**python设置虚拟环境**

**windows虚拟环境操作**

1. 进入你想要创建python虚拟环境的目录下，新建了一个空的文件夹。
2. 使用pip工具安装virtualenv软件包(python –m pip install virtualenv)。
3. 使用virtualenv命令创建一个名字为newenv的新环境目录(python –m virtualenv –no-site-packages newenv)。
4. 查看新的python的文件目录结构。
5. 在命令行下进入newenv目录下的Scripts的目录下，进入虚拟环境；进入Scripts后输入activate回车即可。
6. 在虚拟环境安装你要开发应用的第三方包，进入虚拟环境后可以直接使用pip安装相关的软件包。
7. 退出虚拟环境直接输入deactivate即可。

<https://www.jb51.net/article/170070.htm>

<https://baijiahao.baidu.com/s?id=1649726683611551609&wfr=spider&for=pc>

**conda虚拟环境操作**

conda在线创建环境时：

1.查看已安装环境（显示所有的虚拟环境）：conda info –e 或 conda info –envs 或 conda env list

2. 创建指定python版本的xxxx虚拟环境：conda create -n xxxx python=3.5

3 .开启xxxx环境conda activate xxxx

4 . 关闭环境conda deactivate

conda离线创建环境时：

5.首先将错误信息里有一项是offline mode：false，改为offline mode：true：conda config --set offline true

6. 然后就可以创建环境了：conda create -n xxxx python=3.5

7. 默认安装好的初始环境下有这11个原始包

# Name Version Build Channel

ca-certificates 2019.11.27 0 defaults

certifi 2019.11.28 py37\_0 defaults

openssl 1.1.1d he774522\_3 defaults

pip 19.3.1 py37\_0 defaults

python 3.7.5 h8c8aaf0\_0 defaults

setuptools 42.0.2 py37\_0 defaults

sqlite 3.30.1 he774522\_0 defaults

vc 14.1 h0510ff6\_4 defaults

vs2015\_runtime 14.16.27012 hf0eaf9b\_0 defaults

wheel 0.33.6 py37\_0 defaults

wincertstore 0.2 py37\_0 defaults

8. 删除虚拟环境。

使用命令conda remove -n your\_env\_name(虚拟环境名称) --all， 即可删除。

9. 删除环境中的某个包。

使用命令conda remove --name your\_env\_name package\_name 即可。

**补充：安装时可能出现的bug：**

10. anaconda装环境遇到无法定位程序输入点OPENSSL\_sk\_new\_reserve……问题：首先进入Anaconda\DLLS目录，查看libssl-1\_1-x64.dll的日期；然后进入Anaconda\Library\bin目录，比较两者日期；如果不一致，将DLLS中的替换到bin中的；最后再执行conda create -n xxxx python=3.5

即可。

# 11.anaconda出现CondaHTTPError问题：(一般是设置代理的问题)

CondaHTTPError: HTTP 000 CONNECTION FAILED for url <https://repo.continuum.io/pkgs/free/noarch/repodata.json.bz2>

**Anaconda中jupyter切换虚拟环境**

## conda安装conda\_nb: conda install nb\_conda完成后jupyter网页上将多出conda标签

## 2.创建conda虚拟环境并进入(在已经创建的环境中安装好ipykernel)

conda create -n tensorflow python = x.x(python版本)

activate tensorflow

## 3.安装ipykernel模块

pip install ipykernel

# 将环境添加到ipython的kernel中

python -m ipykernel install --user --name tensorflow --display-name tf

执行完这个语句之后，会自动在目录【C:\ProgramData\jupyter\kernels】(类似)生成一个【adda】文件夹，里面有kernel.json文件（[将虚拟环境写入jupyter notebook的kernel中（kernel.json文件）](https://blog.csdn.net/Eric_Young/article/details/89428738" \l "jupyter_notebookkernelkerneljson_2" \t "_self)

## 4.重新启动notebook,在kernel -> change kernel中即可切换到指定的虚拟环境。

如果经常需要用jupyter notebook，那么最好在创建虚拟环境的时候便安装好ipykernel，命令如下：

conda create -n python2 python=2.7 ipykernel

<https://blog.csdn.net/Mrxxyz/article/details/84866504>

<https://blog.csdn.net/weixin_41813895/article/details/84750990>

<https://www.cnblogs.com/crxis/p/9133058.html>

？？？？

C:\Users\shuangyang.luo\AppData\Roaming\jupyter\kernels

C:\ProgramData\jupyter

C:\Users\shuangyang.luo\AppData\Local\Continuum\anaconda3\share\jupyter\kernels