VSCode&GDB—Demo工程使用说明

[文档副标题]

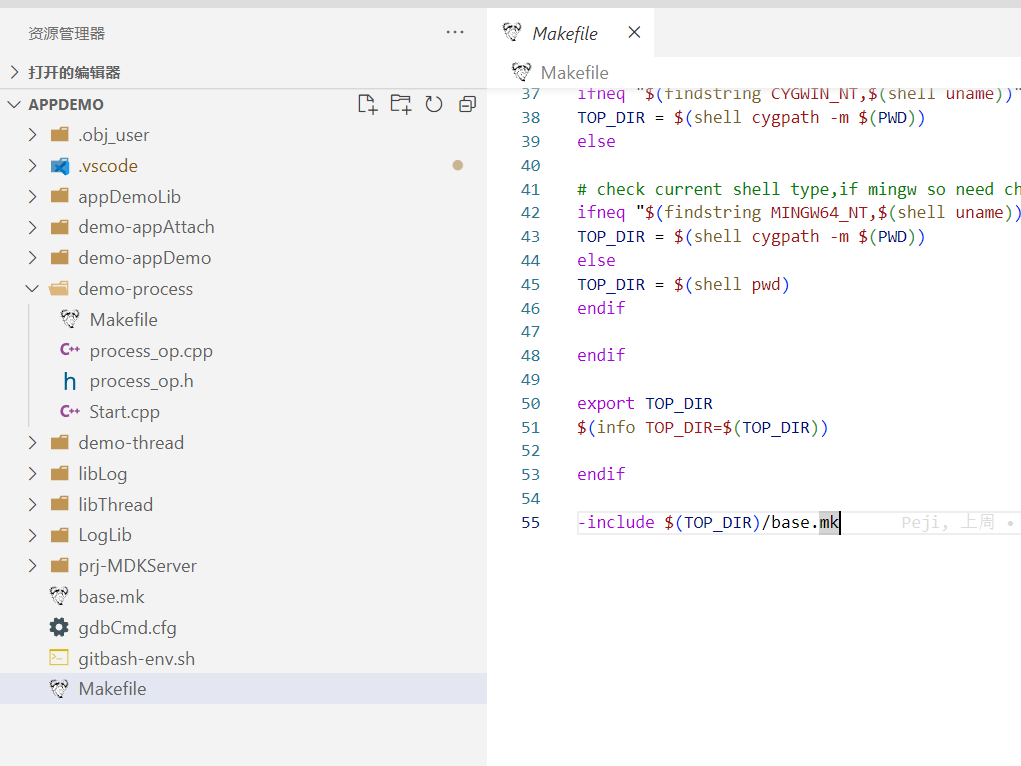
Peji

[年]

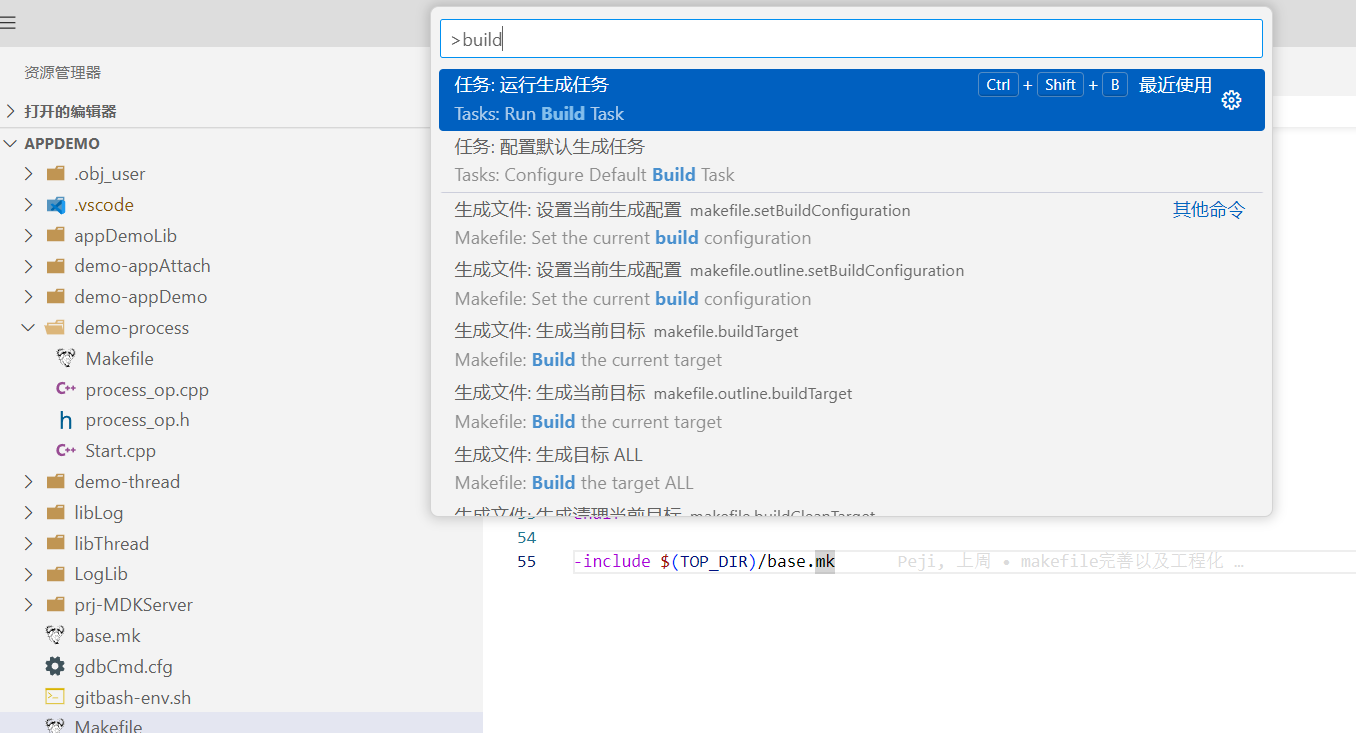
# 步骤

## 编译

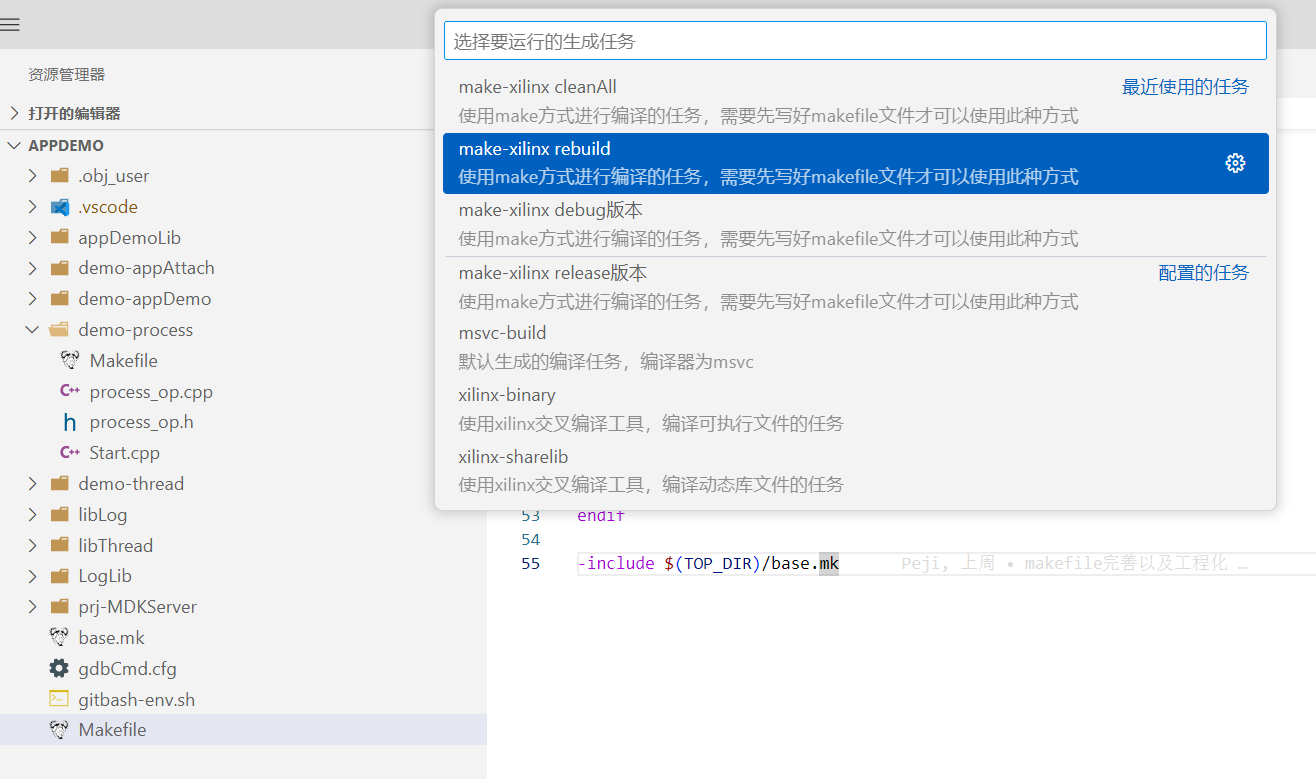
1. 使用VSCode打开Demo目录
2. 选中顶层makefile文件(一定要选中，并且每次进行编译时都需要选择此文件，否则出现问题)



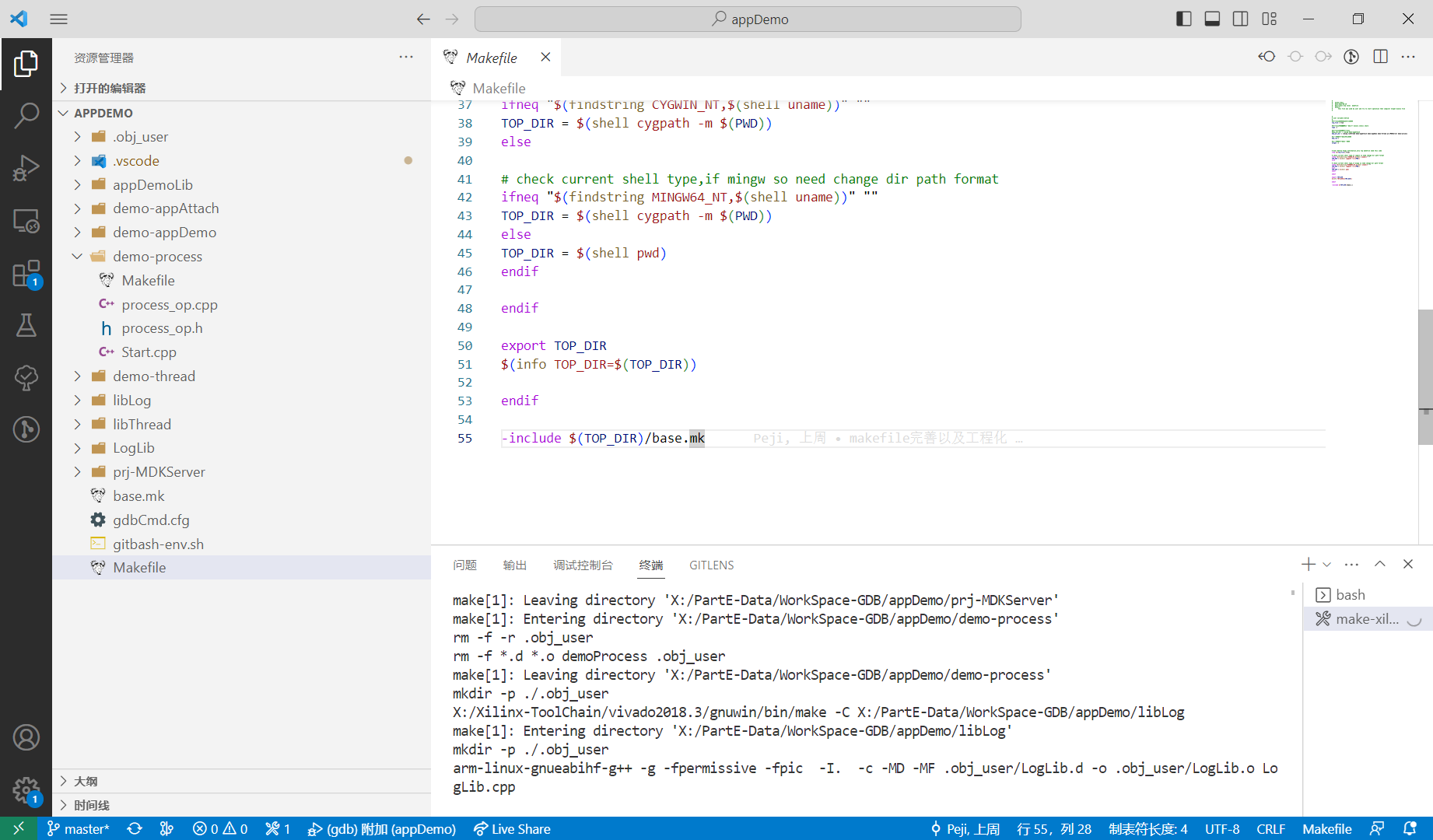
1. 按下F1,在弹出的窗口中输入build，然后选择build task

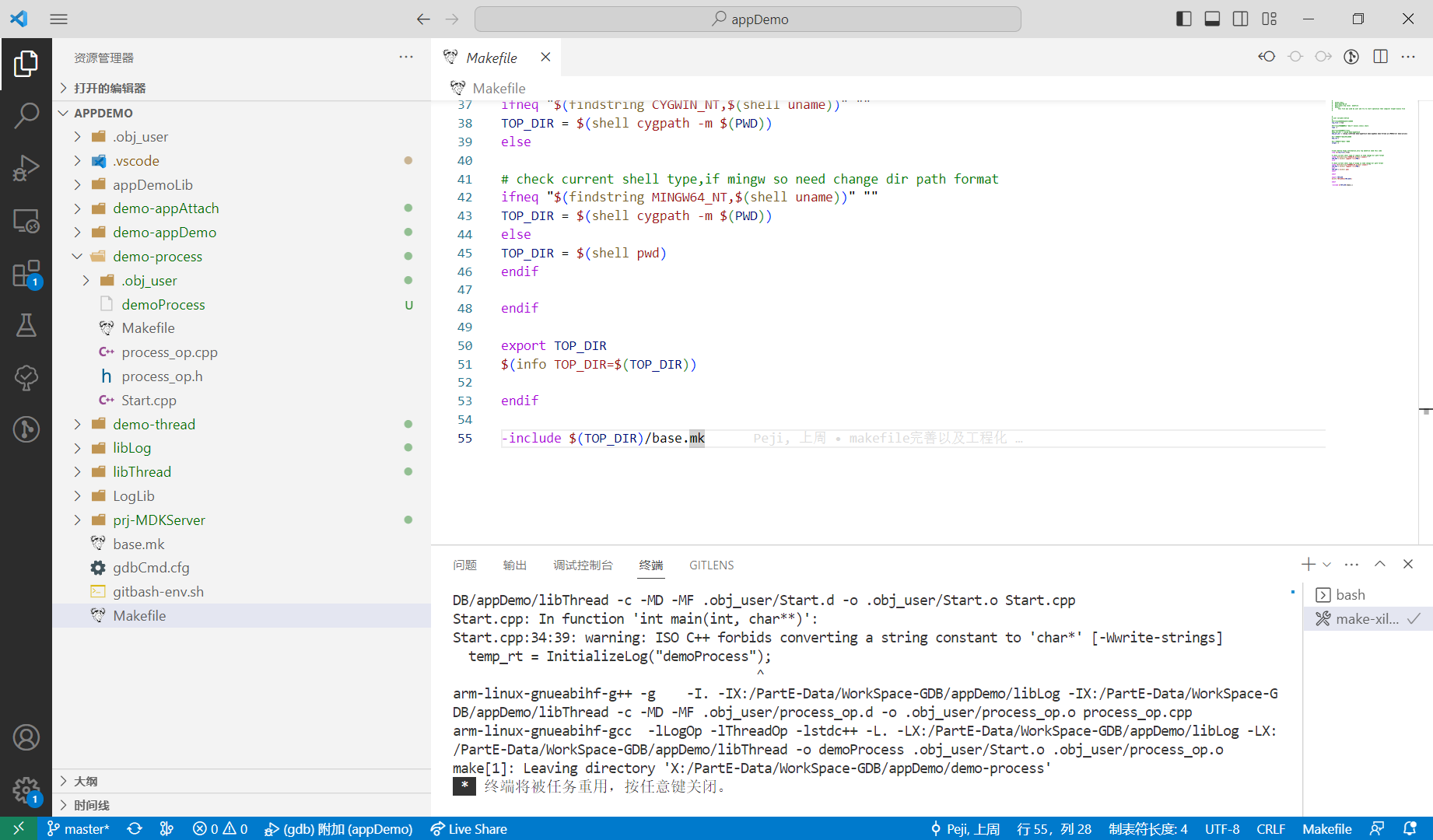


1. 选择需要执行的编译任务就会开始编译



1. 开始编译



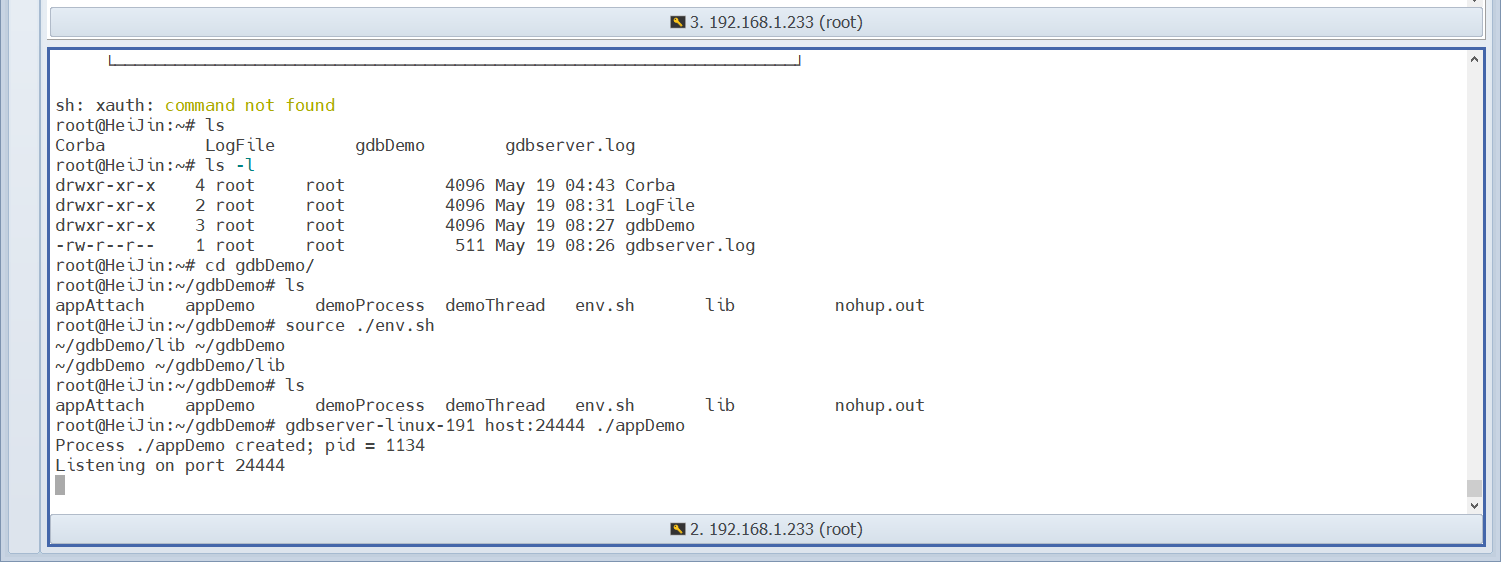


最终编译完成。如果存在问题，那么编译会提前终止，并显示错误信息。

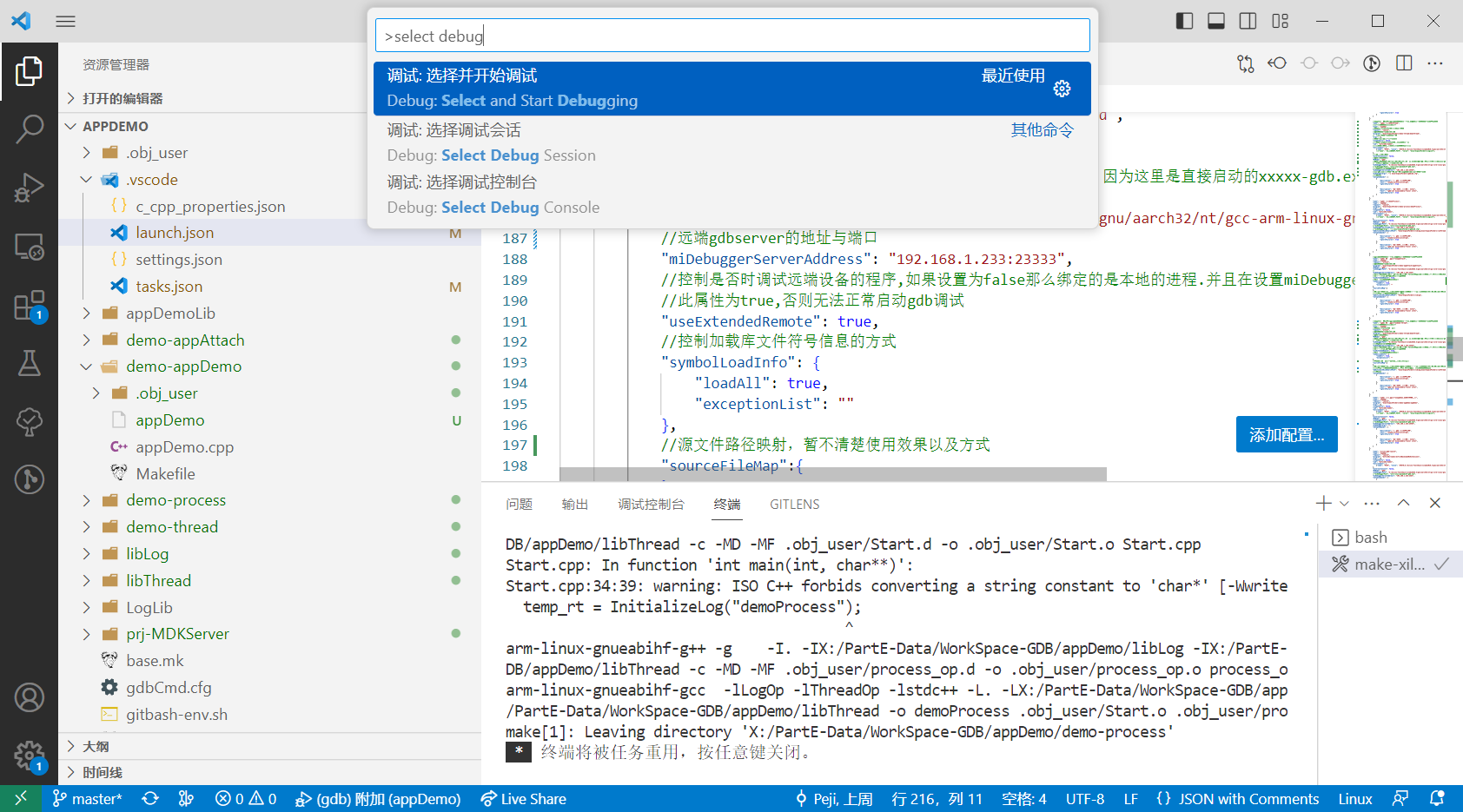
## 调试

### 直接启动

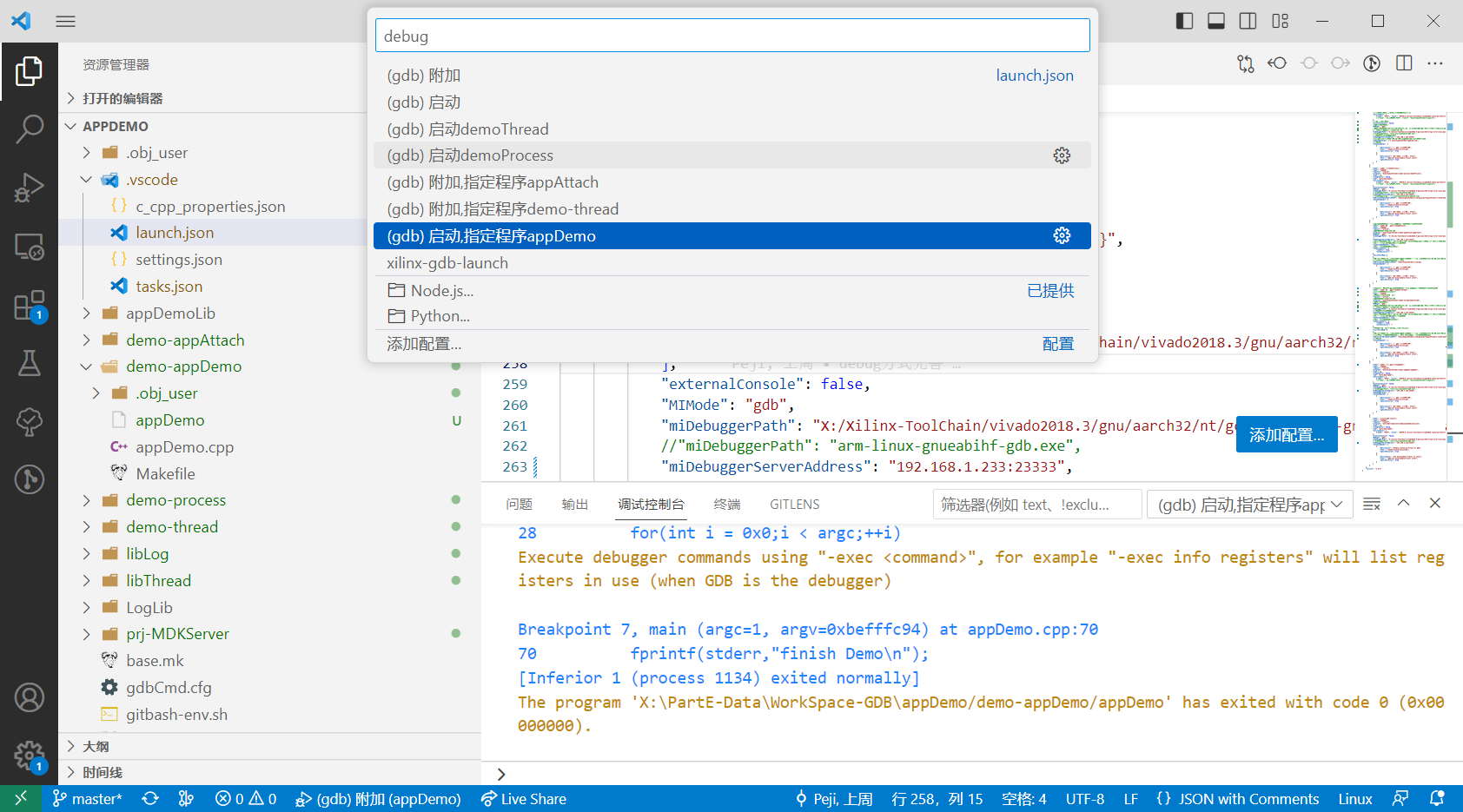
1. 将编译出来的文件上传到目标设备中
2. 修改目标设备中的文件权限，并确定文件可以运行
3. 执行命令gdbserver启动程序 gdbserver-linux-191 host:24444 ./appDemo



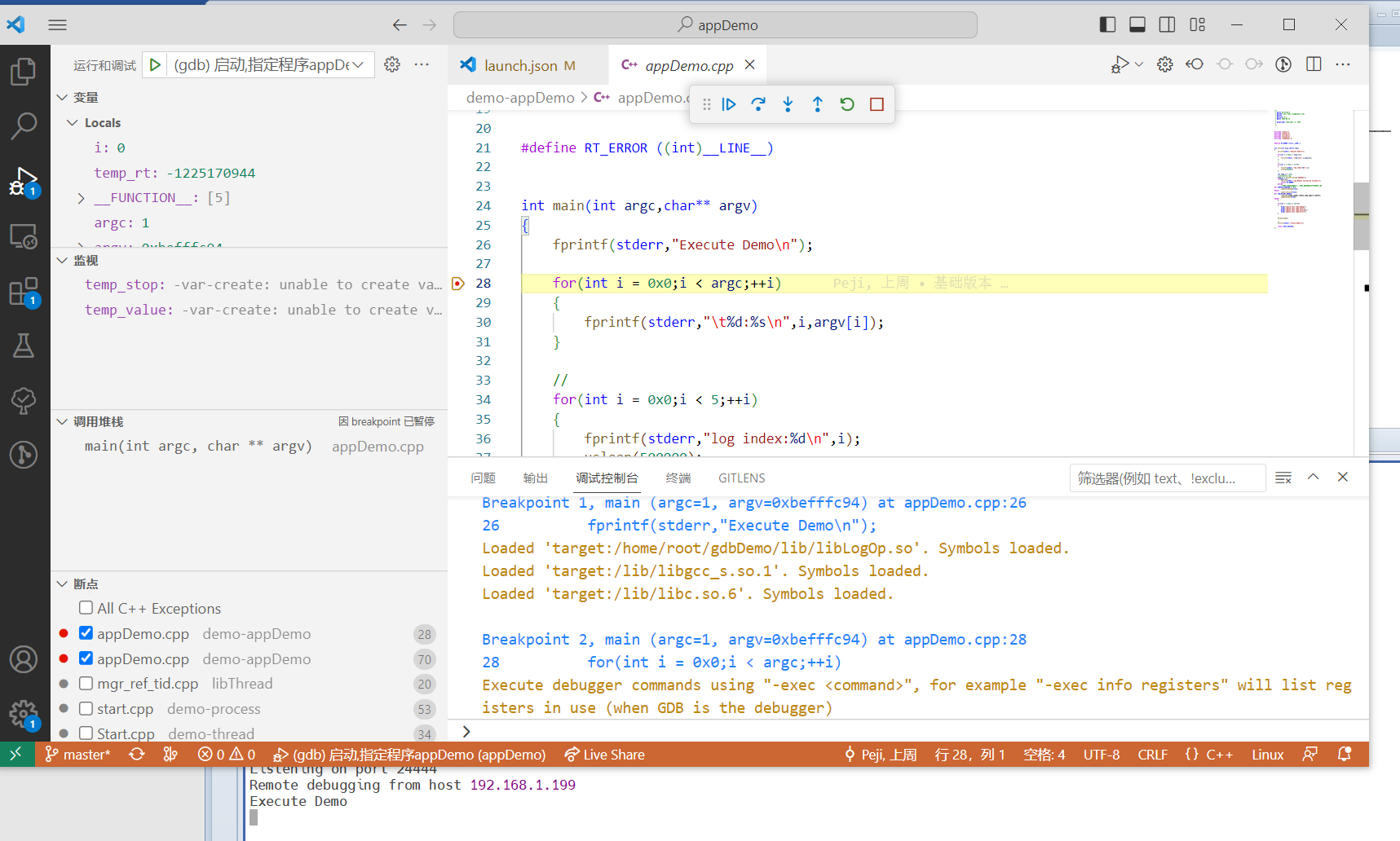
1. 在VSCode中按下F1,并在其中输入select debug,然后选择select debut选项

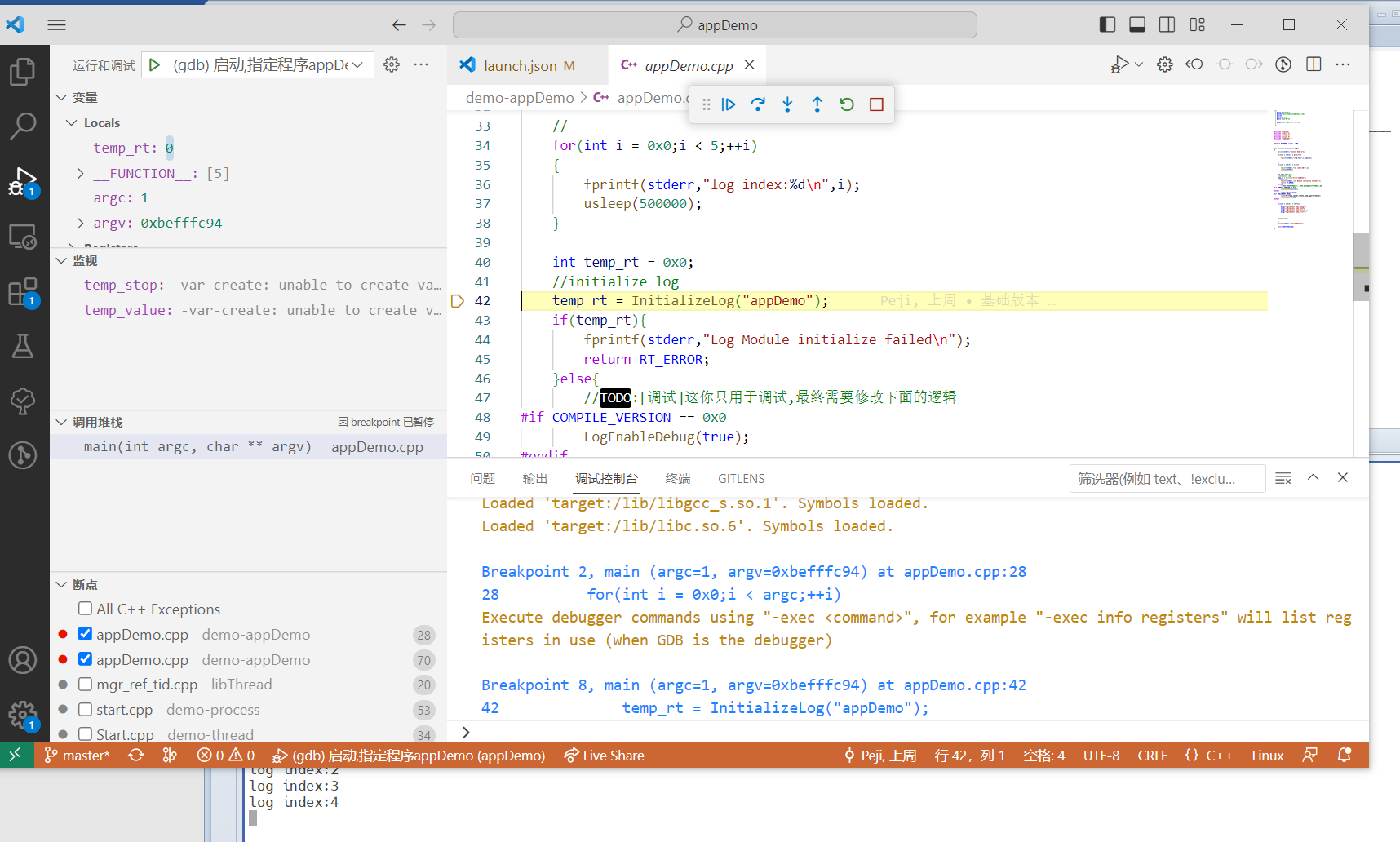


1. 选择需要加载的debug选项



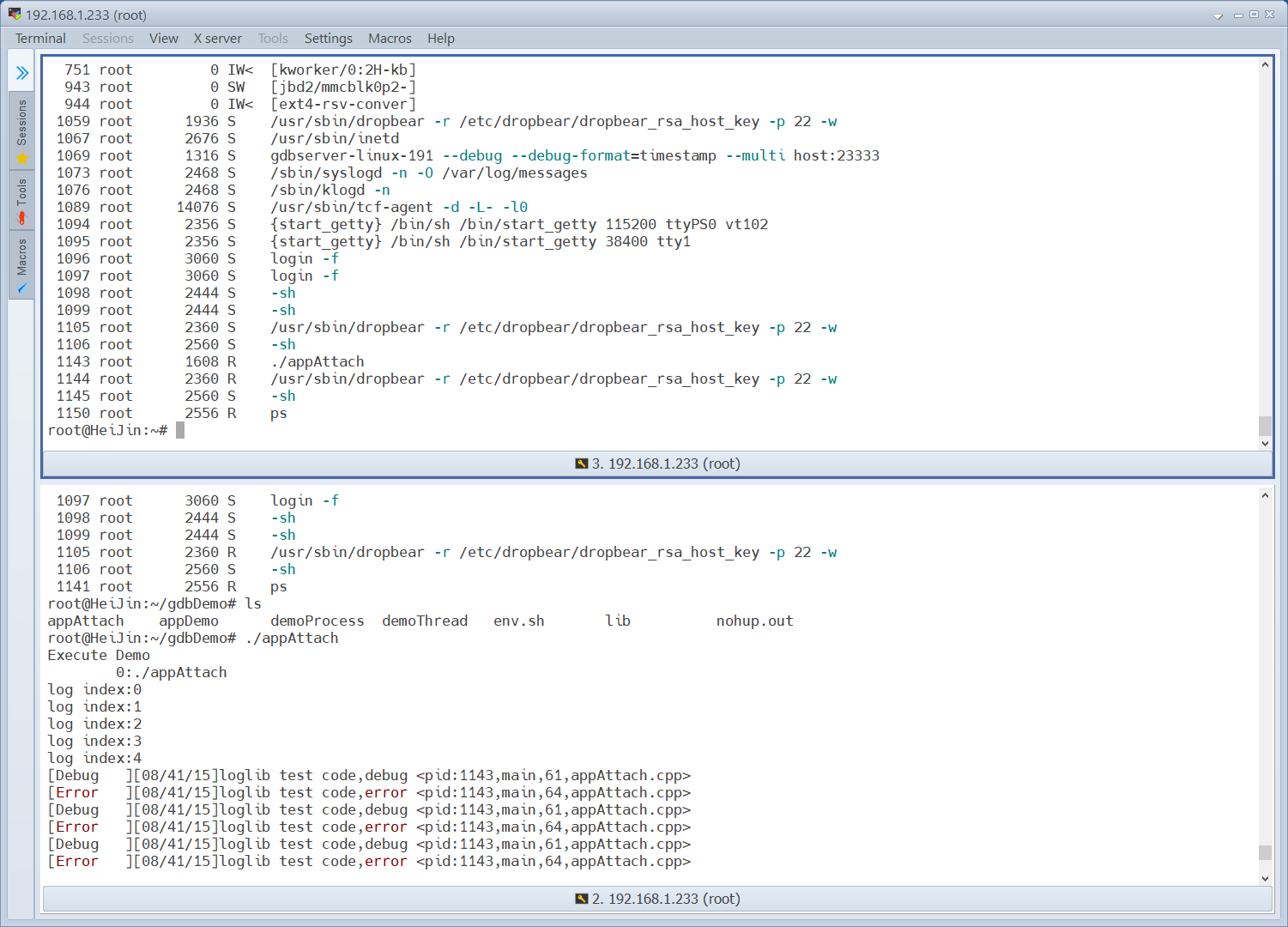
1. 开始调试



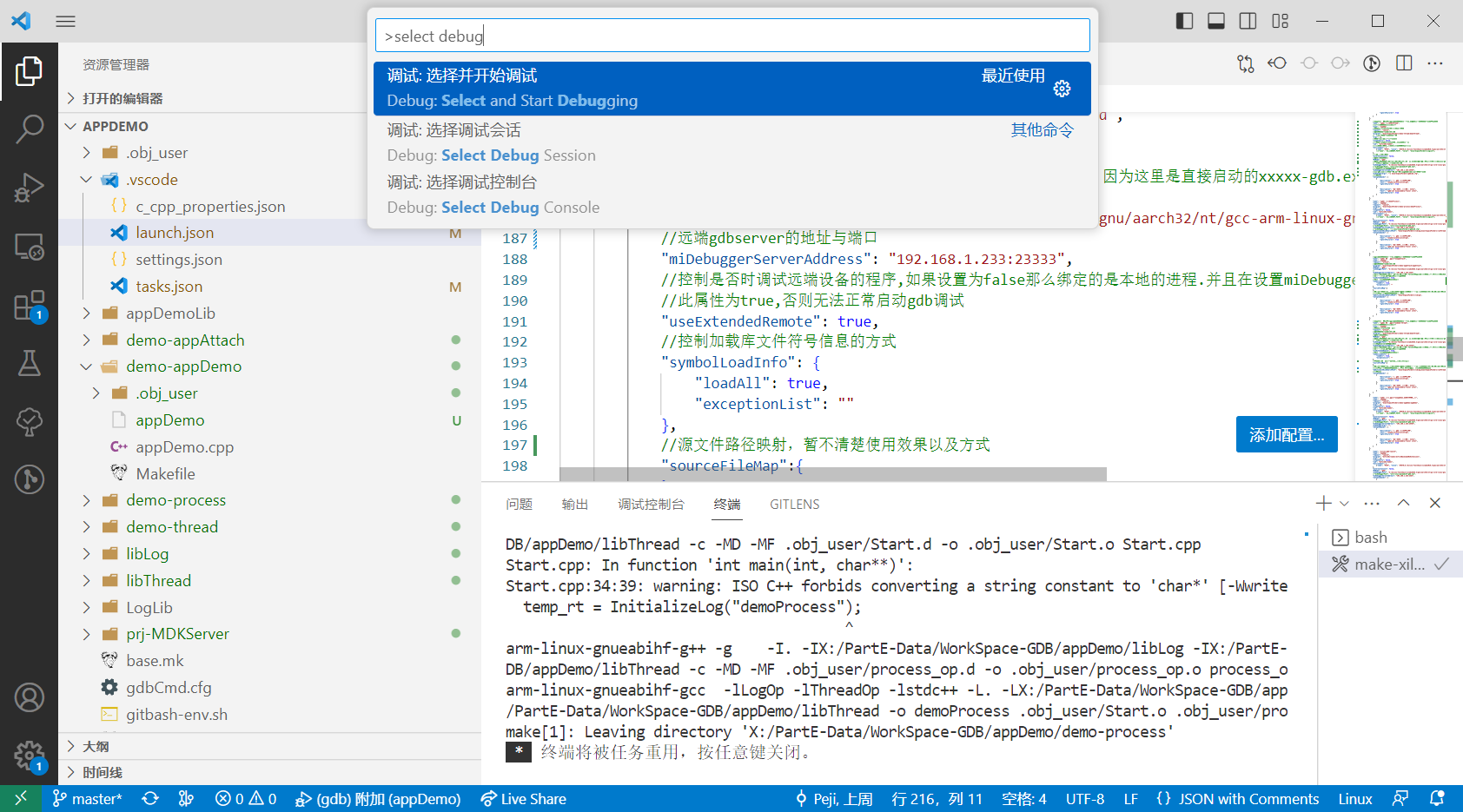


### 附加调试

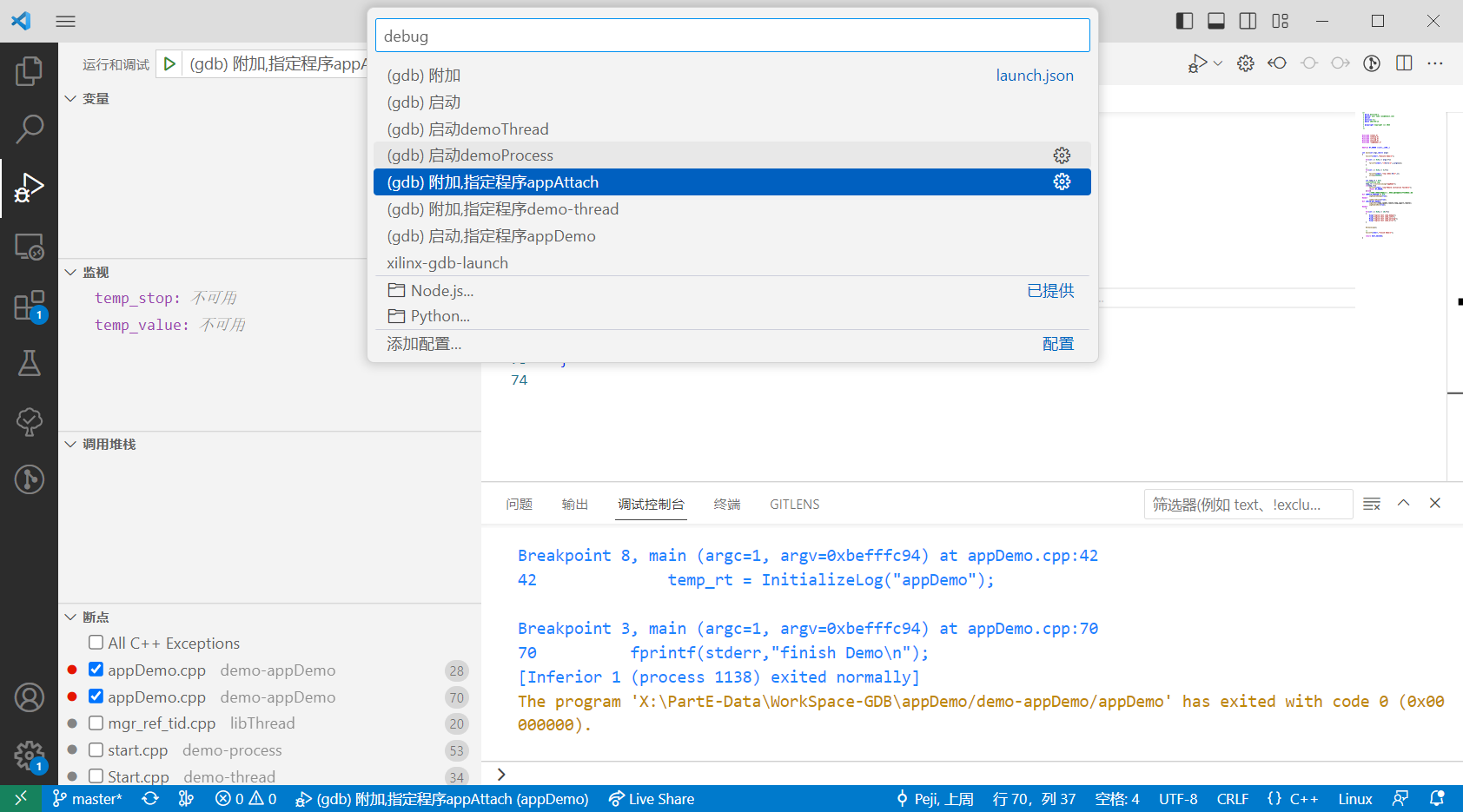
1. 在目标设备中启动gdbserver命令   
   gdbserver-linux-191 --debug --debug-format=timestamp --multi host:23333
2. 将编译出来的文件上传到目标设备中
3. 修改目标设备中的文件权限，并确定文件可以运行
4. 执行命令启动被调试程序



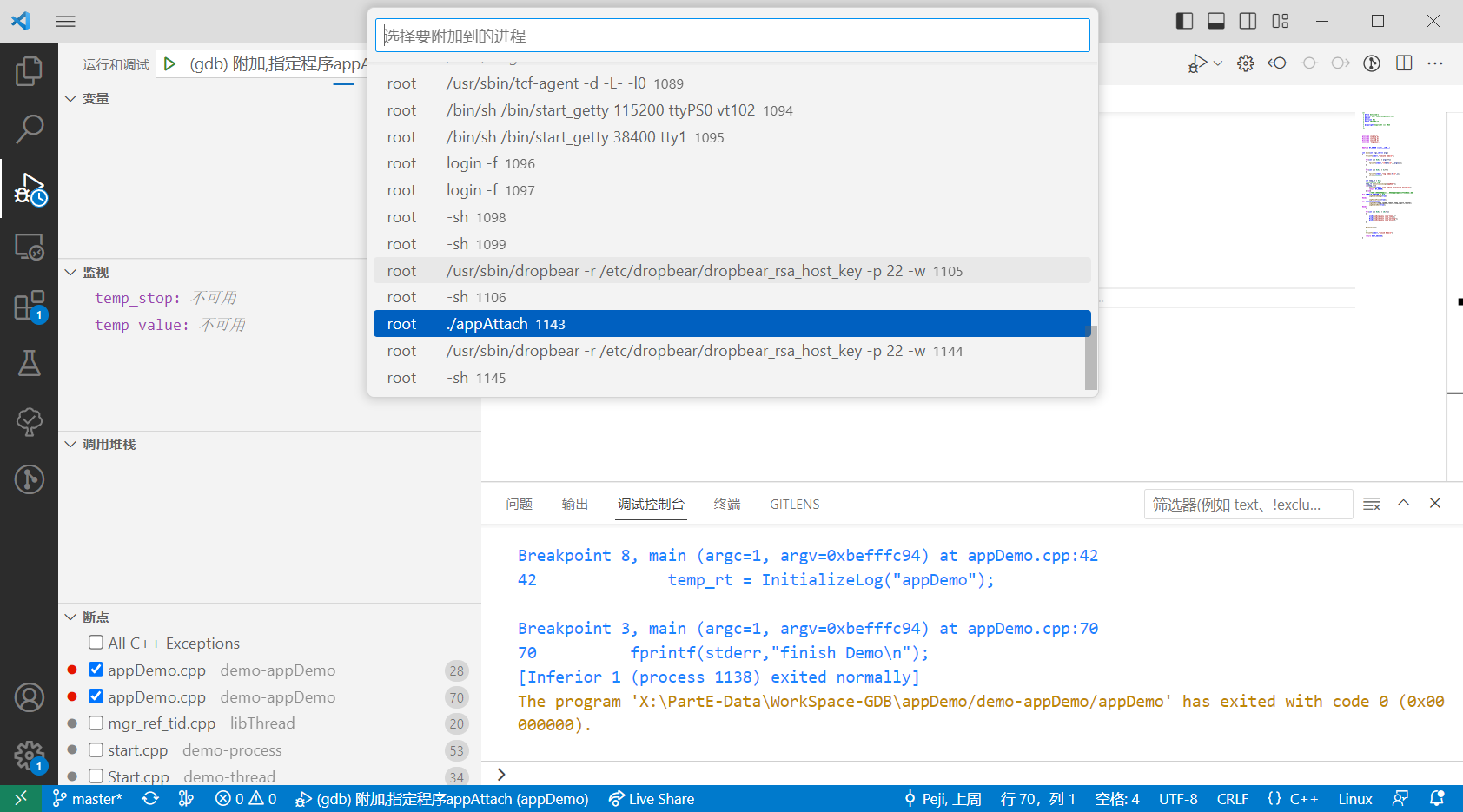
1. 在VSCode中按下F1,并在其中输入select debug,然后选择select debut选项



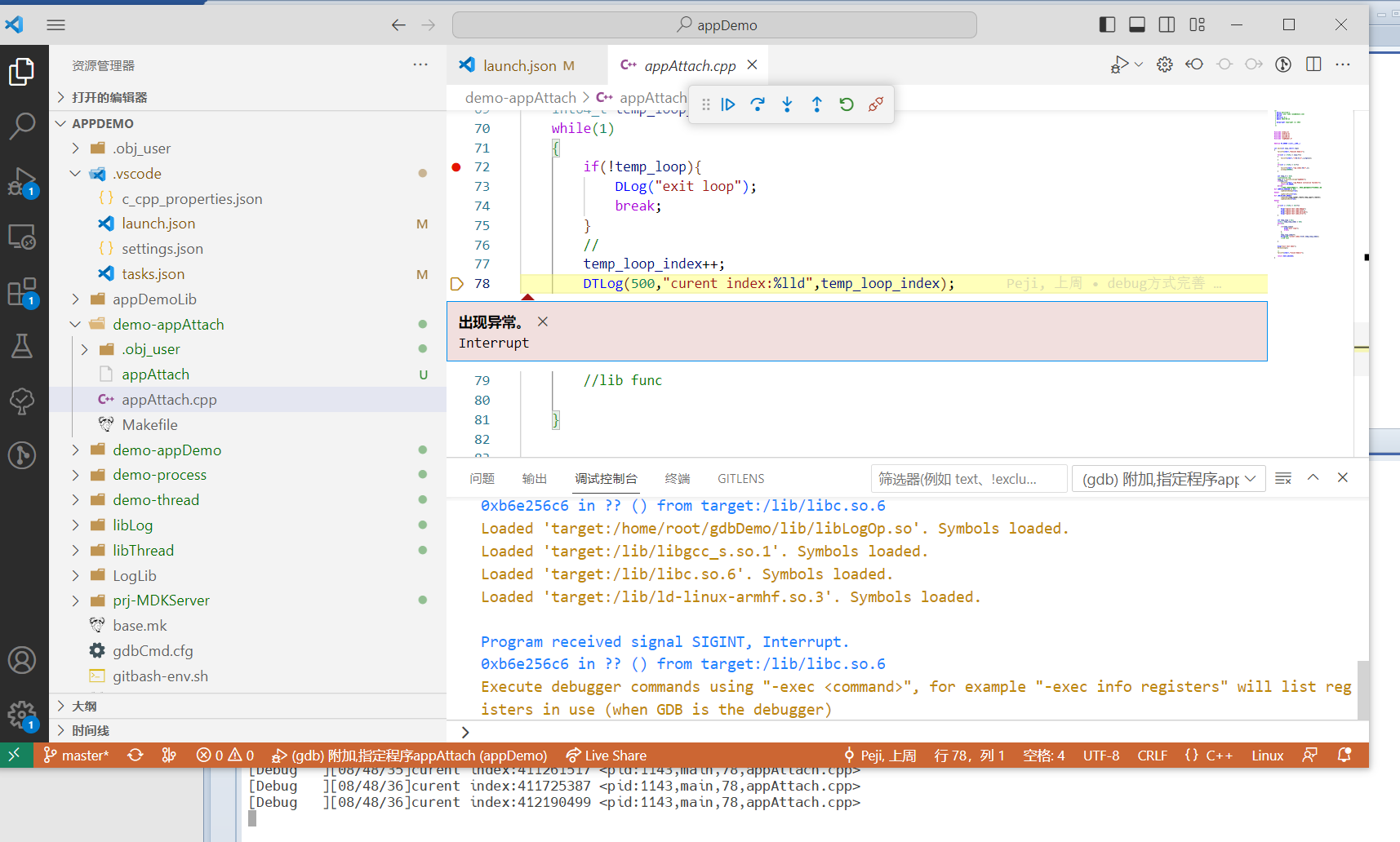
1. 选择需要执行的debug选项

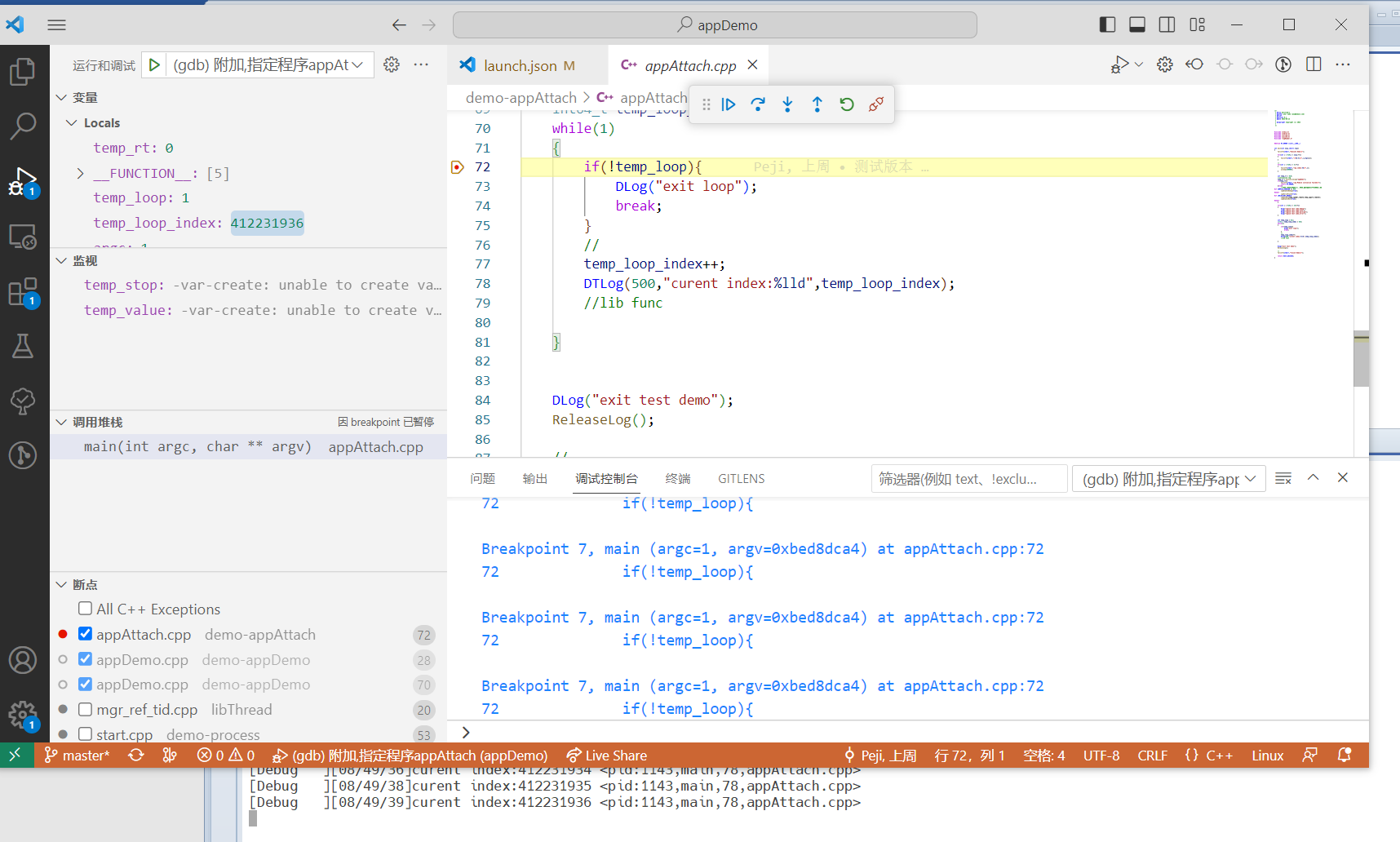


1. 选择需要附加的调试程序



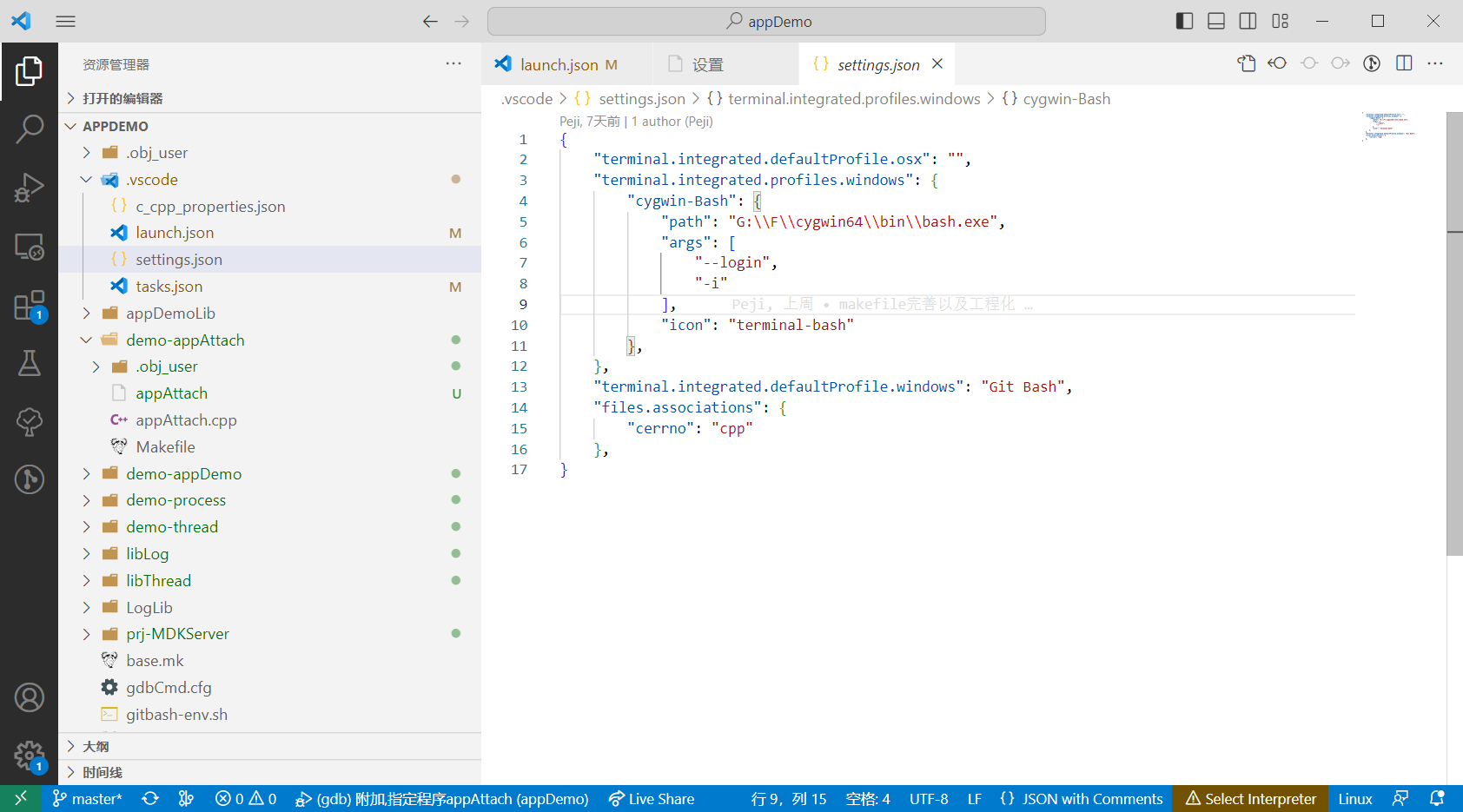
1. 开始调试





# 常见问题

1. VSCode需要按照标准的VSCode进行c/c++开发的流程准备好环境
2. VSCode的shell需要设置为git-bash



1. 对应的交叉编译工具链以及make，gdb命令的路径需要自行确定并进行设置

