# Django构建一个xinwen

## 在pycharm中创建django的步骤：

1. 新建选择django类型
2. Create（）选择的编译器可以根据需要
3. 在settings里面可以配置相应的需要的包
4. Terminal中 python manage.py runserver进行验证
5. Setting中中英文设置

LANGUAGE\_CODE='zh-hans'  
TIME\_ZONE = 'Asia/Shanghai'

## Mysql与django配置

1. 将settings中的database配置为mysql设置端口号并**import os**等 **python manage.py runserver**

DATABASES = {  
 **'default'**: {  
 **'ENGINE'**: **'django.db.backends.mysql'**,  
 **'NAME'**: **'uploadimage'**,  
 **'HOST'**: **'127.0.0.1'**,  
 **'PORT'**: 3306,  
 **'USER'**: **'root'**,  
 **'PASSWORD'**: **'yyf314989',** }  
}

**出现“：是否安装mysqlclient**

**添加：**import pymysql  
pymysql.install\_as\_MySQLdb() 语句在init文件中

1. 打开旁边的数据库设置添加用户名密码数据库类型等
2. 打开另一个terminal python，输入**python manage.py migrate**
3. 出现Server returns invalid timezone. Go to 'Advanced' tab and set 'serverTimezone' property manually.

在设置里面的advanced里面新建一栏 **serverTimezone CST**

1. 访问<http://127.0.0.1:8000/admin/login/?next=/admin/>

在T2里面创建超级管理员 **python manage.py createsuperuser**

## 三、创建组件

1)T2 输入**Python manage.py startapp xinwen出现文件夹xinwen**

**2）**在根目录/setting.py文件中找到INSTALLED\_APPS（**功能引进项目**）

加入我们的文件 格式：功能名称.apps.功能名称Config,(功能名称Config在文件夹中的apps中)

3）功能数据写入数据库 mysql

**T2 python manage.py makemigrations 功能名称** 此时在该文件夹下的migrations文件夹下多出来一个0001——initial.py(**用于将数据存到数据库中**)

4）将数据引入本地mysql

再次输入**python manage.py migrate**

5）重启**python manage.py runserver**

6）在目录里面的admin文件进入后台管理

from django.contrib import admin  
from .models import xinwen

admin.site.register(xinwen,XinwenAdmin)*#前面写模型名称，后面写命名的组件配置*

1. 进入<http://127.0.0.1:8000/admin/xinwen/xinwen/add/>可以在xinwen下面添加项目
2. 在mysql中可以看到xinwen——xinwen中出现了对应创建的内容

# Django 后端部分时间类型avi6

1. 打开pycharm后端的models

在class中加入

from datetime import datetime

class中加入

add\_time = models.DateTimeField(default=datetime.now, verbose\_name=**u"添加时间"**)

其中（, verbose\_name=**u"添加时间"）**在其他项也可以用

1. 生成数据库对接文件

新建一个T 输入**python manage.py makemigrations**

在xinwen/migrations/文件夹下出现了新的0002\_xinwen\_addtime文件

1. 对接数据库

**python manage.py migrate**

1. 重启数据库

**python manage.py runserver**

**5)**进入<http://127.0.0.1:8000/admin/xinwen/xinwen/add/>

选择xinwen增加

1. 进入<http://127.0.0.1:8000/xinwen/>

发现没有时间，还需要到serializers添加

# Django后端上传图片avi7

1. 在models的 class中添加

img\_url = models.ImageField(upload\_to==**'img\_url/'**,verbose\_name=**"图片地址"**)

#上传地址=img\_url

img\_url = models.ImageField(upload\_to=**r'C:\Users\86182\Desktop\PythonMission\Part7\practice\djangoProject\upimg\img\_url/'**,verbose\_name=**u"图片地址"**)  
*#该图片所的绝对路径*

实现保存图片的绝对地址

2)出现NameError: name 'upload\_to' is not defined

打开 T2 输入 **pip install Pillow**

1. **在serializers中添加显示**

4) 生成数据库对接文件T2中输入

**python manage.py makemigrations**

对接数据库

**python manage.py migrate**

5）

打开：<http://127.0.0.1:8000/xinwen/>

添加了图片的显示路径"<http://127.0.0.1:8000/img_url/5524.jpg>

，未添加的显示为"<http://127.0.0.1:8000/111>",

打开：<http://127.0.0.1:8000/img_url/5524.jpg>

在pycharm中img\_url中有显示相应的图片

显示为空page not found ，需要对url进行设置

1. 在manage.py的同级目录下新建一文件夹upimg

7）设置在settings添加图片上传地址、图片数据库的引用位置

MEDIA\_URL =**"/upimg/"**MEDIA\_ROOT = os.path.join(BASE\_DIR,**'upimg'**) #加入数据的目录

1. 在django/ urls.py文件下加入

from djangoProject.settings import MEDIA\_ROOT#引入settings文件

from django.views.static import serve

在urlpatterns中加入

url(**r'^upimg/(?P<path>.\*)$'**,serve,{**"document\_root"**: MEDIA\_ROOT}),

1. 刷新<http://127.0.0.1:8000/img_url/5524.jpg>

出现网页仍不对

进入http://127.0.0.1:8000/admin/xinwen/xinwen/5/change/

在xinwen中新建图片后

进入<http://127.0.0.1:8000/> 接着进入 <http://127.0.0.1:8000/xinwen/>

点击对应的网址即可看到对应的图片

# Django后端时间显示设置avi8

在settings文件中

REST\_FRAMEWORK = {  
 **'DATETIME\_FORMAT'**:**"%Y-%m-%d %H:%M:%S"**,*#修改时间样式* **'DEFAULT\_PERMISSION\_CLASSES'**: [  
 **'rest\_framework.permissions.DjangoModelPermissionsOrAnonReadOnly'** ]  
}

重新启动python manage.py runserver