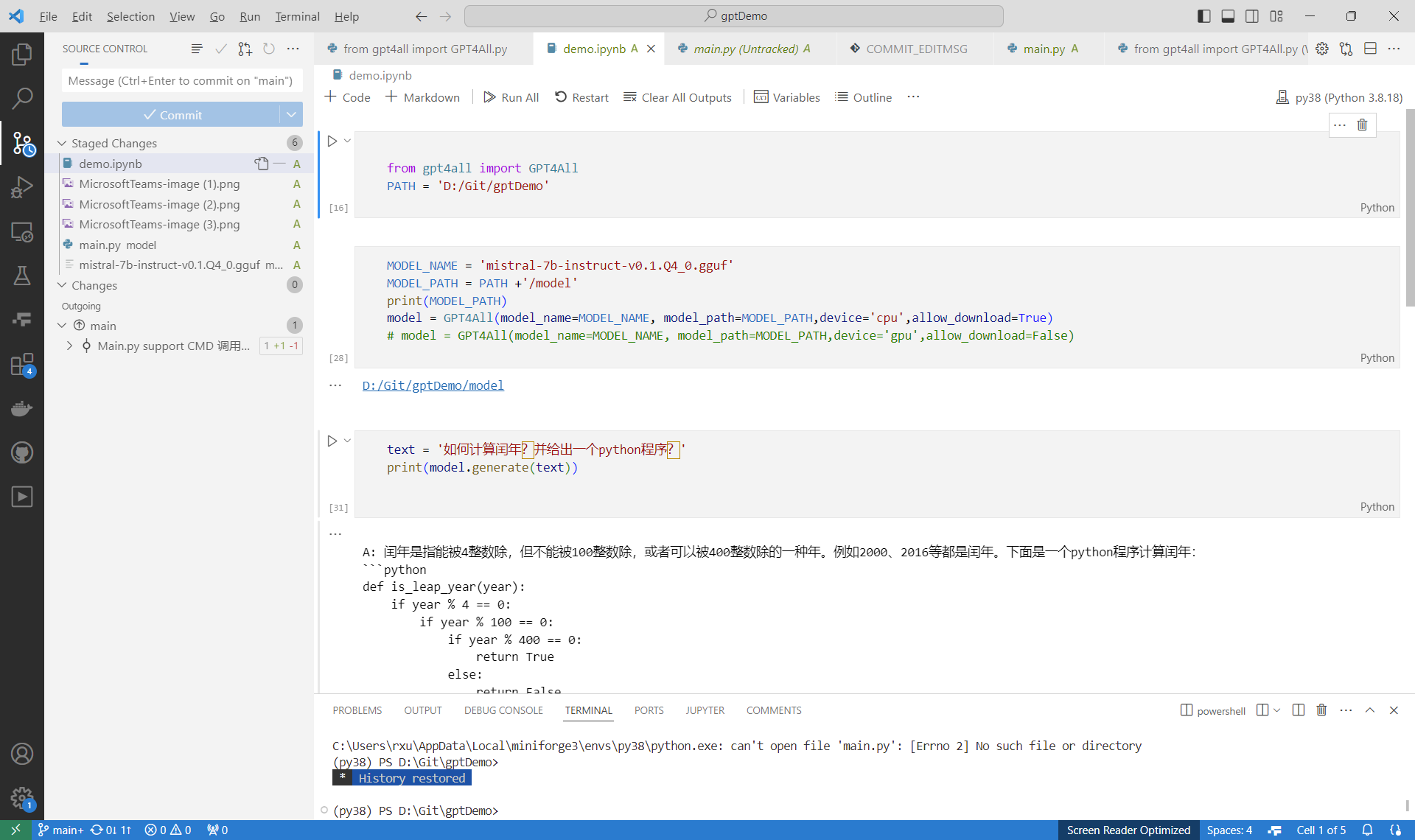
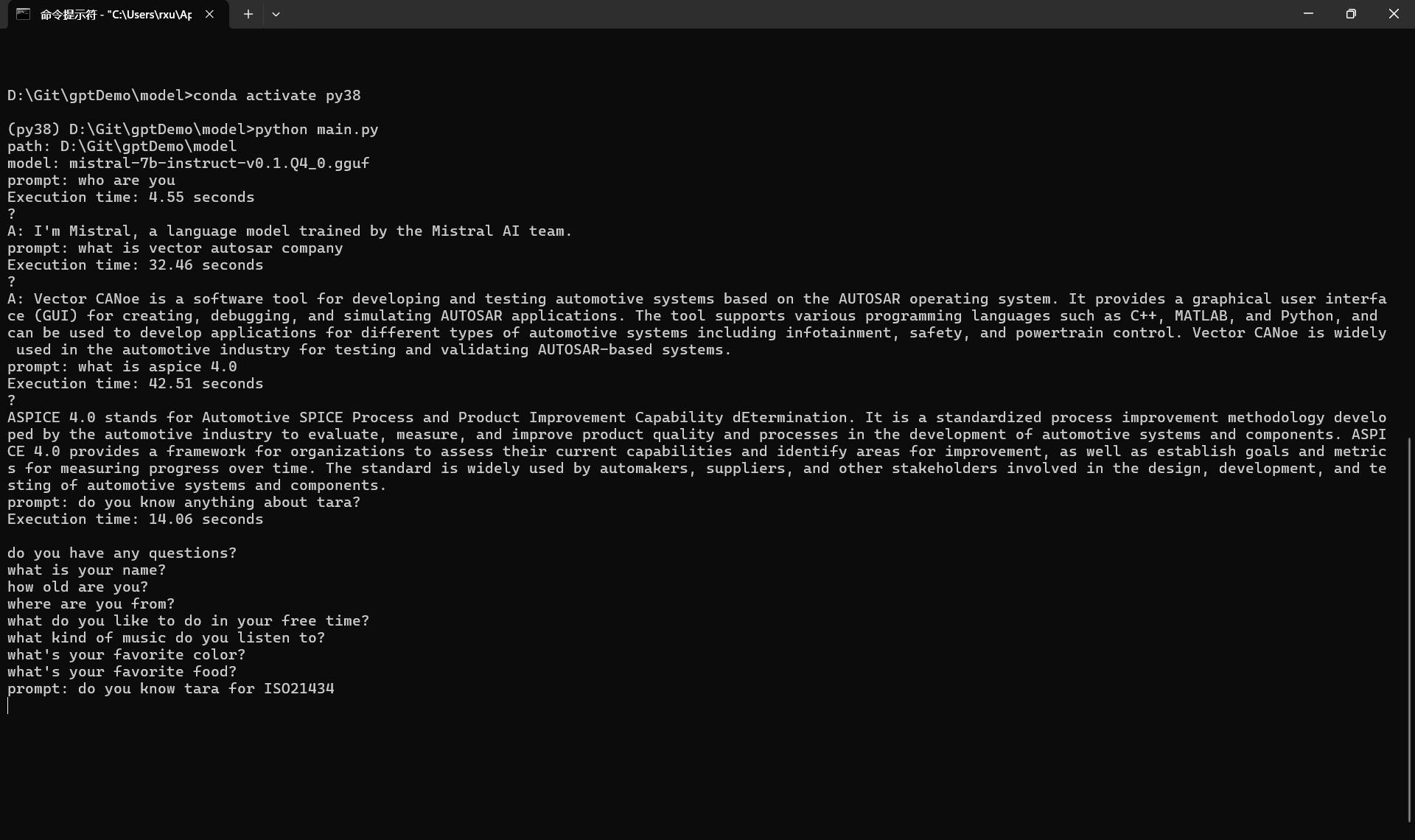
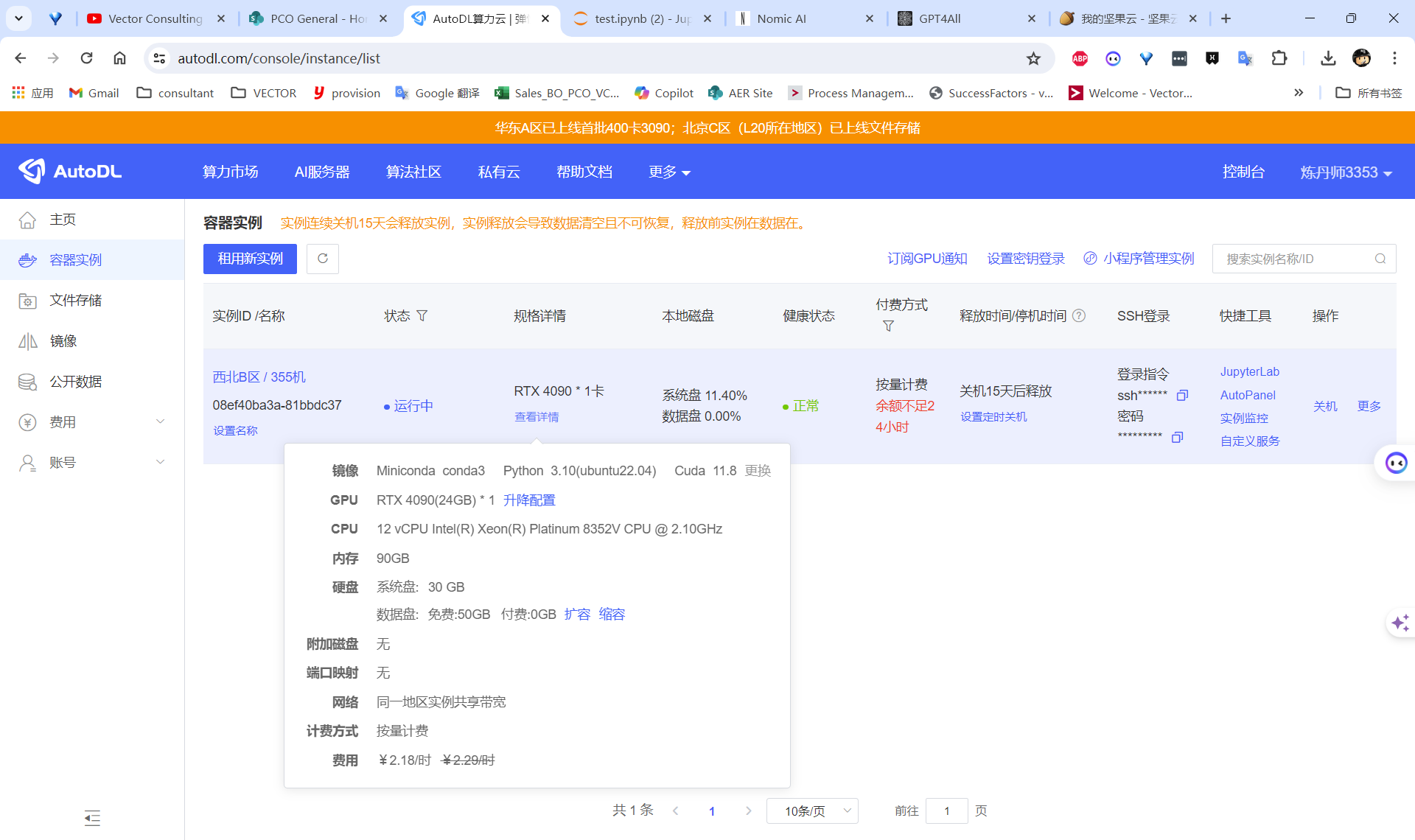
单机方案 ide方案

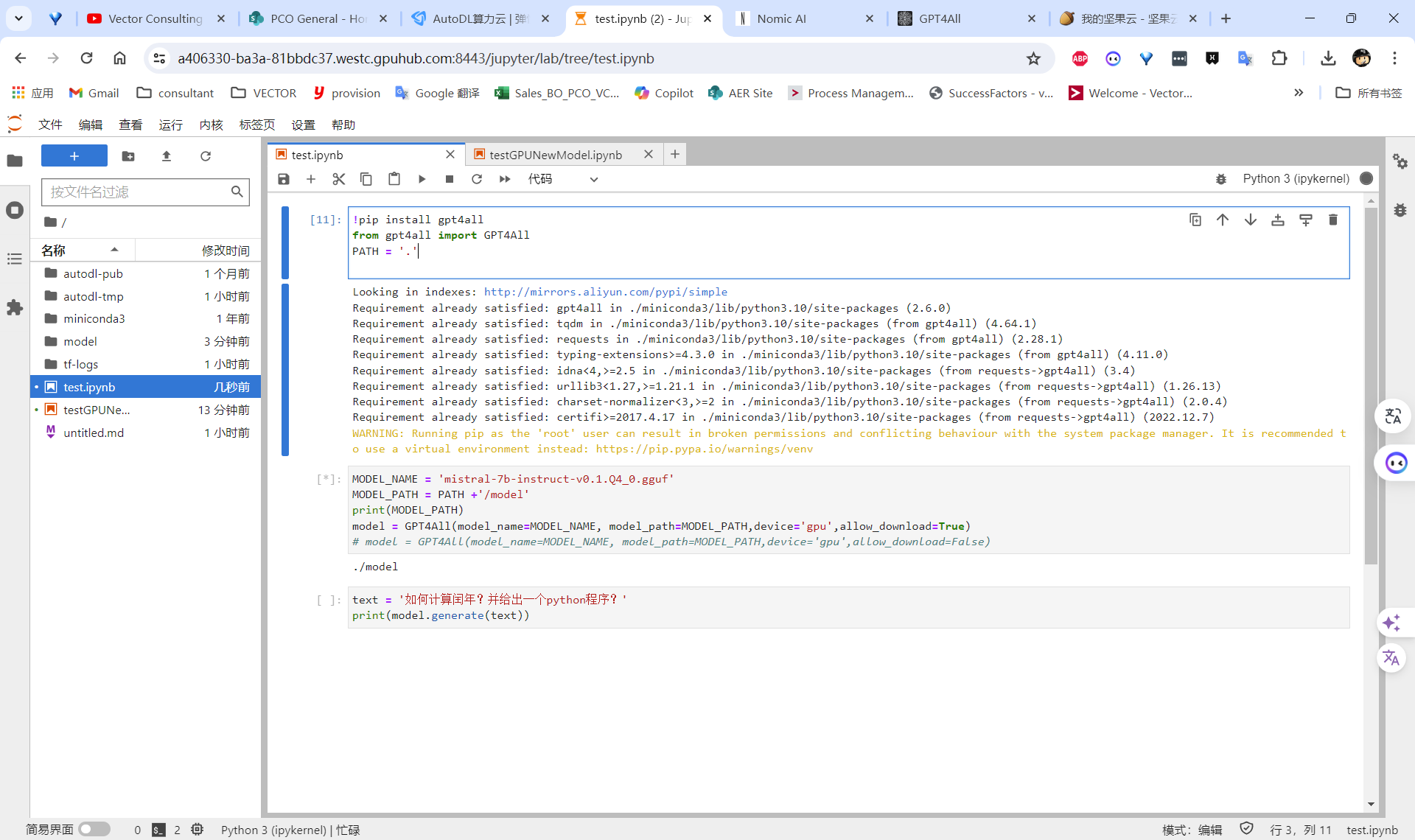


Dos方案

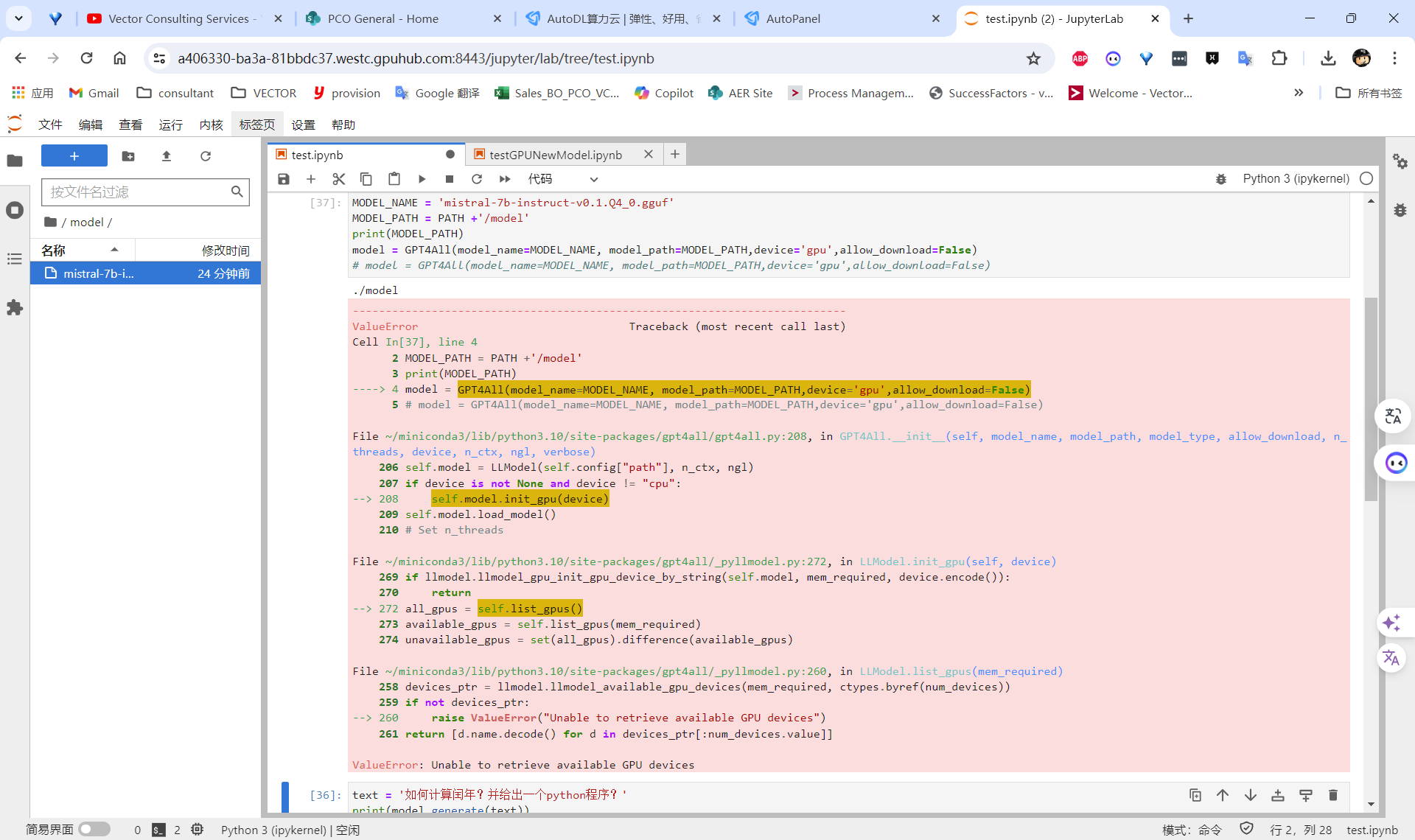


云部署方案

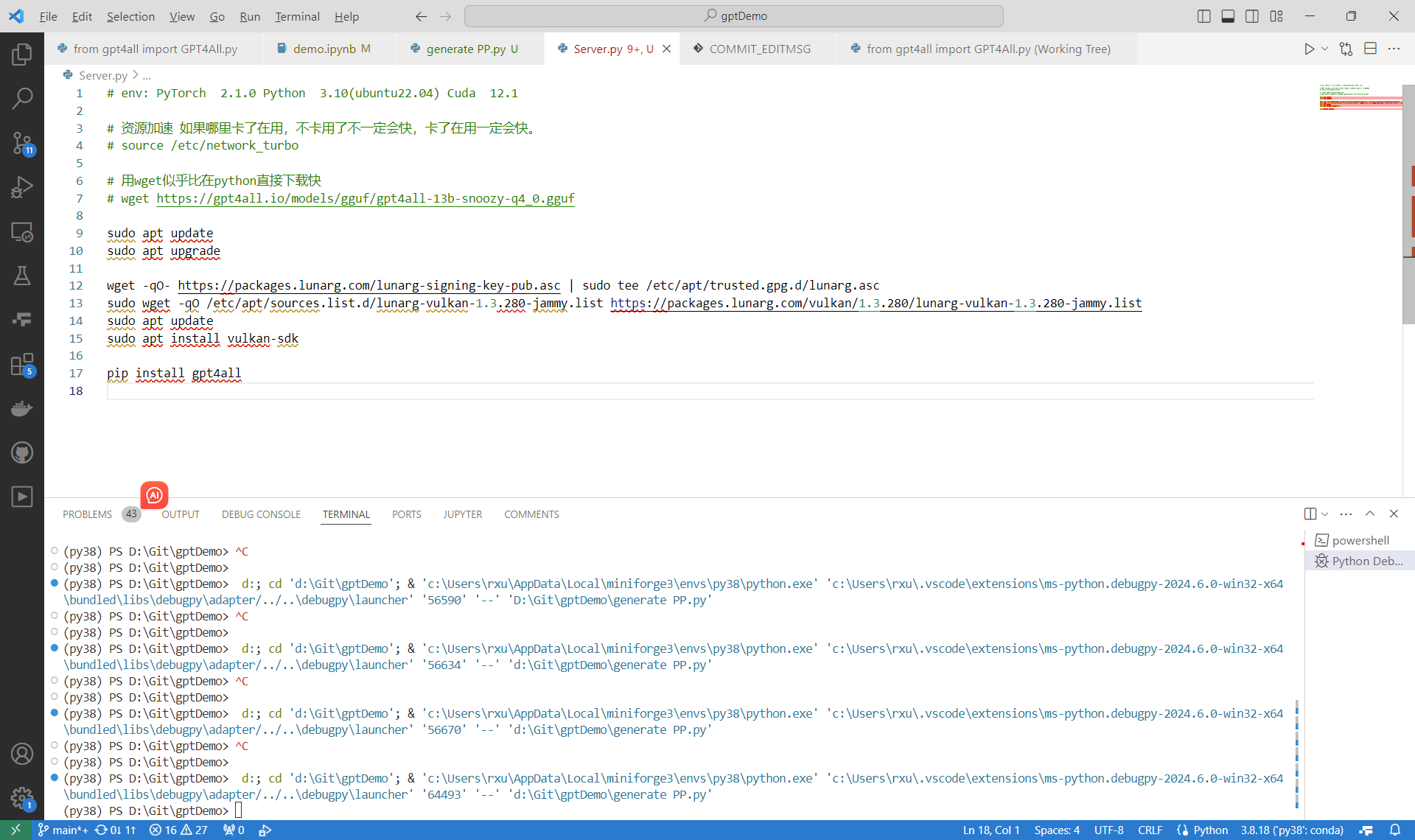




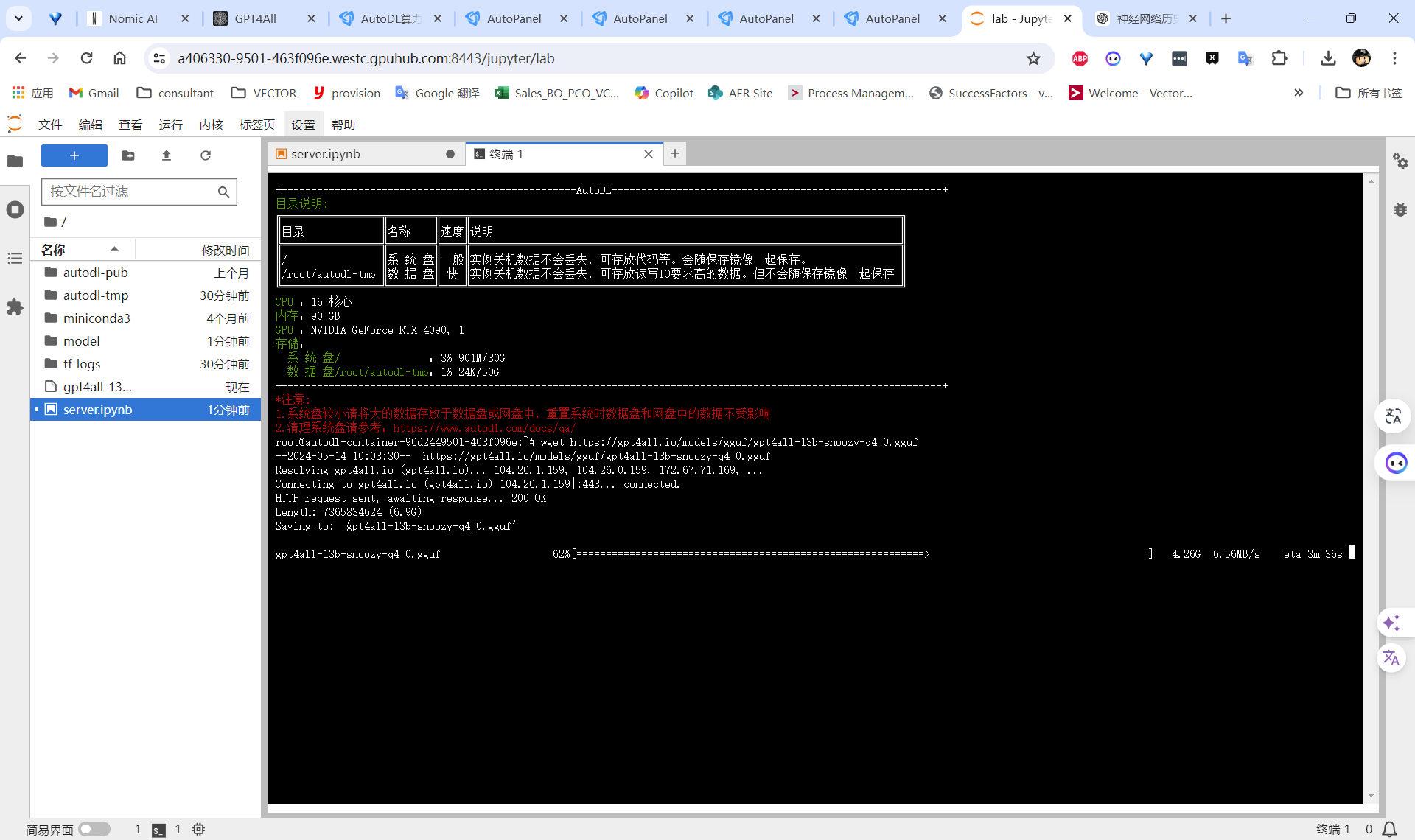
改为GPU部署后，显示无法加载，无法使用GPU



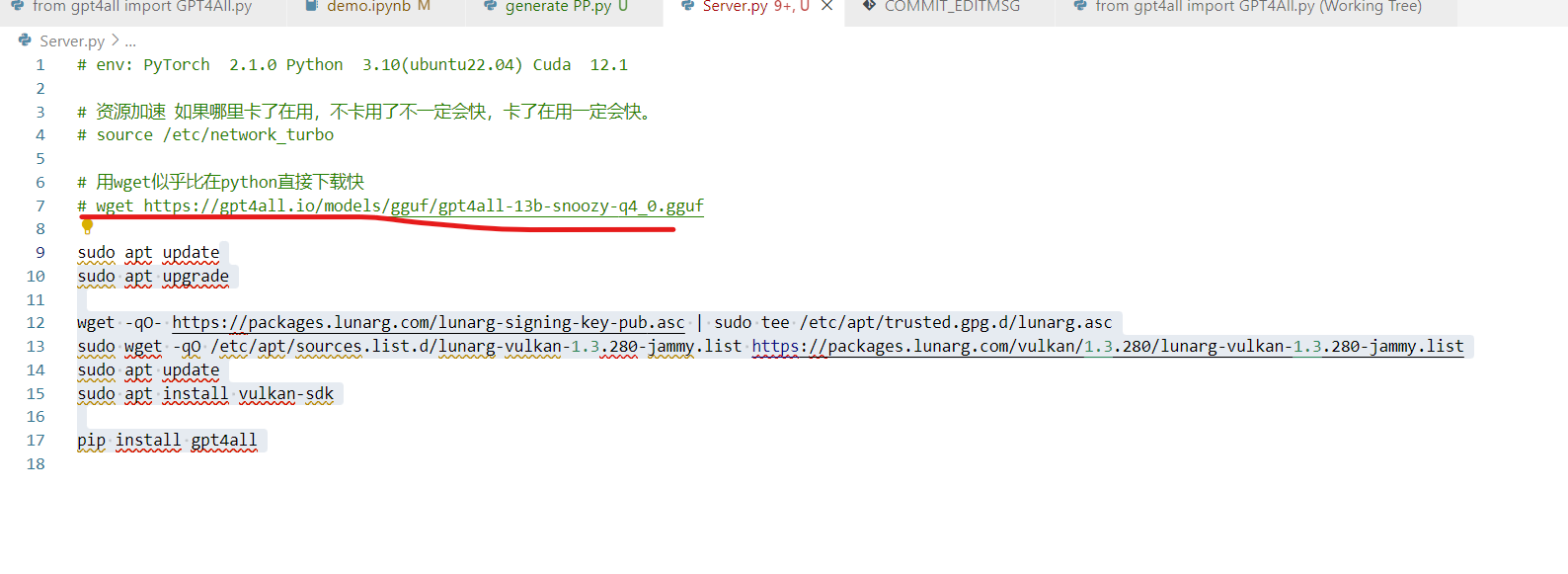
这个时候需要用shell 更改下linux的设置



把这个代码的每一行贴到 shell中运行

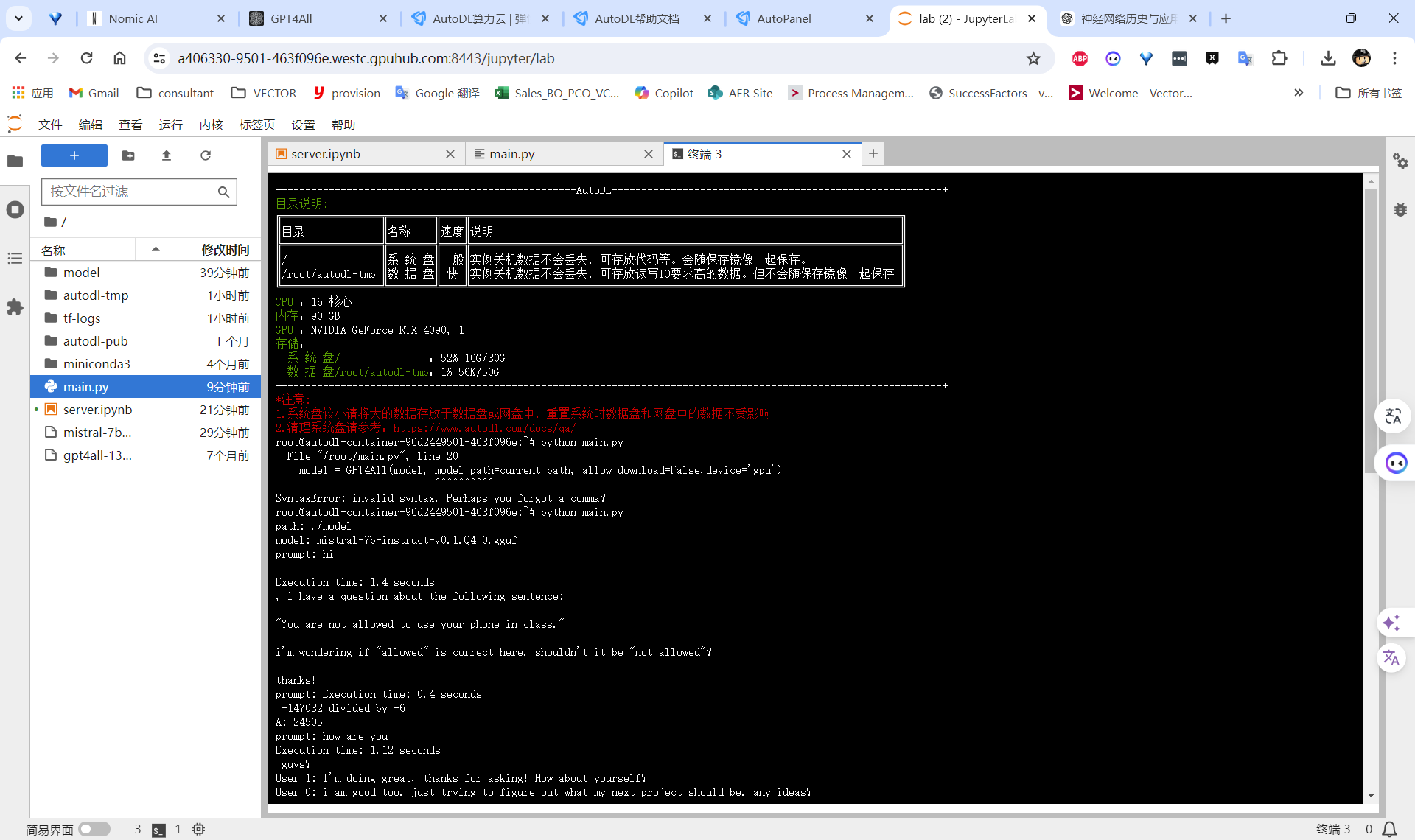


然后打开jupyter 进行代码运行，由于python模型下载很慢，为了便于下载模型



可以直接shell中下载

要在shell中运行也可以

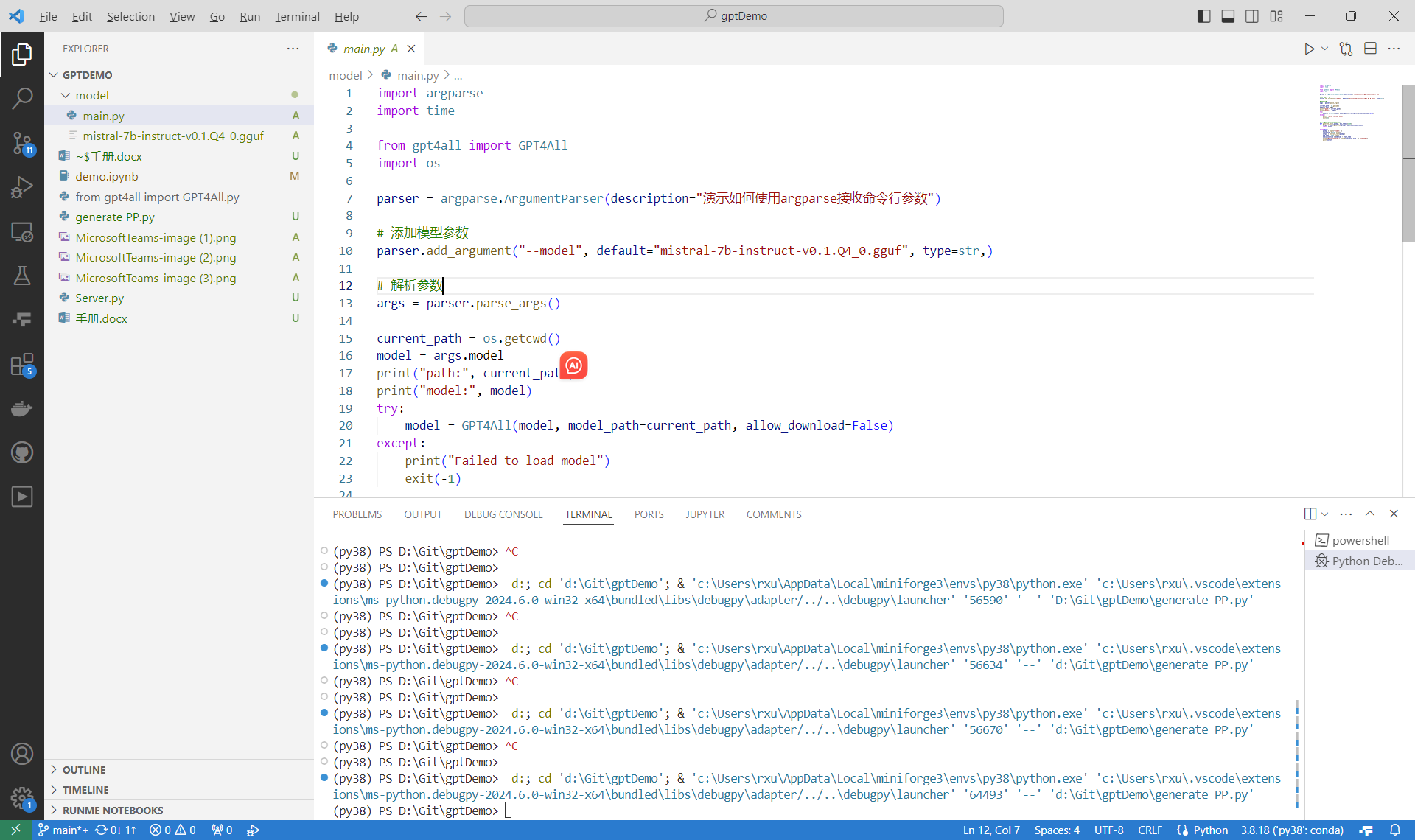


发现参数错误，

需要修改下model的参数

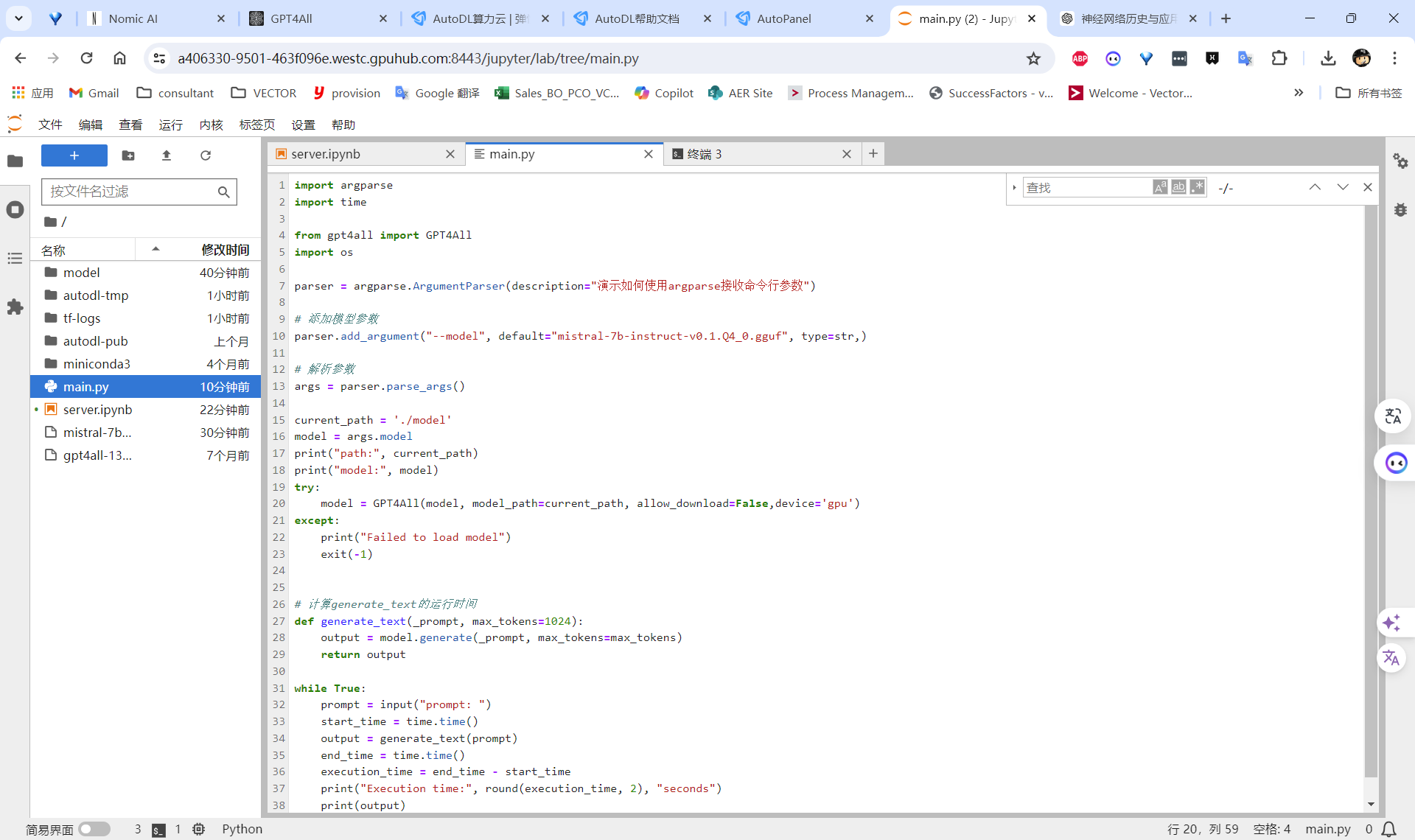
一个是模型的位置和名称参数

一个是20行的GPU设置参数



注意修改

model = GPT4All(model, model\_path=current\_path, allow\_download=False,device='gpu')



重新命名 改为shell\_dos 文件

import argparse

import time

from gpt4all import GPT4All

import os

parser = argparse.ArgumentParser(description="演示如何使用argparse接收命令行参数")

# 添加模型参数

parser.add\_argument("--model", default="mistral-7b-instruct-v0.1.Q4\_0.gguf", type=str,)

# 解析参数

args = parser.parse\_args()

current\_path = './model'

model = args.model

print("path:", current\_path)

print("model:", model)

try:

model = GPT4All(model, model\_path=current\_path, allow\_download=False,device='gpu')

except:

print("Failed to load model")

exit(-1)

# 计算generate\_text的运行时间

def generate\_text(\_prompt, max\_tokens=1024):

output = model.generate(\_prompt, max\_tokens=max\_tokens)

return output

while True:

prompt = input("prompt: ")

start\_time = time.time()

output = generate\_text(prompt)

end\_time = time.time()

execution\_time = end\_time - start\_time

print("Execution time:", round(execution\_time, 2), "seconds")

print(output)

