

windows11 配置 miniforge3

目录

- windows11 配置 miniforge3
 - miniforge 下载及安装
 - 下载
 - 安装
 - .condarc 文件配置
 - miniforge指令
 - 🚀 常用指令
 - 📦 包管理常用指令
 - 🛠️ 环境配置相关指令
 - 📁 环境导出与迁移
 - 🌈 AI 项目常用 Conda 指令

miniforge 下载及安装

下载

1. 清华大学开源软件镜像站: [清华镜像 - Miniforge](#)
2. GitHub: [conda-forge/miniforge](#)

- 下载 Miniforge3-Windows-x86_64.exe 后安装

安装

几个特殊选项:

- Install for: 选 Just Me (其实选 All Users 也没啥区别)
- 选 create shortcuts, add installation to my PATH environment variable 和 Register Miniforge3 as my default Python 3.12 这前三个选项

.condarc 文件配置

位置: C盘-> user (用户) -> 用户名 -> .condarc 若没有则在此位置新建

```
# 告诉 conda 你的包主要从哪里下载。
# 排序很重要: 上面越靠前的优先级越高。
channels:
  - conda-forge

# 当从 conda-forge 下载包时, 设置去清华镜像找
# 还有别的镜像 https://mirrors.ustc.edu.cn/anaconda/cloud (中科大)
custom_channels:
  conda-forge: https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud

# channel_alias: https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud
# mirrored_channels:
#   conda-forge:
#     - https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
#     - https://conda.anaconda.org/conda-forge
#     - https://prefix.dev/conda-forge

# 环境保存的位置, 自己定
envs_dirs:
  - D:\Tools\miniforge3\envs

# 安装包时, 显示它来自哪个源
show_channel_urls: true
```

```
# 打开终端时，不自动进入 base 环境（推荐）
auto_activate_base: false
# 严格优先级，来自高优先级 channel 的包优先使用，避免冲突
channel_priority: strict
```

保存后运行指令 `mamba clean --all -y` 再创建环境。

注意：若用miniforge专用的mirrored_channels，则需要写全URL。他不像custom_channels和channel_alias，并没有URL补全功能，如指定conda-forge的URL则需要最后加上/conda-forge，鄙人在此处踩过坑，不加的话会出现404的错误。正确使用mirrored_channels是这样的：

```
mirrored_channels:
  conda-forge:
    - https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/conda-forge
    - https://conda.anaconda.org/conda-forge
    - https://prefix.dev/conda-forge
```

同时使用 channel_alias 和 mirrored_channels 时，mirrored_channels 会覆盖 channel_alias，mirrored_channels 优先级高于 channel_alias，所以如果使用mirrored_channels，一定要写对URL！！

miniforge指令

常用指令

① 创建环境（推荐 mamba，更快）

```
mamba create -n myai python=3.10
```

创建一个名为 myai 的环境。

② 激活环境

```
mamba activate myai
# 或 conda activate myai
```

③ 退出环境

```
mamba deactivate
# 或 conda deactivate
```

④ 查看所有虚拟环境

```
conda env list
```

⑤ 删除环境

```
mamba env remove -n myai
```

⑥ 查看信息

```
mamba --version    # 输出版本号
mamba info         # 查看环境信息
# 或 conda --version
# conda info
```

包管理常用指令

安装包

```
mamba install numpy
```

安装 AI 相关的，比如：

```
mamba install pytorch torchvision torchaudio -c conda-forge
```

删除包

```
mamba remove numpy
```

更新包

```
mamba update numpy
```

更新所有包：

```
mamba update --all
```

搜索包

```
mamba search pytorch
```

查看已安装的包

```
conda list
```

环境配置相关指令

查看当前配置

```
conda config --show
```

清理缓存和临时文件

```
mamba clean --all -y
```

(清理 mamba / conda 缓存和临时文件的指令，非常常用，尤其是在遇到下载错误或空间不足时)

添加镜像源（例：清华）

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

环境导出与迁移

导出环境（复制给别人/备份）

```
conda env export > environment.yml
```

根据文件创建环境

```
mamba env create -f environment.yml
```

AI 项目常用 Conda 指令

安装 PyTorch（CPU）

```
mamba install pytorch torchvision torchaudio -c conda-forge
```

安装 PyTorch（CUDA 12.1）

```
mamba install pytorch torchvision torchaudio cudatoolkit=12.1 -c conda-forge
```

安装 Transformers

```
pip install transformers
```

(Transformers 在 pip 上更新最快，推荐 pip 安装。)

安装 Jupyter Notebook

```
mamba install jupyterlab
```

安装 OpenAI SDK

```
pip install openai
```