

Conda安装和卸载

<code>sh Anaconda3-2021.05-Linux-x86_64.sh</code>	#安装包Anaconda3-2021.05-Linux-x86_64.sh在移动硬盘里面
<code>rm -rf anaconda3</code>	#卸载时直接把anaconda3那个文件夹删掉即可

Conda自身

<code>conda --version</code>	#查看本机conda的版本号
<code>conda info</code>	#查看包括conda版本的更多信息
<code>conda update conda</code>	#更新conda到最新版本

帮助信息

<code>conda -h</code>	#查看conda的所有可用指令
<code>conda create -h</code>	#具体查看conda create指令的相关帮助

环境管理

<code>conda create -n test python=3.6</code>	#创建一个名为test，python版本为3.6的conda环境
<code>conda remove -n test --all</code>	#删除名为test的环境
<code>conda create -n test_rename --clone test</code>	#克隆test环境，并将其命名为test_rename
<code>conda create --name test --file file.txt</code>	#使用包信息文件建立环境（配套conda list --explicit > file.txt 用）
<code>conda env create -f environment.yml</code>	#使用配置文件建立（配套conda env export > environment.yml）
#用这种方式创建的话连名字都不用起，会和配置文件复刻的一样	
<code>conda activate test</code>	#在terminal中激活test环境
<code>conda deactivate</code>	#在terminal中关闭当前环境，回到base环境中
<code>conda env list (或conda info -e)</code>	#查看conda中所有已创建环境的名称及其存放位置

管理环境中的包

安装

#创建环境的同时安装所需的包（命令中指定版）	
<code>conda create -n test python=3.6 pytorch=1.0.1 numpy</code>	
#创建一个名为test，python版本为3.6的conda环境，并且在创建时就安装1.0.1版本的pytorch和numpy	
#创建环境的同时安装所需的包（用包信息文件指定版）	

```
conda list --explicit          # 查看包信息
conda list --explicit > file.txt  # 导出包信息到当前目录的file.txt文件中
conda create --name test --file file.txt  # 使用包信息文件建立环境
# 往往用来跨设备构建相同的conda环境

# 创建环境的同时安装所需的包（使用环境配置文件版）
conda env export > environment.yml      # 激活所要拷贝的环境，并记录其配置文件
conda env create -f environment.yml      # 在其他设备上利用该配置文件创建一个完全相同的环境
# 往往用来跨设备构建相同的conda环境（注意这种方式创建的环境不用起名字，名字在配置文件里有）

# 激活环境后，使用conda install安装指定的库
conda install paddlepaddle-gpu==2.2.0 cudatoolkit=10.2 --channel
https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/Paddle/
# 不推荐使用pip install，conda install会把所需库的依赖都自动装齐补全，更方便

# 激活环境后，使用包信息文件安装所需的包
conda install --file file.txt          # 使用包信息文件向一个已经存在的环境中安装指定包
```

删除

```
conda remove pytorch          # 在当前环境中删除包
conda remove -n test pytorch  # 在指定环境中删除包
```

查找包（当你记不清包的名字和版本时）

```
conda search py              # 模糊查找，即模糊匹配，只要含py字符串的包名就能匹配到，如pytorch
# 由于他默认的channel只有那4个，所以像paddle那种需要自己额外指定channel的包是搜不到的
```