## 创建虚拟conda环境

### 1.创建

1. 打开anaconda prompt
2. 输入conda create -n env\_name python=3.7（输入python版本）

Note：

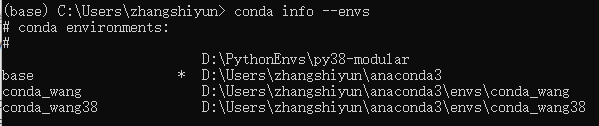
* 1. 创建指定位置下的虚拟环境：

# C:/ProgramData/Anaconda3/envs/环境所在目录，pytorch为环境名

ciC:/ProgramData/Anaconda3/envs/pytorch python=3.8

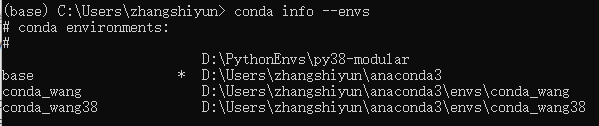
* 1. 克隆base环境

conda info --envs（查看conda环境）



conda create -n 新环境的名称 --clone 老环境名称

1. 查看虚拟环境输入：conda info -e



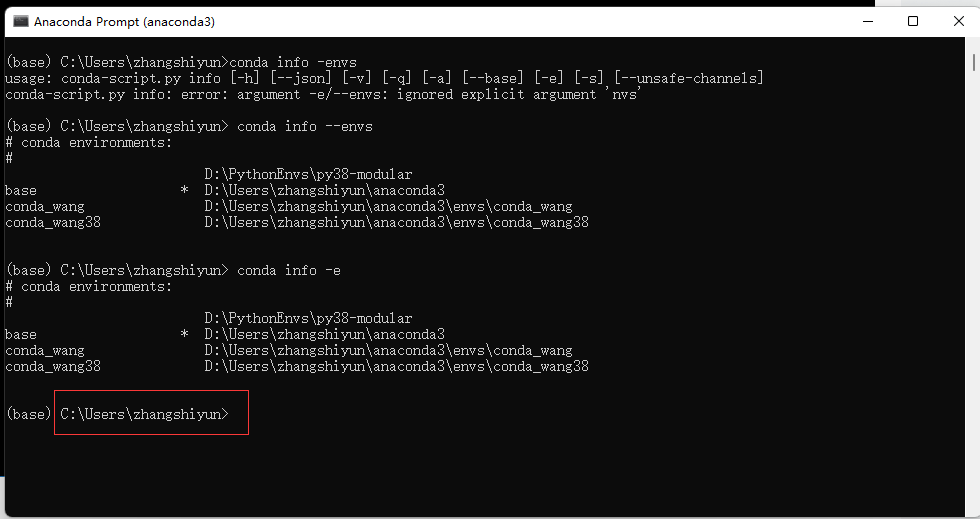
1. 复制包：（暂没有找的合适的）

### 2.复制先前的配置环境

#进入需要导出的虚拟环境中，env\_name为虚拟环境名

conda activate env\_name

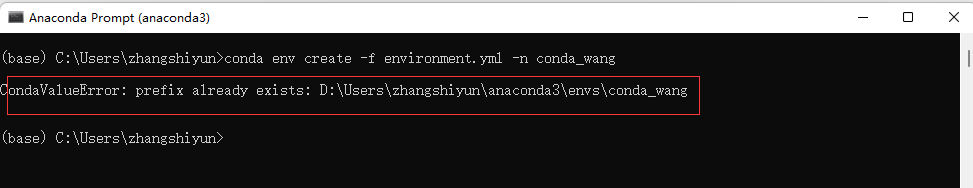
#输入：（导出环境到environment.yml文件中，文件前＋地址可导入到指定地址）#yml默认文件位置，yml与yaml文件效果相同



#输入（通过environment.yml复制一个与被复制的虚拟环境相同的环境）

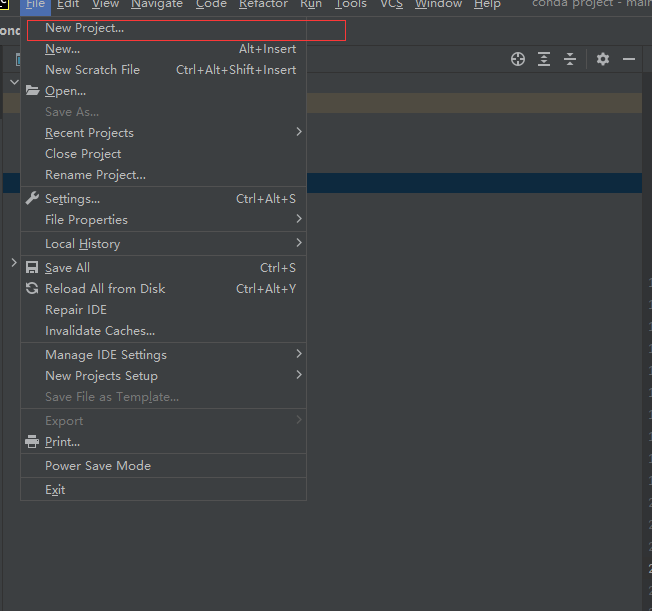
conda env create -f environment.yml -n new\_env\_name

# new\_env\_name必须为系统中不存在的虚拟环境，若失败（如下图）可能是系统已存在同名虚拟环境，重新命名new\_env\_name一个新的名即可，或者进入environment.yml（txt文件打开即可）修改首行name中的环境名

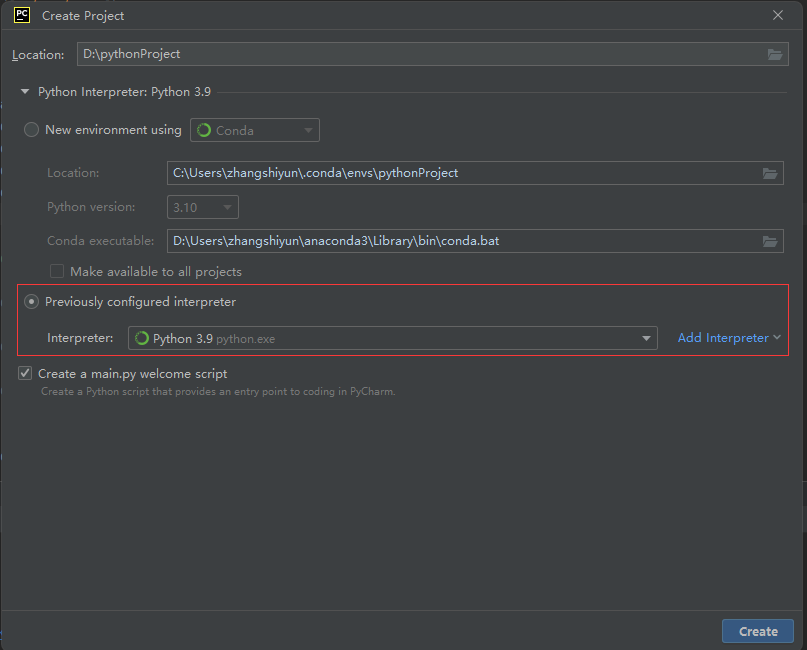


## 二.pycharm配置conda的虚拟环境

1. 创建新项目

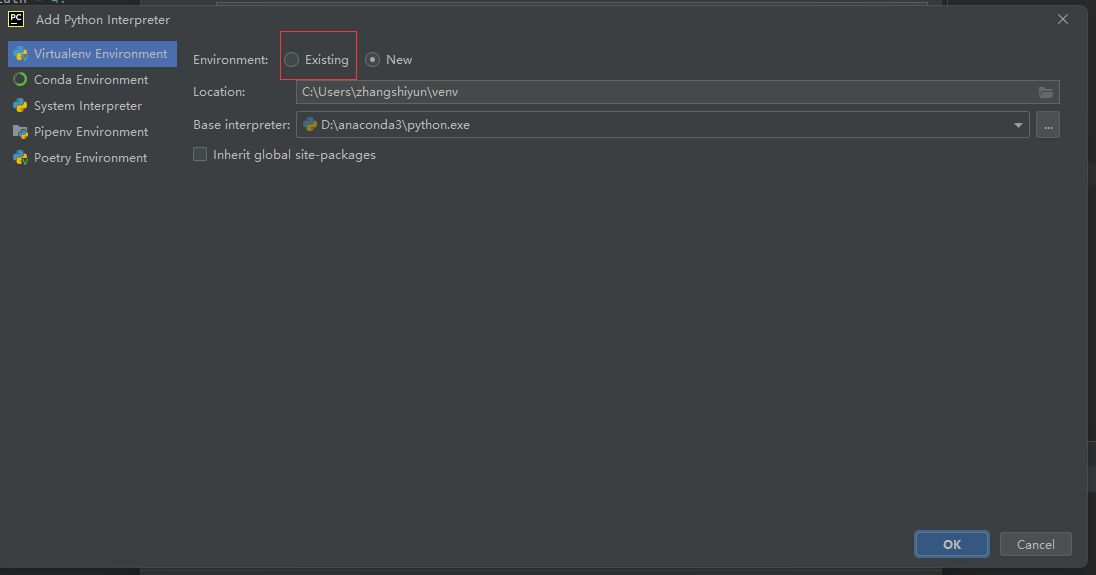


2.选择add interpreter

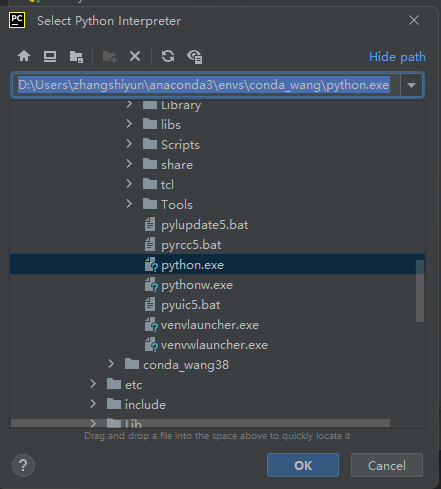


3.

选择已存在的



4.选择你创建conda虚拟环境路径下的python.exe文件。



## 三．安装包

#进入虚拟环境

conda activate env\_name

#安装包

conda install name(库的名字)

#若搜索不到，可以先添加清华镜像源，在进行安装库

conda config --add channels <https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/free/>

conda config --add channels <https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main/>

conda config --add channels <https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge/>

#显示添加的镜像源

conda config --show channels

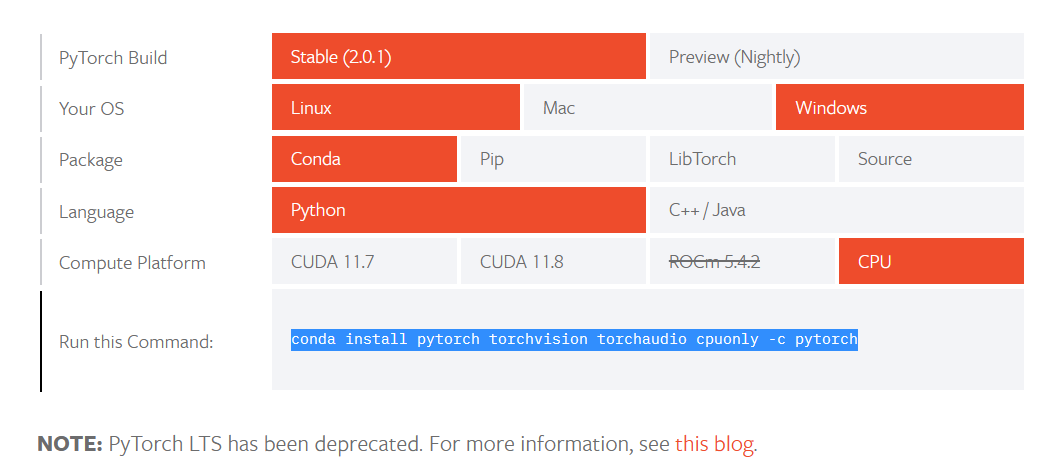
#删除指定源

conda config --remove channels

## 四.conda部分常用命令

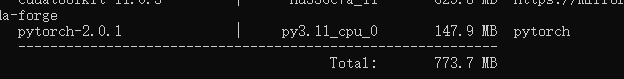


## 五、安装pytorch

1. 建立一个pytorch环境，
2. 

conda install pytorch torchaudio cpuonly -c pytorch

conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/pytorch/

1. 输入连接查看需要的版本，我当时时cpu，选择cuda是可以选择低版本的。
2. 

## 六、提示更新conda

1.conda install pytorch torchaudio cpuonly -c pytorch

2.点y确认

## 七、使用镜像源直接下载包

pip install -i https://pypi.tuna.tsinghua.edu.cn/simple pyspider，

conda install -c 镜像源 包名

pip install 包名 -i 镜像源

## 八、Django

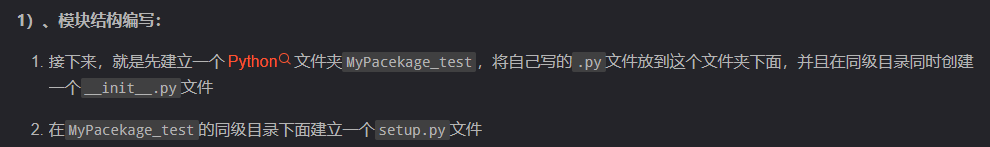
查看django版本：django-admin –version

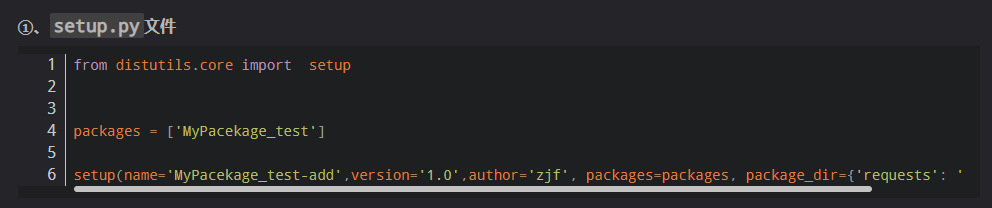
创建项目：**django-admin startproject 名称**

django-admin startproject site\_module

## python打包模块

网站：https://blog.csdn.net/weixin\_42081389/article/details/113251670?ops\_request\_misc=&request\_id=&biz\_id=102&utm\_term=python%20%E6%89%93%E5%8C%85%E6%88%90%E6%A8%A1%E5%9D%97&utm\_medium=distribute.pc\_search\_result.none-task-blog-2~all~sobaiduweb~default-5-113251670.142^v99^pc\_search\_result\_base6&spm=1018.2226.3001.4187





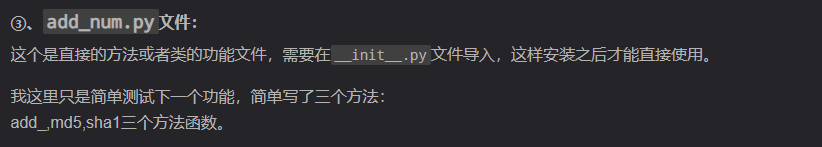
from distutils.core import setup

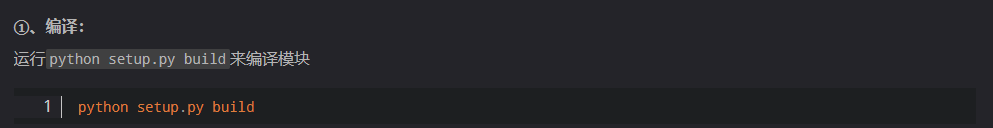
packages = ['MyPacekage\_test']

setup(name='MyPacekage\_test-add',version='1.0',author='zjf', packages=packages, package\_dir={'requests': 'requests'},)

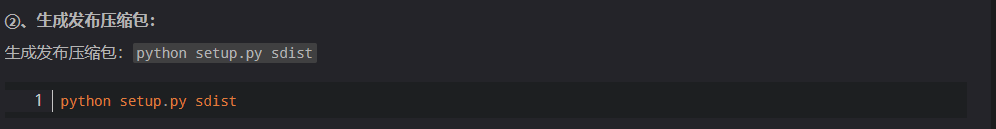


接下来自定义导出的py文件





python setup.py build



python setup.py sdist



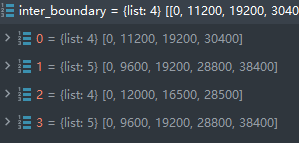
tar -zxvf message-1.0.tar.gz #解压

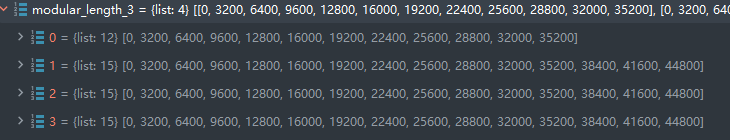
python3 setup.py install #安装

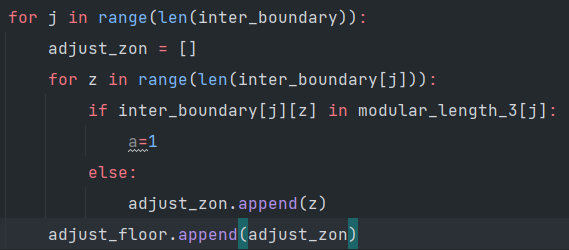
最后解压后的文件提取，删除外面的文件夹

## Python测试

python manage.py test







adjust\_zon = [1,3]

## git问题汇总

1. ssh: connect to host github.com port 22: Connection timed outfatal: Could not read from remote repository.



$ git config --global "url.ssh://git@ssh.github.com:443/.insteadOf" [git@github.com](mailto:git@github.com):

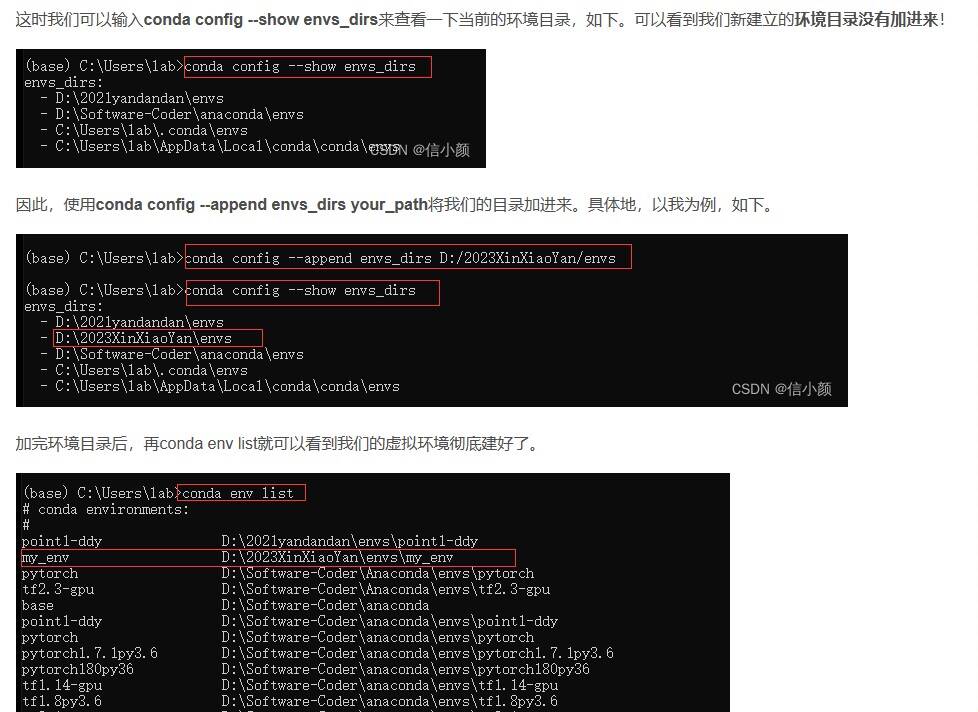
### Git 443报错

git config --global https.proxy

git config --global --unset https.proxy

git config --global --unset http.proxy

## conda环境名加目录



## 十. Node软件下载及环境配置,bug解决

下载安装及配置环境:

https://blog.csdn.net/WHF\_\_/article/details/129362462#:~:text=%EF%BC%881%EF%BC%89%E4%B8%8B%E8%BD%BD%E5%AE%8C%E6%88%90%E5%90%8E%EF%BC%8C%E5%8F%8C%E5%87%BB%E5%AE%89%E8%A3%85%E5%8C%85%EF%BC%8C%E5%BC%80%E5%A7%8B%E5%AE%89%E8%A3%85Node.js%20%282%29%E7%9B%B4%E6%8E%A5%E7%82%B9%E3%80%90Next%E3%80%91%E6%8C%89%E9%92%AE%EF%BC%8C%E6%AD%A4%E5%A4%84%E5%8F%AF%E6%A0%B9%E6%8D%AE%E4%B8%AA%E4%BA%BA%E9%9C%80%E6%B1%82%E4%BF%AE%E6%94%B9%E5%AE%89%E8%A3%85%E8%B7%AF%E5%BE%84%EF%BC%8C%E4%BF%AE%E6%94%B9%E5%AE%8C%E6%AF%95%E5%90%8E%E7%BB%A7%E7%BB%AD%E7%82%B9%E5%87%BB%E3%80%90Next%E3%80%91%E6%8C%89%E9%92%AE%20%EF%BC%883%EF%BC%89%E5%8F%AF%E6%A0%B9%E6%8D%AE%E8%87%AA%E8%BA%AB%E9%9C%80%E6%B1%82%E8%BF%9B%E8%A1%8C%EF%BC%8C%E6%AD%A4%E5%A4%84%E6%88%91%E9%80%89%E6%8B%A9%E9%BB%98%E8%AE%A4%E5%AE%89%E8%A3%85%EF%BC%8C%E7%BB%A7%E7%BB%AD%E7%82%B9%E5%87%BB%E3%80%90Next%E3%80%91%E6%8C%89%E9%92%AE%20%EF%BC%884%EF%BC%89%E4%B8%8D%E9%80%89%E4%B8%AD%EF%BC%8C%E7%9B%B4%E6%8E%A5%E7%82%B9%E5%87%BB%E3%80%90Next%E3%80%91%E6%8C%89%E9%92%AE%20%EF%BC%885%EF%BC%89%E7%82%B9%E5%87%BB%E3%80%90Install%E3%80%91%E6%8C%89%E9%92%AE%E8%BF%9B%E8%A1%8C%E5%AE%89%E8%A3%85,%EF%BC%886%EF%BC%89%E5%AE%89%E8%A3%85%E5%AE%8C%E6%AF%95%EF%BC%8C%E7%82%B9%E5%87%BB%E3%80%90Finish%E3%80%91%E6%8C%89%E9%92%AE%20%EF%BC%887%EF%BC%89%E6%B5%8B%E8%AF%95%E5%AE%89%E8%A3%85%E6%98%AF%E5%90%A6%E6%88%90%E5%8A%9F%EF%BC%8C%E6%8C%89%E4%B8%8B%E3%80%90win%2BR%E3%80%91%E9%94%AE%EF%BC%8C%E8%BE%93%E5%85%A5cmd%EF%BC%8C%E6%89%93%E5%BC%80cmd%E7%AA%97%E5%8F%A3%20%E8%BE%93%E5%85%A5%EF%BC%9Anode%20-v%20%2F%2F%20%E6%98%BE%E7%A4%BAnode.js%E7%89%88%E6%9C%AC

* npm ci

出现问题:

<https://blog.csdn.net/JEREMY_GYJ/article/details/130117116>

npm证书失效：

使用最新淘宝镜像源

npm config set registry <https://registry.npmmirror.com>

启动：npm run dev

## 端口配置

git config --global http.proxy http://127.0.0.1:10809

git config --global https.proxy <http://127.0.0.1:10809>

查看端口方法

https://blog.csdn.net/qq\_55888300/article/details/131498092