**尚硅谷Python**

（讲师：刘渊）

## 通过nginx+uwsgi将django部署在Ubuntu

**1、安装python3-pip**

Sudo apt-get install python3-pip

**2、安装虚拟环境和虚拟环境管理包**

Sudo pip3 install virtualenv (报错的话需要把pip也安装一下)

Sudo pip3 virtualenvwrapper

在~/.bashrc 添加

export WORKON\_HOME=~/Envs

source /usr/local/bin/virtualenvwrapper.sh

VIRTUALENVWRAPPER\_PYTHON=/usr/bin/python3

source .bashrc

创建虚拟环境：mkvirtualenv guliedu –p /usr/bin/python3

**3、虚拟环境当中安装项目所依赖的包（mysqlclient会出一个问题，需要首先安装下面这个包）**

我们可以通过 pip freeze > requirements.txt 将windows的虚拟环境安装包相应信息导出来移动到ubuntu

Pip3 install -r requirements.txt 在安装过程中mysqlclient会报错，按照下面两步走

sudo apt-get install libmysqlclient-dev

pip install mysqlclient

**4、安装MySQL数据库：**

Sudo apt-get install mysql-server

进入数据库创建自己的账户并且分配所有的权限，并且刷新权限，在配置文件当中bind 0.0.0.0

在ubuntu 当中创建我们用的数据库shangonline，通过windows的Navicat将windows数据库中的数据传输到ubuntu当中

将我们的项目文件夹拖入到我们的虚拟环境当中，进入项目python manage.py runserver保证能拉起项目

**5、安装uwsgi**

Pip3 install uwsgi

**6、测试uwsgi**

uwsgi --http :8000 --module GuLiEdu.wsgi

**7、安装配置nginx**

Sudo apt-get install nginx

安装完成会自动启动nginx的服务，我们在外部windows浏览器直接访问ubuntu ip 会进入到nginx的环境界面

在项目 根目录新建文件夹config

新建guli\_nginx.conf

# the upstream component nginx needs to connect to

upstream django {

# server unix:///path/to/your/mysite/mysite.sock; # for a file socket

server 127.0.0.1:8000; # for a web port socket (we'll use this first)

}

# configuration of the server

server {

# the port your site will be served on

listen 80;

# the domain name it will serve for

server\_name 你的ip地址 ; # substitute your machine's IP address or FQDN

charset utf-8;

# max upload size

client\_max\_body\_size 75M; # adjust to taste

# Django media

location /media {

alias /home/ly/ShangOnline/static/media; # 指向django的media目录

}

location /static {

alias /home/ly/ShangOnline/static; # 指向django的static目录

}

# Finally, send all non-media requests to the Django server.

location / {

uwsgi\_pass django;

include /etc/nginx/uwsgi\_params; # the uwsgi\_params file you installed

}

}

**8、将该配置文件加入到nginx的启动配置文件夹中**

sudo cp /home/ly/ShangOnline/config/sol\_nginx.conf /etc/nginx/conf.d/

**9、拉取所有需要的static file 到同一个目录**

在django的setting文件中，添加下面一行内容：把原来配的STATICFILES\_DIRS注释掉，在生产环境当中，这个配置失效

STATIC\_ROOT = os.path.join(BASE\_DIR, "static/")

运行命令

python manage.py collectstatic

**10、运行nginx**

sudo service nginx restart

**11、通过配置文件启动uwsgi**

新建uwsgi.ini 配置文件， 内容如下：

# mysite\_uwsgi.ini file

[uwsgi]

# Django-related settings

# the base directory (full path)

chdir = /home/ly/ShangOnline/

# Django's wsgi file

module = ShangOnline.wsgi

# the virtualenv (full path)

# process-related settings

# master

master = true

# maximum number of worker processes

processes = 10

# the socket (use the full path to be safe

socket = 127.0.0.1:8000

# ... with appropriate permissions - may be needed

# chmod-socket = 664

# clear environment on exit

vacuum = true

virtualenv = /home/ly/Envs/ShangOnline

logto = /tmp/mylog.log

注：

chdir： 表示需要操作的目录，也就是项目的目录

module： wsgi文件的路径

processes： 进程数

virtualenv：虚拟环境的目录

workon ShangOnline

uwsgi -i /home/ly/GuLiEdu/config/uwsgi.ini &

**12、访问**http://你的ip地址/