2024/4/22

蓝区PC前端环境搭建教程（新人向）：

基于

<https://github.com/AustinXT/tensorboard/blob/dev/sync/compile.md>

做的一些因时间原因而失效的步骤的改动和补充

**请搭一个梯子**

请先为自己的电脑搭个梯子，这样可以免去很多上github和其他网站的麻烦

并且本人工作场景没有Huawei-Guest不清楚是否可以直接微软商店下载，但是本人的手机流量并不能直接从微软商店下载，挂了梯子之后才不会下载报错.

若可以直接下载或者流畅访问外网请忽略本条目

**环境搭建**

环境搭建采用在Windows系统上安装WSL2的方式作为操作系统，在此基础上搭建VPN去实现网络访问自由，然后按照 Dockerfile 文件记录的步骤搭建环境。

为什么不直接在Windows系统上，Windows系统上直接安装遇到的坑太多了（linux上也不少就是了），比如 yarn package-patch 无法应用成功、（yarn抽风，某个包下载不下来）、VS Studio 下载、路径格式差异等。

**启用WSL功能**

首先，你需要在Windows上启用WSL功能。

打开“控制面板”。

转到“程序” > “程序和功能”。

选择左侧的“启用或关闭Windows功能”。

在弹出的窗口中找到“适用于Linux的Windows子系统”或“Windows Subsystem for Linux”，勾选它。

点击“确定”并重启计算机。

本人在安装时也勾上了Hyper-v，不知是否有作用

**安装 WSL2**

要求Windows 10版本需要是2004的更新（Build 19041）

本人在刚入职时用的系统皆为Win11，WSL2请自行百度下载，

若文件名为: WSL\_update\_x64 大概就没错了，记得要先启动WSL，并重启计算机之后再进行安装，否则会提示安装失败.（因为你不重启WSL还没生效嘛）

**安装Linux发行版**

打开Microsoft Store。

搜索你想要的Linux发行版（如Ubuntu, Debian, Fedora等）。

选择一个发行版，点击“获取”进行安装。

按照屏幕上的指示完成设置。



笔者安装的版本为Ubuntu 22.04.3 LTS （因本人只用过Ubuntu，对其他软件熟练可自行尝试，接下来的步骤皆以此版本为基础）

**升级到WSL 2（如果需要）**

如果你安装的是WSL 1，需要升级到WSL 2。

打开“PowerShell”为管理员，运行以下命令来设置WSL 2为默认版本：

wsl --set-default-version 2

wsl -l -v

wsl --set-version <distribution name> 2

（此步骤可能有用，不太清楚，建议试一下）

### 搭建 VPN（请务必找到一个VPN）

由于 WSL2 是独立的虚拟网络，需要单独配置VPN。

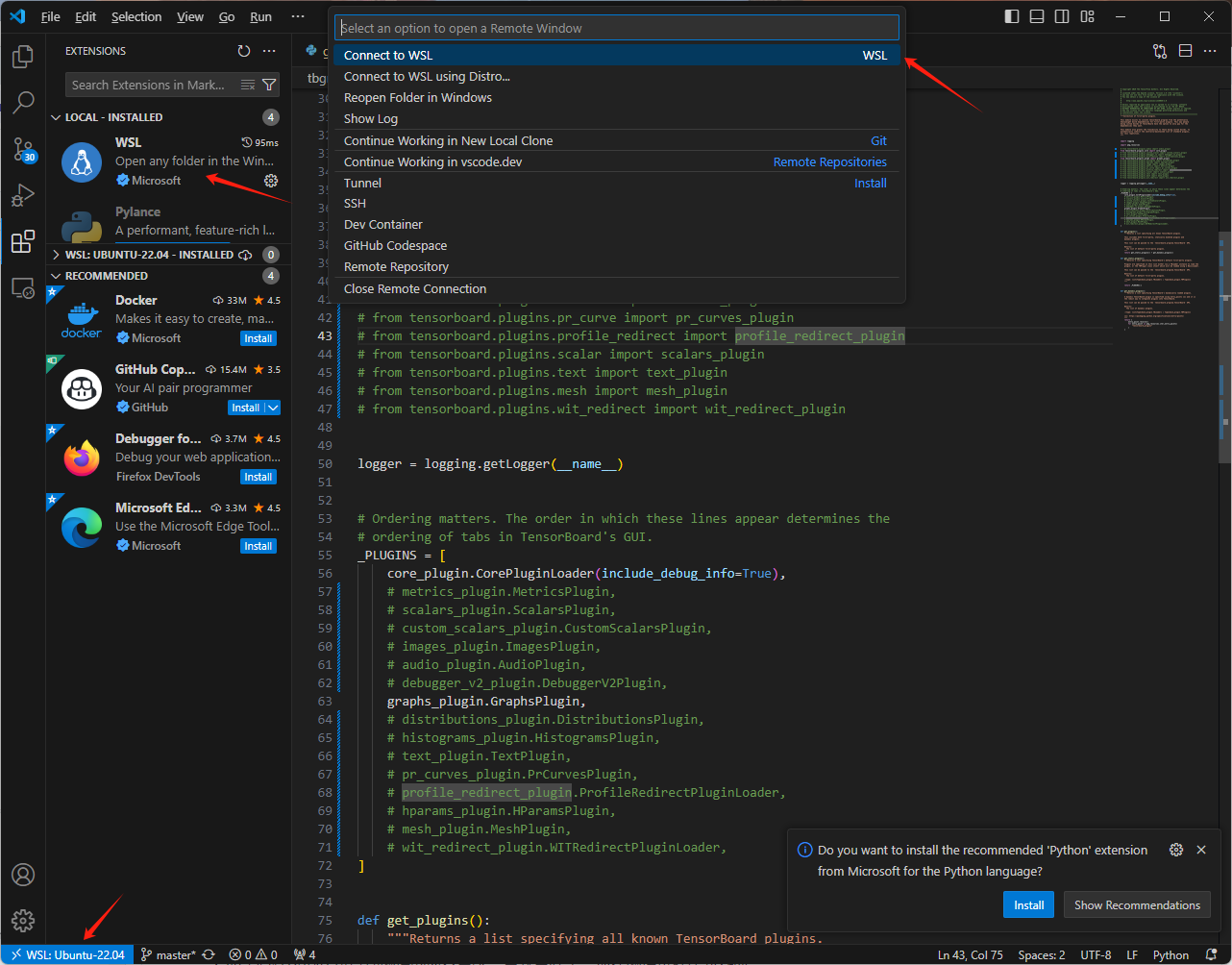
如果你是用 clash 的话，可以参考 [在 Linux 中使用 Clash | CodeSwift](https://blog.iswiftai.com/posts/clash-linux/)，安装包在这 [Clash](https://www.clash.la/releases/)（已过期，暂不推荐按照此链接使用）

（相信我，没有vpn下50多m的东西会下的想死，6kb/s真不是来开玩笑的，而且不仅下载慢，还会有找不到文件源的情况，一定要搭梯子）

梯子还是要自己搞定的，如果从来没用过vpn建议咨询你的pl  
本人使用的教程为<https://github.com/Elegycloud/clash-for-linux-backup>

（如果此教程过期请上github自行搜索关键词 “clash for linux” 寻找其他可用教程）

如果用的vscode：



先点击左下角<>这样的一个符号，然后点击conncet to wsl

Btw，别忘记下载vs-code的WSL插件，他会自动提示你的

前面的都很简单，不会出啥问题.

### 安装环境依赖

按照 Dockerfile 文件记录的步骤安装环境依赖。

sudo apt-get update

sudo apt-get install -y wget unzip python3 python3-pip python3-dev python-is-python3 default-jdk nodejs npm

python3 -m pip install -U pip

# 如果没有 yarn

npm install -g yarn

yarn config set registry https://registry.npm.taobao.org/

### 安装Bazel

您好，欢迎来到最讨厌的一集，安装Bazel，简而言之就是

你跟着下面走会给你安装3.7.1，然后他告诉你需求 4.0.0<= version <= 6.9.99

然后你安装Bazelisk给你自动更新版本会给你安装7.1.1，虽然bazelisk可以选择版本，但是笔者选择版本的时候一选就出问题（可能我有问题），这边有一个笨蛋方法，如果各位熟练Bazel安装没我这么笨蛋的话建议自行尝试

好了，“简”而言之完了，下面说下方法

1. 安装bazelisk

<https://github.com/bazelbuild/bazelisk>

npm install -g @bazel/bazelisk （应该这一个指令就够了）

1. 安装bazel

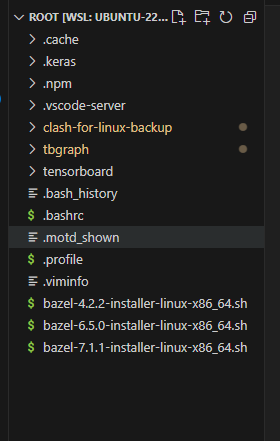
简单方法：<https://blog.csdn.net/liang_baikai/article/details/137234088>（记得把版本号改一下）

跟着上面教程搞不定之后的方法：

<https://github.com/bazelbuild/bazel/releases>，去这找你想下载的版本

<https://github.com/bazelbuild/bazel/releases/download/6.5.0/bazel-6.5.0-installer-linux-x86_64.sh>，这是我下的6.5.0版本, 如果没有特殊的版本要求可以下这个，

下载好拖到你的编译器里，例如我自己的：



给予权限

chmod +x bazel-6.5.0-installer-linux-x86\_64.sh

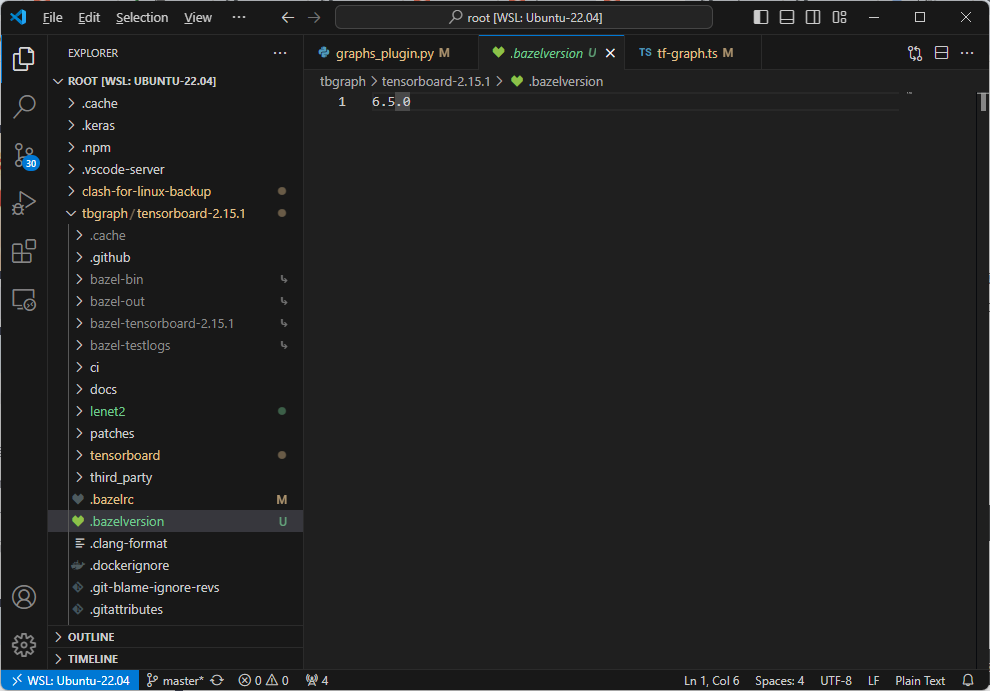
安装

./bazel-6.5.0-installer-linux-x86\_64.sh

安装好之后在/root/tbgraph/tensorboard-2.15.1里添加一份文件：

命名为.bazelversion，里面输入你所用的版本.

例如：



安装ibazel

npm i -g @bazel/ibazel

（以下部分为原旧版引导）

export BAZEL\_VERSION='4.2.2'

export BAZEL\_SHA256SUM='11dea6c7cfd866ed520af19a6bb1d952f3e9f4ee60ffe84e63c0825d95cb5859'

ci/download\_bazel.sh "${BAZEL\_VERSION}" "${BAZEL\_SHA256SUM}" ~/bazel

mv ~/bazel /usr/local/bin/bazel && chmod +x /usr/local/bin/bazel

npm i -g @bazel/ibazel

### 安装Python依赖（这步一定不能省略）

不要使用镜像源，可能会缺包

export BUILDTOOLS\_VERSION='3.0.0'

export BUILDIFIER\_SHA256SUM='e92a6793c7134c5431c58fbc34700664f101e5c9b1c1fcd93b97978e8b7f88db'

export BUILDOZER\_SHA256SUM='3d58a0b6972e4535718cdd6c12778170ea7382de7c75bc3728f5719437ffb84d'

export TENSORFLOW\_VERSION='tf-nightly'

pip install -r ./tensorboard/pip\_package/requirements.txt -r ./tensorboard/pip\_package/requirements\_dev.txt "$TENSORFLOW\_VERSION" && pip freeze --all

## 构建

### 拉取项目工程

[git@github.com:tensorflow/tensorboard.git](mailto:git@github.com:tensorflow/tensorboard.git)（已过时）

我用的是

吴毓龙的库，如库一样请与毓龙哥交流并申请加入仓库，如不同，请咨询负责人

git clone https://gitee.com/wuyulong11/tbgraph.git

### 拉取 bazel 构建依赖

cd tbgraph/tensorboard-2.15.1

bazel fetch //tensorboard/...

### 构建前准备

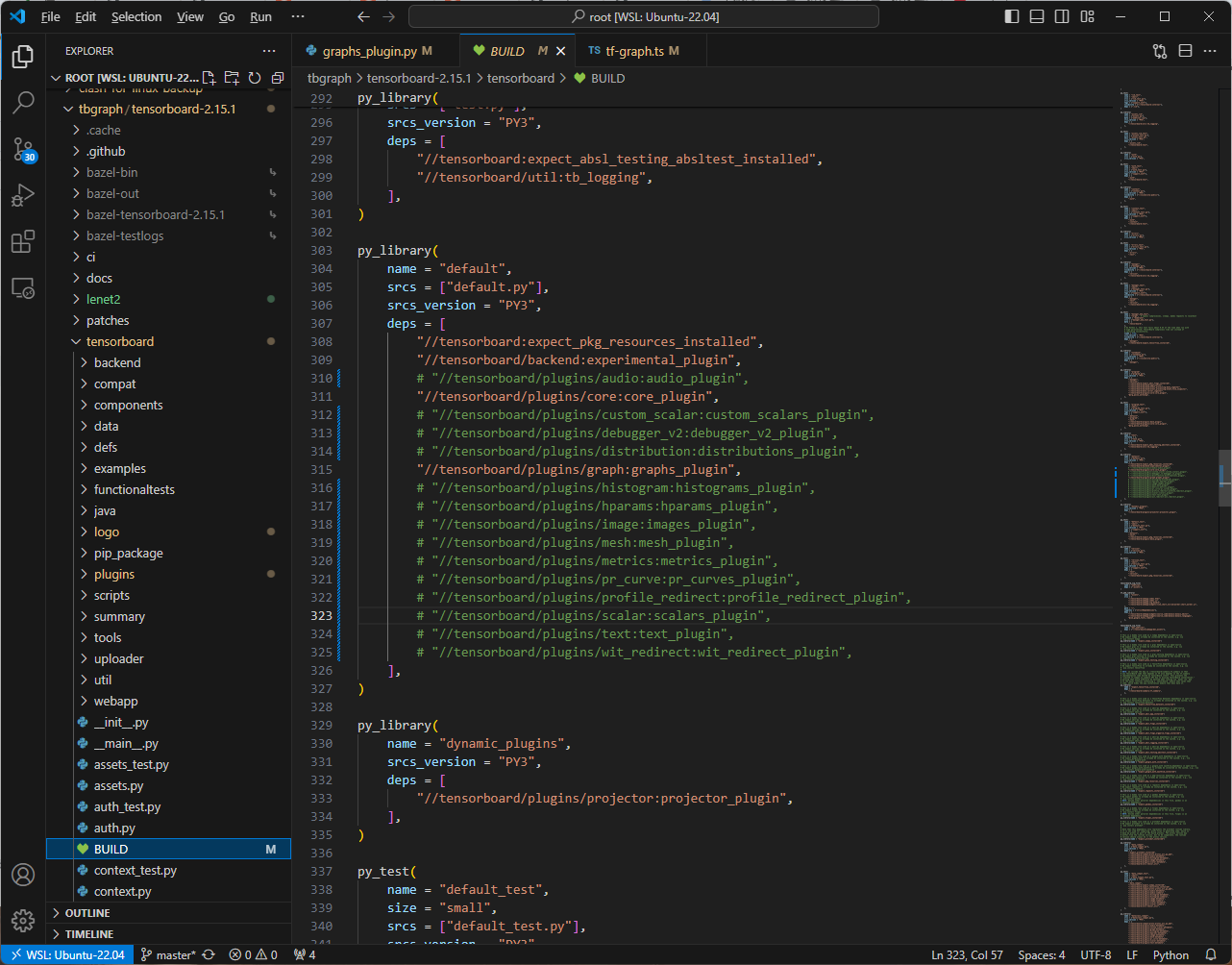
将shell脚本添加执行权限

find . -name \*.sh | xargs chmod u+x

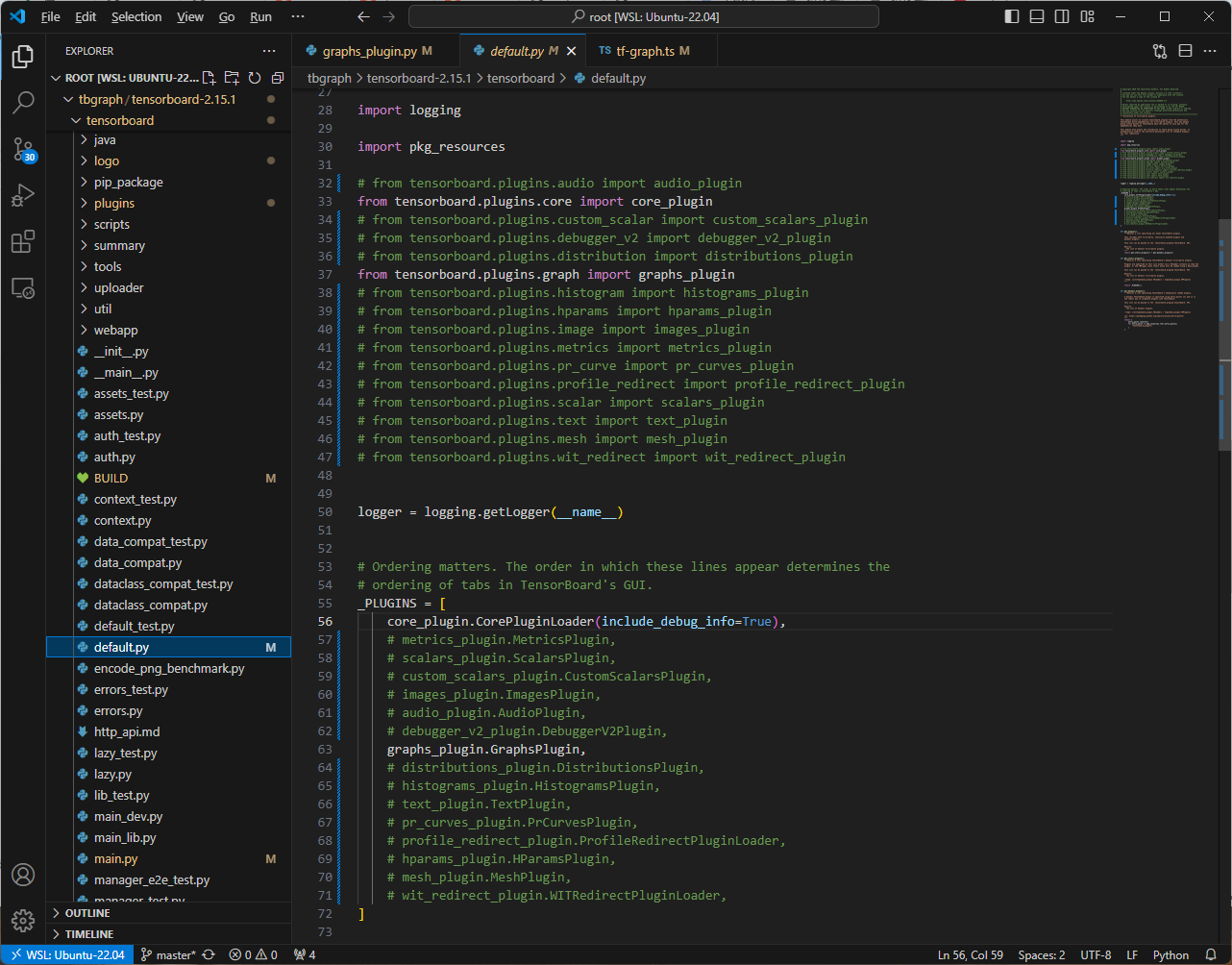
注释其他plugin:

此注释为2024/4/22 因只需要部分功能才进行的注释，是否注释此功能可能会根据时间流动而改变，请注意使用的功能并及时与自己的pl或导师沟通，此仅暂做展示：

/root/tbgraph/tensorboard-2.15.1/tensorboard/BUILD



/root/tbgraph/tensorboard-2.15.1/tensorboard/default.py



/root/tbgraph/tensorboard-2.15.1/tensorboard/main.py

/root/tbgraph/tensorboard-2.15.1/.bazelrc

添加缓存，将下面一行加到 .bazelrc

common --repository\_cache=".cache"

### 构建

bazel build tensorboard:tensorboard

ll ./bazel-bin/tensorboard/tensorboard

### 执行

要一份测试文件，我这有如下示例

建一个文件夹，把文件丢进去

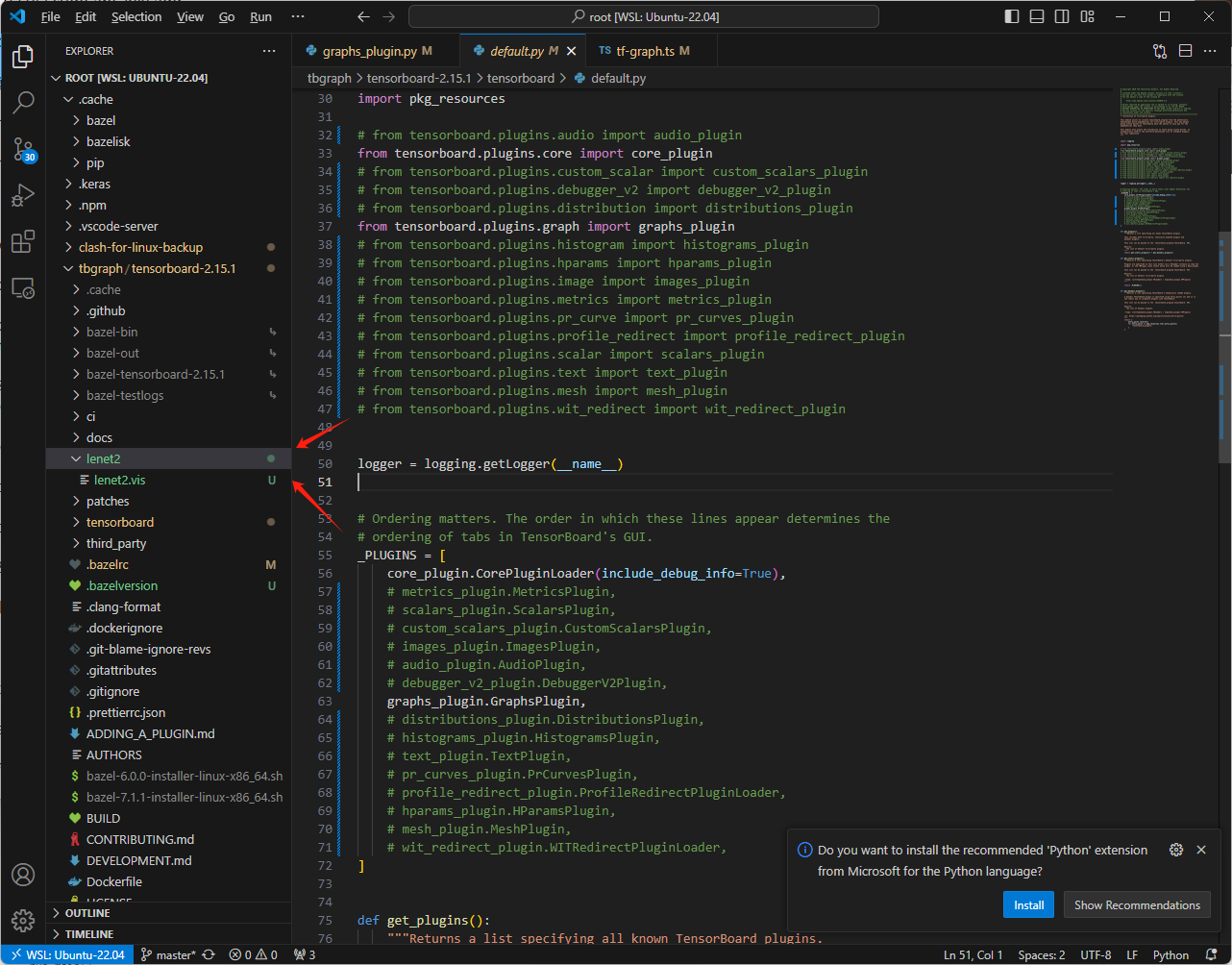
我的运行命令

ibazel run tensorboard -- --logdir /root/tbgraph/tensorboard-2.15.1/lenet2

格式：

Ibazel/bazel(热运行或正常运行) + run tensorboard -- --logdir + 你的文件夹路径

（指向的不是文件而是文件所在的文件夹）



bazel run tensorboard

# visit: <http://localhost:6006>

hot load

ibazel run tensorboard