**ref\_id**))**

context **=** **{**"form"**:**form**}**

template **=** "home.html"

**return** render**(**request**,** template**,** context**)**

# Create a Social Sharing Page Part 2

分享页的显示

SHARE\_URL **=** "http://127.0.0.1:8000/?ref="

**from** django**.**shortcuts **import** render**,** HttpResponseRedirect**,** Http404

**from** django**.**conf **import** settings

**def** share**(**request**,** ref\_id**):**

**try:**

join\_obj **=** Join**.**objects**.**get**(**ref\_id**=**ref\_id**)**

friends\_referred **=** Join**.**objects**.**filter**(**friend**=**join\_obj**)**

count **=** join\_obj**.**referral**.**all**().**count**()**

ref\_url **=** settings**.**SHARE\_URL **+** str**(**join\_obj**.**ref\_id**)**

context **=** **{**"ref\_id"**:** join\_obj**.**ref\_id**,** "count"**:** count**,** "ref\_url"**:** ref\_url**}**

template **=** "share.html"

**return** render**(**request**,** template**,** context**)**

**except:**

**raise** Http404

# Load Static Files (CSS, JS, & Images) in Django

静态文件处理

* 定义文件路径
* 在html文件中添加静态文件处理

# Absolute filesystem path to the directory that will hold user-uploaded files.

# Example: "/var/www/example.com/media/"

MEDIA\_ROOT **=** os**.**path**.**join**(**BASE\_DIR**,** 'static'**,**'media'**)**

# URL that handles the media served from MEDIA\_ROOT. Make sure to use a

# trailing slash.

# Examples: "http://example.com/media/", "http://media.example.com/"

MEDIA\_URL **=** '/media/'

# Absolute path to the directory static files should be collected to.

# Don't put anything in this directory yourself; store your static files

# in apps' "static/" subdirectories and in STATICFILES\_DIRS.

# Example: "/var/www/example.com/static/"

STATIC\_ROOT **=** os**.**path**.**join**(**BASE\_DIR**,** 'static'**,**'static\_root'**)**

# URL prefix for static files.

# Example: "http://example.com/static/", "http://static.example.com/"

STATIC\_URL **=** '/static/'

# Additional locations of static files

STATICFILES\_DIRS **=** **(**

# Put strings here, like "/home/html/static" or "C:/www/django/static".

# Always use forward slashes, even on Windows.

# Don't forget to use absolute paths, not relative paths.

os**.**path**.**join**(**BASE\_DIR**,** 'static'**,**'static\_dirs'**),**

**)**

base.html

**{% load staticfiles %}**

<link href = **"{% static 'css/bootstrap.min.css' %}"** rel = **"stylesheet"**>

<link href = **"{% static 'css/jumbotron.css' %}"** rel = **"stylesheet"**>

<link href = **"{% static 'js/bootstrap.min.css' %}"** rel = **"stylesheet"**>

<div class=**"jumbotron"**>

<div class=**"container"**>

<div class=**"col-sm-6"**>

<h1>**Hello, world!**</h1>

<p>**This is a template for a simple marketing or informational website. It includes a large callout called a jumbotron and three supporting pieces of content. Use it as a starting point to create something more unique.**</p>

<p><a class=**"btn btn-success btn-lg"** href=**"#"** role=**"button"**>**Learn more** &raquo;</a></p>

</div>

<div class=**"col-sm-6"**>

<img src = **"{% static 'img/iphone.png' %}"** class = **"img-responsive"** />

</div>

</div>

</div>

# Update Front End Design with CSS

前段用css进行美化

将下面从base.html移到home.html

背景改成白色，与照片一致，不然会有色差

<!-- Main jumbotron for a primary marketing message or call to action -->

<div class=**"jumbotron"** style=**"background-color:white"**>

<div class=**"container"**>

<div class=**"col-sm-6"**>

<h1>**Hello, world!**</h1>

<p>**This is a template for a simple marketing or informational website. It includes a large callout called a jumbotron and three supporting pieces of content. Use it as a starting point to create something more unique.**</p>

<p><a class=**"btn btn-success btn-lg"** href=**"#"** role=**"button"**>**Learn more** &raquo;</a></p>

</div>

<div class=**"col-sm-6"**>

<img src = **"{% static 'img/iphone.png' %}"** class = **"img-responsive"** />

</div>

</div>

</div>

share.html

**{% include "navbar.html" %}**

<!-- Main jumbotron for a primary marketing message or call to action -->

<div class=**"jumbotron"**>

<div class=**"container"**>

<div class=**"row"**>

<div class=**"col-md-6 pull-right"** >

<h1>**Hello, world!**</h1>

<p>**This is a template for a simple marketing or informational website. It includes a large callout called a jumbotron and three supporting pieces of content. Use it as a starting point to create something more unique.**</p>

<p><a class=**"btn btn-success btn-lg"** href=**"#"** role=**"button"**>**Learn more** &raquo;</a></p>

</div>

<div class=**"col-md-6 pull-left"**>

<img src = **"{% static 'img/launch.jpg' %}"** class = **"img-responsive"** />

</div>

</div>

</div>

</div>

navbar.html

<nav class=**"navbar navbar-inverse"**>

<div class=**"container"**>

<div class=**"navbar-header"**>

<button type=**"button"** class=**"navbar-toggle collapsed"** data-toggle=**"collapse"** data-target=**"#navbar"** aria-expanded=**"false"** aria-controls=**"navbar"**>

<span class=**"sr-only"**>**Toggle navigation**</span>

<span class=**"icon-bar"**></span>

<span class=**"icon-bar"**></span>

<span class=**"icon-bar"**></span>

</button>

<a class=**"navbar-brand"** href=**"#"**>**Project name**</a>

</div>

</div>

</nav>

<style>

**.navbar{**

**margin-bottom:0px:important;**

**border-radius:0px:important;**

**}**

**.navbar-inverse .navbar-brand, .navbar-brand:hover{**

**color:white;**

**}**

</style>

# Create a Social Sharing Page Part 3

<https://www.codingforentrepreneurs.com/>

创建到facebook，wechat等的分享

<http://fortawesome.github.io/Font-Awesome/>

<http://fontawesome.io/icon/facebook-square/>

share.html

<div class=**"jumbotron"**>

<div class=**"container"**>

<div class=**"row"**>

<div class=**"col-md-6 pull-right"** style=**"text-align:center"** >

<i class=**"fa fa-share-alt fa-5x"**></i>

<h1>**Inivte Friend** &amp; **Earn Rewards**</h1>

<p>**Share your unique link below via Email, Facebook, Twitter, Reddit, or LinkedIn to earn rewards from LaunchWithCode.com**</p>

<div class=**"well"** style=**"background-color:white"**>**{{ref\_url}}**</div>

<br/>**Share**<br/>

<i class=**"fa fa-facebook-square fa-4x"** style=**"color:#213E75"**></i>

<i class=**"fa fa-twitter-square fa-4x"** style=**"color:#D42665"**></i>

<i class=**"fa fa-linkedin-square fa-4x"** style=**"color:#5413B9"**></i>

<i class=**"fa fa-reddit-square fa-4x"** style=**"color:#75213F"**></i>

</div>

<div class=**"col-md-6 pull-left"**>

<img src = **"{% static 'img/launch.jpg' %}"** class = **"img-responsive"** />

</div>

</div>

</div>

</div>

Icon CDN

<https://www.bootstrapcdn.com/fontawesome/>

<link href="https://maxcdn.bootstrapcdn.com/font-awesome/4.6.3/css/font-awesome.min.css" rel="stylesheet">

<http://www.colourlovers.com/>

进度条

<http://getbootstrap.com/components/#progress>

<div class=**"container"** style=**"text-align:center"**>

**Joined number is: {{count}}** <br/>

<div class=**"row"**>

<div class=**"col-sm-3"**>

<h3>**5 Joined**</h3>

<div class=**"progress"**>

<div class=**"progress-bar"** role=**"progressbar"** aria-valuenow=**"60"** aria-valuemin=**"0"** aria-valuemax=**"100"** style=**"width: 60%;"**>

**60%**

</div>

</div>

<h4>**Rewards Name**</h4>

<img src="http://www.icon100.com/up/4257/72/BattleNet.png"/>

<img src="http://www.icon100.com/up/4252/72/32-extraterrestrial-head.png"/>

<hr/>

<span style=**"text-align:left!important"**>

<h5>**Rewards Include:**</h5>

<ul>

<li> **Reward 1**</li>

<li> **Reward 1**</li>

</ul>

</span>

<hr>

</div>

# Use jQuery to make Bootstrap progress bars function

base.html

<script>

$**(**document**).**ready**(function(){**

**{%** block jquery **%}**

**{%** endblock **%}**

**});**

</script>

# Implement Social Sharing HTML Buttons

<https://github.com/codingforentrepreneurs/Guides/blob/master/all/social_share_links.md>

Basic URL Text Encoder: <http://meyerweb.com/eric/tools/dencoder/>

<http://blog.csdn.net/p793049488/article/details/45666279>

微博分享

<a href="http://v.t.sina.com.cn/share/share.php?title=%E4%BD%A0%E5%A5%BD%20hello%20world&url={{ ref\_url }}" target="\_blank">

<i class="fa fa-weibo fa-4x" aria-hidden="true"></i></a>

微信分享

<http://dev.wechat.com/wechatapi/messages-moments#title-Links>

# CSS Background Image & Styles & Parallax

<http://www.w3school.com.cn/cssref/pr_background-size.asp>

在base.html

{% block styles %}

{% endblock %}

在home.html

{% block styles %}

<style>

body {

border-top: 4px solid #E47541;

/\*min-height: 5000px;

/\*min-width: 5000px;\*/

background: url('{% static "img/beach.jpg" %}') no-repeat center center fixed;

-webkit-background-size: cover;

-moz-background-size: cover;

-o-background-size: conver;

background-size: cover;

/\*background-image: url('{% static "img/beach.jpg" %}');\*/

/\*background-image:url('https://lh4.googleusercontent.com/-FsgMNhfQLSg/AAAAAAAAAAI/AAAAAAAAaRY/3VnF3vSuWKk/photo.jpg');\*/

/\*background-repeat: repeat-x;\*/

}

</style>

{% endblock %}

# Update Email Form Design to Bootstrap Inline Form

将email从form.as\_p改到bootstrap内联的form

<http://getbootstrap.com/css/#forms>

<div class=**'container'** style=**'text-align:center' "** >

<div class=**'col-sm-6 col-sm-offset-3'**>

<h1> **Welcome Home!** </h1>

<form class=**'form-horizontal'** method=**"POST"** action=**""**> **{% csrf\_token %}**

<div class=**'form-group form-group-lg'**>

**{% if form.email.errors %}**

**{% for err in form.email.errors %}**

<div class=**"alert alert-danger"** role=**"alert"**>**{{ err }}**</div>

<div class=**"alert alert-danger"** role=**"alert"**>**Your Email is required**</div>

**{% endfor %}**

**{% endif %}**

<div class=**"input-group"**>

<input class=**'form-control'** type=**"email"** name=**"email"** placeholder=**"Your email..."** />

<span class=**'input-group-btn'**>

<input type=**'submit'** value=**'Join'** class=**'btn btn-primary btn-lg'** />

</span>

</div>

</div>

</form>

</div>

</div>

# Final Styles Update to Complete Bootstrap Design

修饰登录部分的css

<div class=**"jumbotron"** style=**"background-color:white"**>

<div class=**"container"**>

<div class=**"col-sm-6"**>

<h1>**Hello, world!**</h1>

<p>**Use it as a starting point to create something more unique.**</p>

<p><a class=**"btn btn-default btn-lg btn-block"** href=**"#**signup**"** role=**"button"**>**Be First** &raquo;</a></p>

</div>

<div class=**"col-sm-6"**>

<img src = **"{% static 'img/iphone.png' %}"** class = **"img-responsive"** />

</div>

</div>

</div>

<div class=**'container'** style=**'text-align:center; color:#E47541;'** id=**"signup"** >

<i class=**"fa fa-unlock fa-5x"**></i>

<h1>**Unlock First**</h1>

<form class=**'form-horizontal'** method=**"POST"** action=**""**> **{% csrf\_token %}**

<div class=**'form-group form-group-lg'**>

**{% if form.email.errors %}**

**{% for err in form.email.errors %}**

<div class=**"alert alert-danger"** role=**"alert"**>**{{ err }}**</div>

<div class=**"alert alert-danger"** role=**"alert"**>**Your Email is required**</div>

**{% endfor %}**

**{% endif %}**

<div class=**"input-group"**>

<input class=**'form-control'** type=**"email"** name=**"email"** placeholder=**"Your email..."** />

<span class=**'input-group-btn'**>

<input type=**'submit'** value=**'Join'** class=**'btn btn-primary btn-lg'** />

</span>

</div>

</div>

</form>

<p class=**'lead'**>**Sign up here.** <br/> **Get First News on Content**</p></div>

</div>

</div>

.jumbotron h1 {

font-weight: 100;

}

.jumbotron p {

font-weight: 100;

font-size: 30px;

}

添加一些其他的feature row，主要是css操作，需要练习

# Prepare for Production using Heroku.com as Server

HTTP 错误 400

400 请求出错

由于语法格式有误，服务器无法理解此请求。不作修改，客户程序就无法重复此请求。

HTTP 错误 401

401.1 未授权：登录失败

此错误表明传输给服务器的证书与登录服务器所需的证书不匹配。

请与 Web 服务器的管理员联系，以确认您是否具有访问所请求资源的权限。

401.2 未授权：服务器的配置导致登录失败

此错误表明传输给服务器的证书与登录服务器所需的证书不匹配。此错误通常由未发送正确的 WWW 验证表头字段所致。

请与 Web 服务器的管理员联系，以确认您是否具有访问所请求资源的权限。

401.3 未授权：由于资源中的 ACL 而未授权

此错误表明客户所传输的证书没有对服务器中特定资源的访问权限。此资源可能是客户机中的地址行所列出的网页或文件，也可能是处理客户机中的地址行所列出的文件所需服务器上的其他文件。

请记录试图访问的完整地址，并与 Web 服务器的管理员联系以确认您是否具有访问所请求资源的权限。

401.4 未授权：授权服务被筛选程序拒绝

此错误表明 Web 服务器已经安装了筛选程序，用以验证连接到服务器的用户。此筛选程序拒绝连接到此服务器的真品证书的访问。

请记录试图访问的完整地址，并与 Web 服务器的管理员联系以确认您是否具有访问所请求资源的权限。

401.5 未授权：ISAPI/CGI 应用程序的授权失败

此错误表明试图使用的 Web服务器中的地址已经安装了 ISAPI 或 CGI程序，在继续之前用以验证用户的证书。此程序拒绝用来连接到服务器的真品证书的访问。

请记录试图访问的完整地址，并与 Web服务器的管理员联系以确认您是否具有访问所请求资源的权限

HTTP 错误 403

403.1 禁止：禁止执行访问

如果从并不允许执行程序的目录中执行 CGI、ISAPI或其他执行程序就可能引起此错误。

如果问题依然存在，请与 Web 服务器的管理员联系。

403.2 禁止：禁止读取访问

如果没有可用的默认网页或未启用此目录的目录浏览，或者试图显示驻留在只标记为执行或脚本权限的目录中的HTML 页时就会导致此错误。

如果问题依然存在，请与 Web 服务器的管理员联系。

403.3 禁止：禁止写访问

如果试图上载或修改不允许写访问的目录中的文件，就会导致此问题。

如果问题依然存在，请与 Web服务器的管理员联系。

403.4 禁止：需要 SSL

此错误表明试图访问的网页受安全套接字层（SSL）的保护。要查看，必须在试图访问的地址前输入https:// 以启用 SSL。

如果问题依然存在，请与 Web服务器的管理员联系。

403.5 禁止：需要 SSL 128

此错误消息表明您试图访问的资源受 128位的安全套接字层（SSL）保护。要查看此资源，需要有支持此SSL 层的浏览器。

请确认浏览器是否支持 128 位 SSL安全性。如果支持，就与 Web服务器的管理员联系，并报告问题。

403.6 禁止：拒绝 IP 地址

如果服务器含有不允许访问此站点的 IP地址列表，并且您正使用的 IP地址在此列表中，就会导致此问题。

如果问题依然存在，请与 Web服务器的管理员联系。

403.7 禁止：需要用户证书

当试图访问的资源要求浏览器具有服务器可识别的用户安全套接字层（SSL）证书时就会导致此问题。可用来验证您是否为此资源的合法用户。

请与 Web服务器的管理员联系以获取有效的用户证书。

403.8 禁止：禁止站点访问

如果 Web服务器不为请求提供服务，或您没有连接到此站点的权限时，就会导致此问题。

请与 Web 服务器的管理员联系。

403.9 禁止访问：所连接的用户太多

如果 Web太忙并且由于流量过大而无法处理您的请求时就会导致此问题。请稍后再次连接。

如果问题依然存在，请与 Web 服务器的管理员联系。

403.10 禁止访问：配置无效

此时 Web 服务器的配置存在问题。

如果问题依然存在，请与 Web服务器的管理员联系。

403.11 禁止访问：密码已更改

在身份验证的过程中如果用户输入错误的密码，就会导致此错误。请刷新网页并重试。

如果问题依然存在，请与 Web服务器的管理员联系。

403.12 禁止访问：映射程序拒绝访问

拒绝用户证书试图访问此 Web 站点。

请与站点管理员联系以建立用户证书权限。如果必要，也可以更改用户证书并重试。

HTTP 错误 404

404 找不到

Web 服务器找不到您所请求的文件或脚本。请检查URL 以确保路径正确。

如果问题依然存在，请与服务器的管理员联系。

HTTP 错误 405

405 不允许此方法

对于请求所标识的资源，不允许使用请求行中所指定的方法。请确保为所请求的资源设置了正确的 MIME 类型。

如果问题依然存在，请与服务器的管理员联系。

HTTP 错误 406

406 不可接受

根据此请求中所发送的“接受”标题，此请求所标识的资源只能生成内容特征为“不可接受”的响应实体。

如果问题依然存在，请与服务器的管理员联系。

HTTP 错误 407

407 需要代理身份验证

在可为此请求提供服务之前，您必须验证此代理服务器。请登录到代理服务器，然后重试。

如果问题依然存在，请与 Web 服务器的管理员联系。

HTTP 错误 412

412 前提条件失败

在服务器上测试前提条件时，部分请求标题字段中所给定的前提条件估计为FALSE。客户机将前提条件放置在当前资源 metainformation（标题字段数据）中，以防止所请求的方法被误用到其他资源。

如果问题依然存在，请与 Web 服务器的管理员联系。

HTTP 错误 414

414 Request-URI 太长

Request-URL太长，服务器拒绝服务此请求。仅在下列条件下才有可能发生此条件：

客户机错误地将 POST 请求转换为具有较长的查询信息的 GET 请求。

客户机遇到了重定向问题（例如，指向自身的后缀的重定向前缀）。

服务器正遭受试图利用某些服务器（将固定长度的缓冲区用于读取或执行 Request-URI）中的安全性漏洞的客户干扰。

如果问题依然存在，请与 Web 服务器的管理员联系。

HTTP 错误 500

500 服务器的内部错误

Web 服务器不能执行此请求。请稍后重试此请求。

如果问题依然存在，请与 Web服务器的管理员联系。

HTTP 错误 501

501 未实现

Web 服务器不支持实现此请求所需的功能。请检查URL 中的错误，如果问题依然存在，请与 Web服务器的管理员联系。

HTTP 错误 502

502 网关出错

当用作网关或代理时，服务器将从试图实现此请求时所访问的upstream 服务器中接收无效的响应。

如果问题依然存在，请与 Web服务器的管理员联系。

<http://getbootstrap.com/examples/cover/>

这一页作为400,500的base

创建400,404,500文件

注意：只有把Debug设为False，标准的404才能显示

You're seeing this error because you have DEBUG = True in your Django settings file. Change that to False, and Django will display a standard 404 page.

# Hosts/domain names that are valid for this site; required if DEBUG is False

# See https://docs.djangoproject.com/en/1.5/ref/settings/#allowed-hosts

ALLOWED\_HOSTS = ['\*']

DEBUG = False

TEMPLATE\_DEBUG = True

如果Debug设为False，ALLOWED\_HOST一定要设置，否则会抛出404

url.py

handler404 = 'joins.views.server\_error'

handler403 = 'joins.views.server\_error'

handler500 = 'joins.views.server\_error'

**from** django**.**template **import** RequestContext

**from** django**.**shortcuts **import** render\_to\_response

**def** server\_error**(**request**,**template\_name**=**'404.html'**):**

**return** render\_to\_response**(**template\_name**,** context\_instance**=**RequestContext**(**request**))**

# Launch Your Project to Heroku & Setup Domain Name