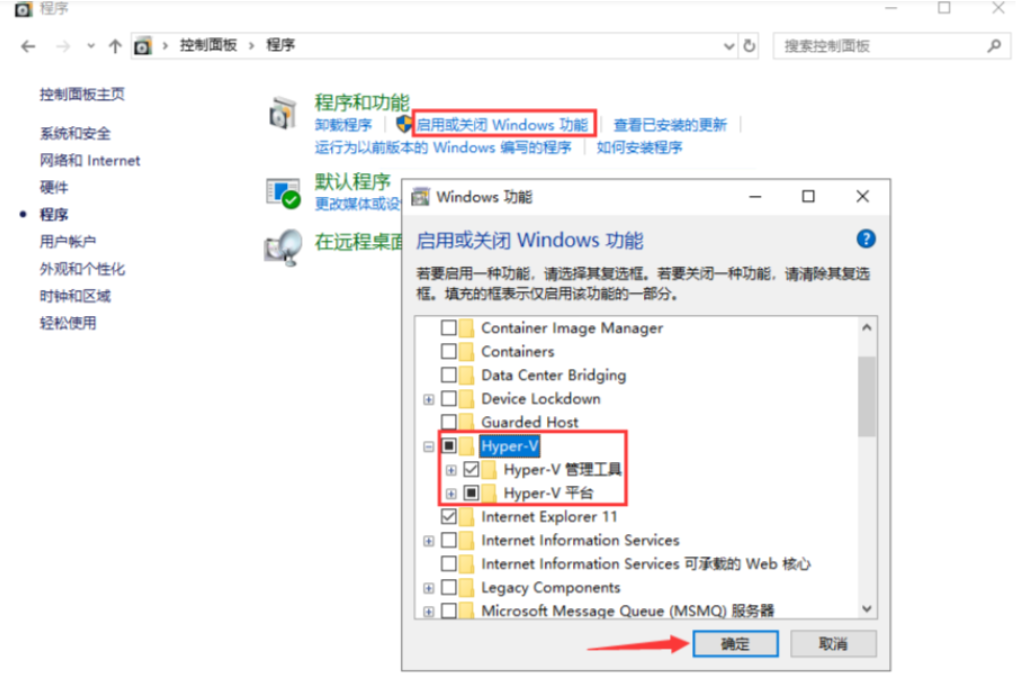
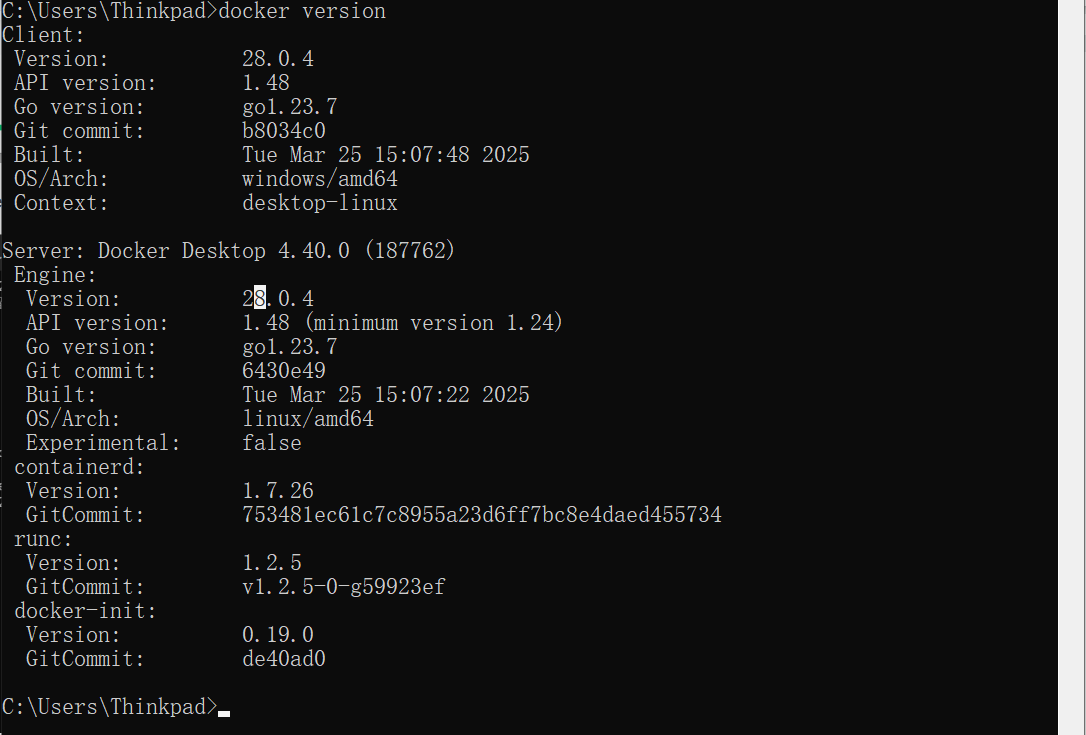
Maxkb本地部署

1. **windows下docker部署:**
2. 在 “控制面板” 打开 “程序”，然后点击 “启用或关闭 Windows 功能”，勾选 “[Hyper-V](https://zhida.zhihu.com/search?content_id=254864145&content_type=Article&match_order=1&q=Hyper-V&zd_token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJ6aGlkYV9zZXJ2ZXIiLCJleHAiOjE3NDQ1MDg0MDgsInEiOiJIeXBlci1WIiwiemhpZGFfc291cmNlIjoiZW50aXR5IiwiY29udGVudF9pZCI6MjU0ODY0MTQ1LCJjb250ZW50X3R5cGUiOiJBcnRpY2xlIiwibWF0Y2hfb3JkZXIiOjEsInpkX3Rva2VuIjpudWxsfQ.W0bEa4-FpgMWqvjfW7mskfu7-wiAGYJFFdhv8qhq4MQ&zhida_source=entity" \o "Hyper-V) 管理工具” 和 “Hyper-V 平台”。



1. 安装 wsl, 操作步骤参考：https://zhuanlan.zhihu.com/p/29209282361
2. 安装docker， 操作步骤参考：<https://zhuanlan.zhihu.com/p/29209282361>

cmd 打开命令行窗口输docker version检查，显示 docker 的版本表明安装成功



1. 进入 docker，选择 Settings – Docker Engine，将镜像源替换如下

{

"builder": {

"gc": {

"defaultKeepStorage": "20GB",

"enabled": true

}

},

"experimental": false,

"features": {

"buildkit": true

},

"registry-mirrors": [

"https://docker.m.daocloud.io",

"https://registry.docker-cn.com",

"https://cr.console.aliyun.com",

"https://mirror.ccs.tencentyun.com",

"https://dockerpull.org",

"https://docker.unsee.tech/",

"https://docker.1panel.live/",

"https://docker.udayun.com/",

"https://docker.nastool.de/",

"https://sad25hpw.mirror.aliyuncs.com",

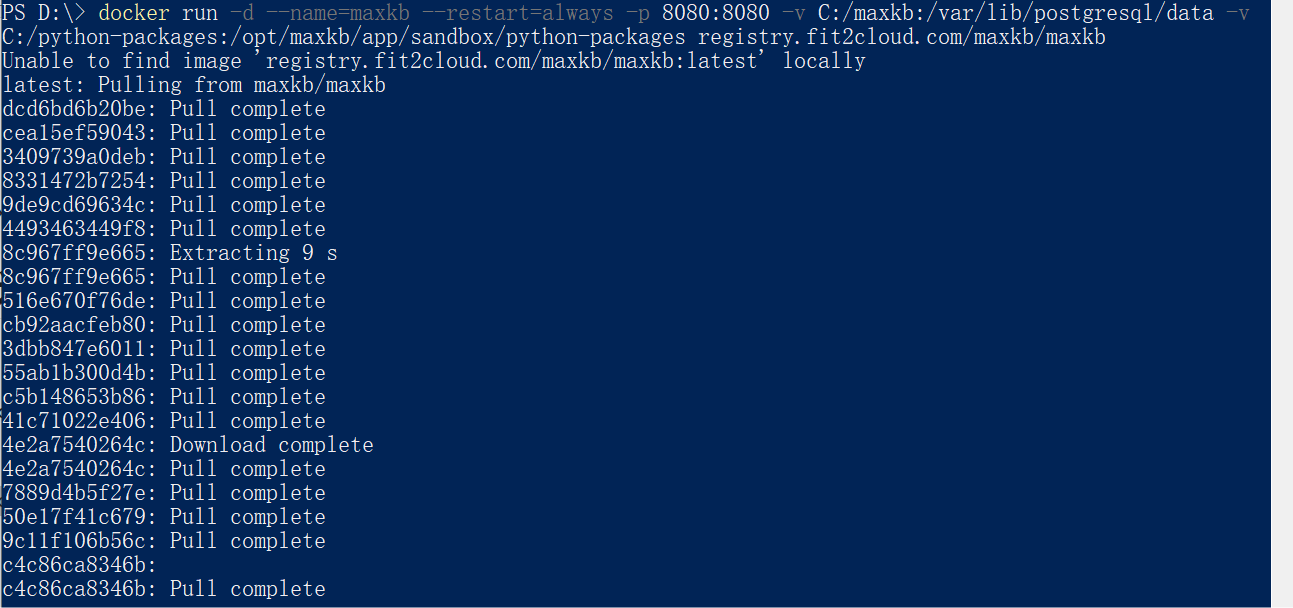
"https://docker.sunzishaokao.com",

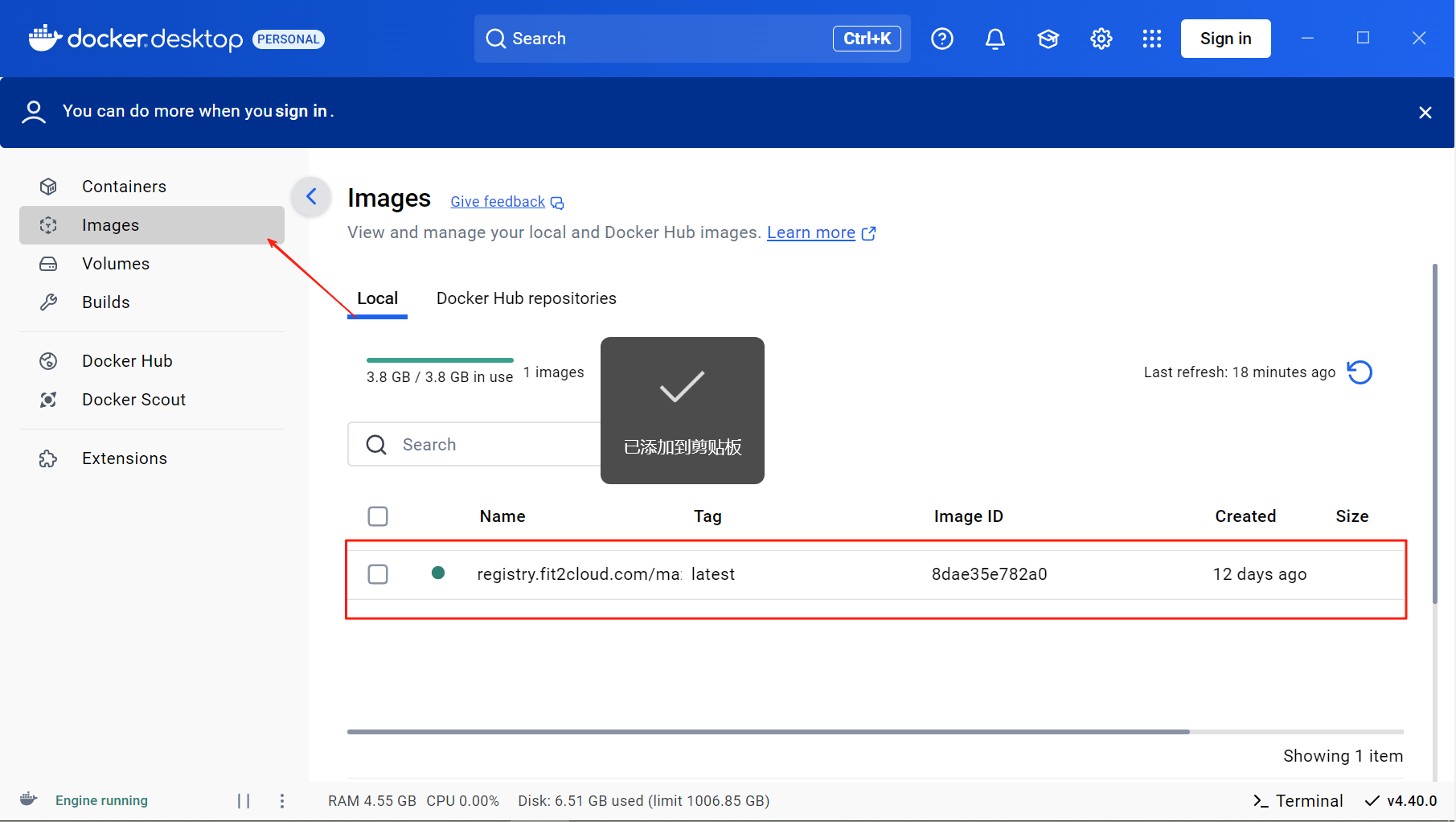
"https://docker.1panel.live"

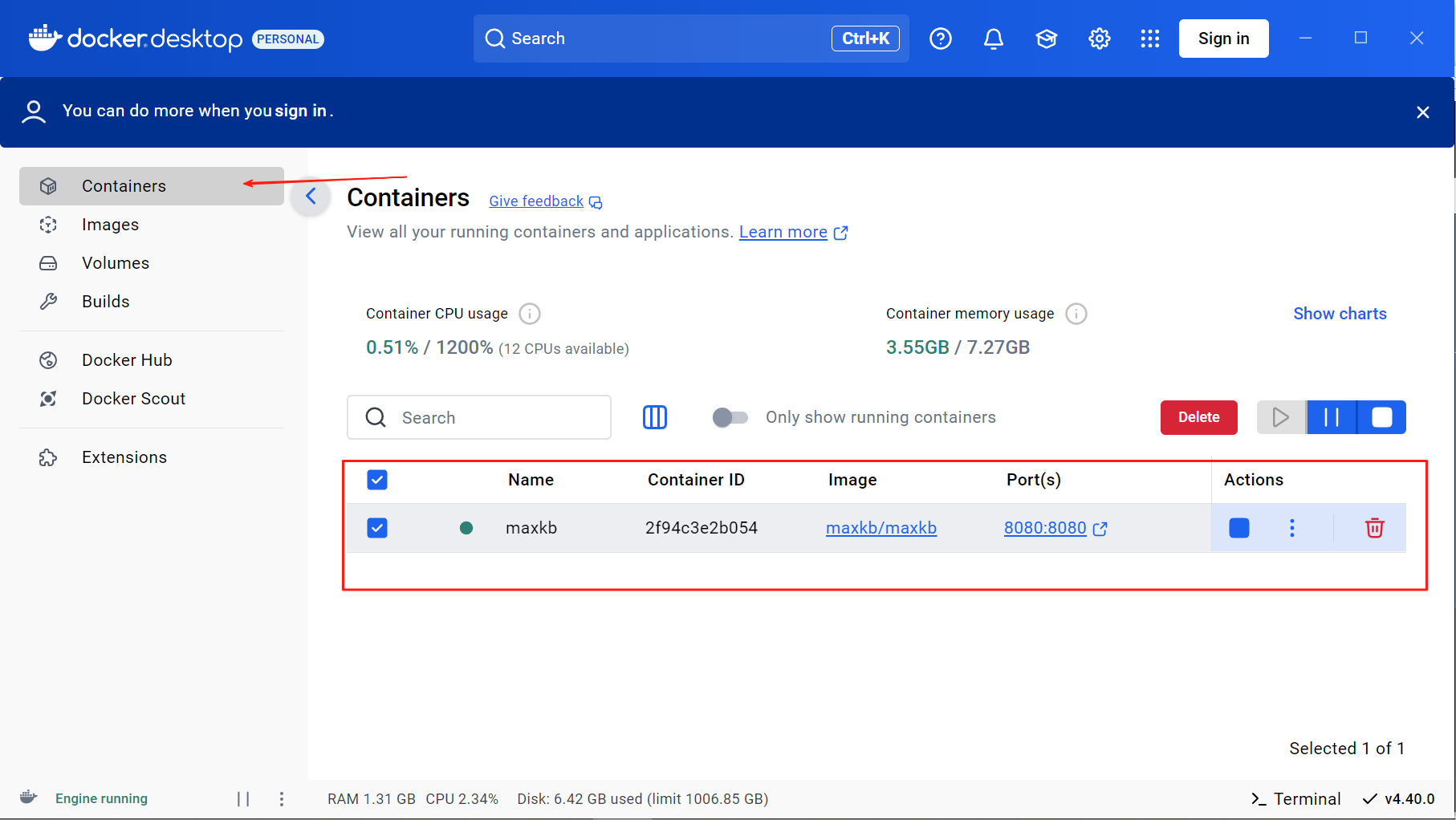
]

}

1. 在powershell中执行docker run -d --name=maxkb --restart=always -p 8080:8080 -v C:/maxkb:/var/lib/postgresql/data -v C:/python-packages:/opt/maxkb/app/sandbox/python-packages registry.fit2cloud.com/maxkb/maxkb4



1. 上述命令执行结束后可在docker中看到如下画面，则表示执行成功  
   



1. 点击8080:8080即可访问maxkb，账号密码：admin/MaxKB@123..



登录后如下，可进行应用，知识库， 函数库等创建



1. **linux下源码部署**
2. 首先创建conda虚拟环境, python版本最低要3.11版本

conda create -n maxkb python=3.11



conda activate maxkb

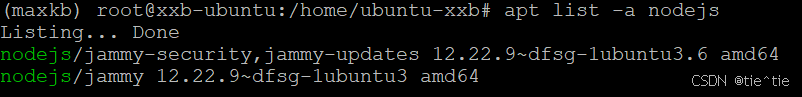


1. 拉取源码

git clone https://github.com/1Panel-dev/MaxKB.git

1. 安装前端依赖

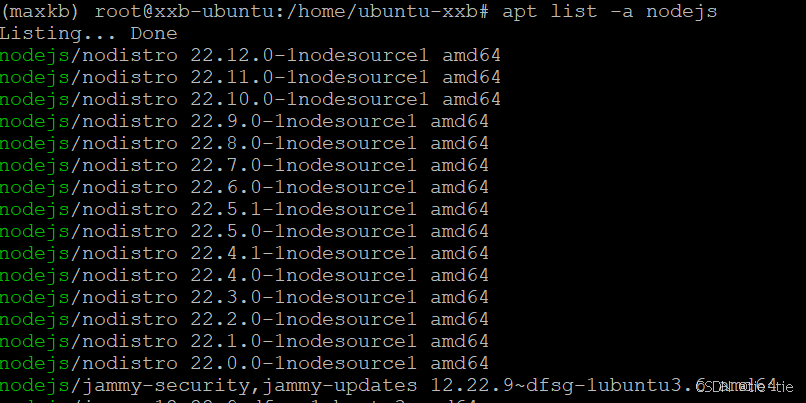
安装新版本的nodejs, 通过apt list -a nodejs来查看版本信息



发现没有比较新的22，23版本，运行下面的命令（如果没有curl需要提前安装）

curl -sL https://deb.nodesource.com/setup\_22.x | bash -

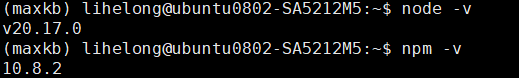
运行之后再使用apt list -a nodejs可发现已经可以使用apt下载较新版本的nodejs了



直接使用下面的命令，会自动下载较新版本：

apt install nodejs

安装结束可以使用node -v和npm -v分别查看node和npm的版本信息



此时安装好npm后，需要配置npm源，建议使用以下命令添加国内源：

npm config set registry https://registry.npmmirror.com

使用以下命令安装前端所需依赖（注意：执行下面的命令前，需要将目录切换到上面拉 取的MaxKB项目中ui目录下，即：MaxKB/ui/下执行）：

npm install

1. 后端依赖安装

安装poetry

pip install poetry==1.8.5

完成poetry的安装之后，即可使用下面的命令进行后端项目依赖的安装（注意：执行 下面的命令前，需要将目录切换到上面拉取的MaxKB项目根目录下，即：MaxKB下执 行，并且这一步骤需要一定时间）：

poetry install

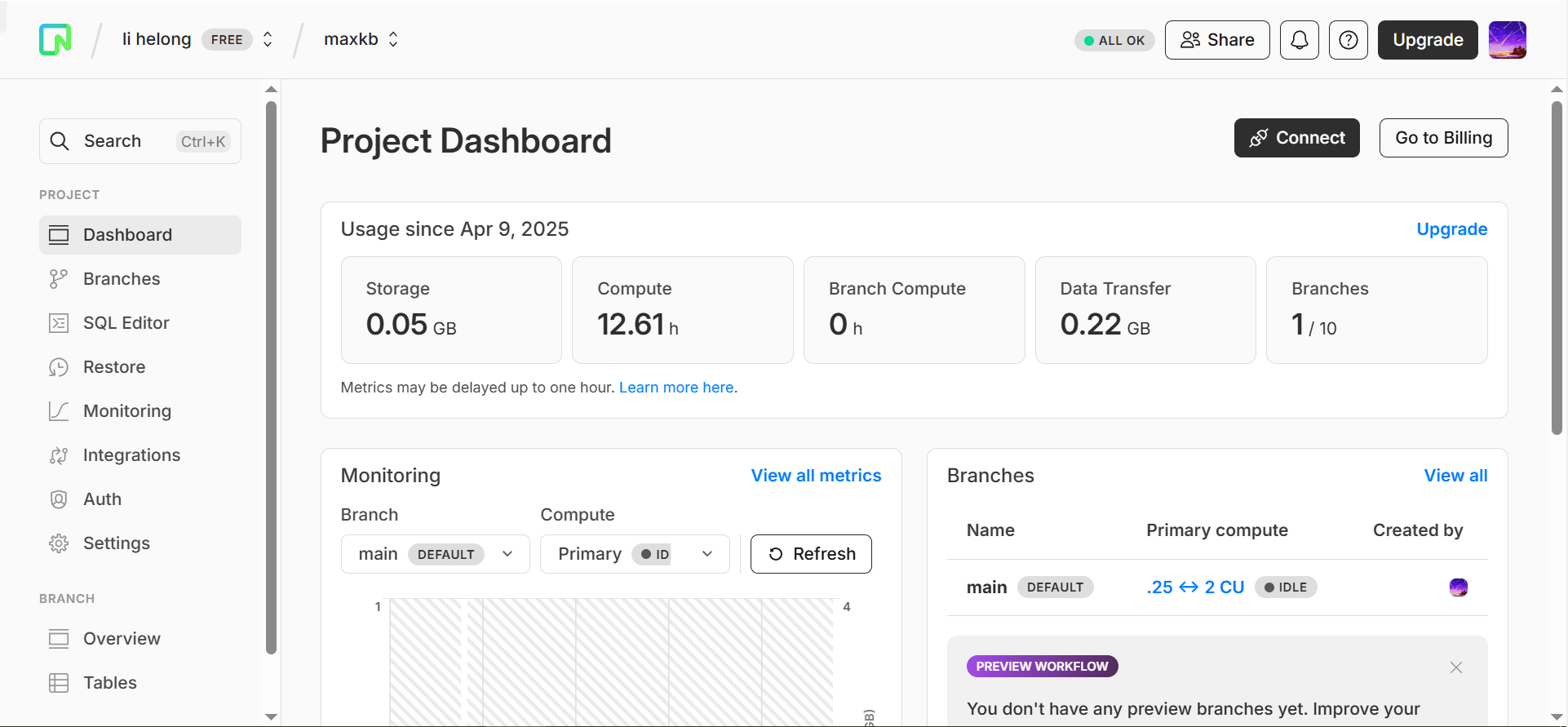
等待安装完成即可

1. Postgresql数据库的安装

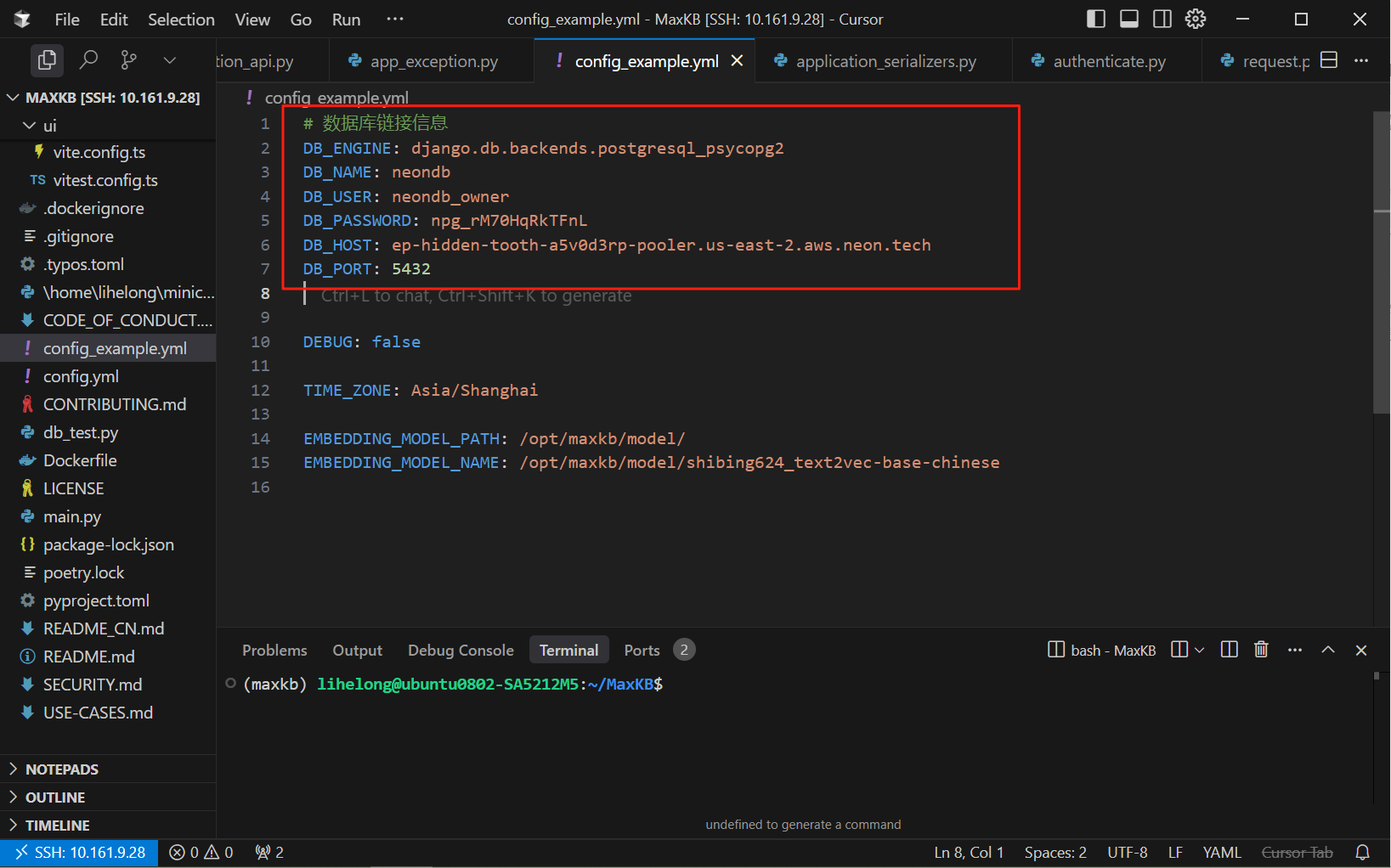
该数据库的安装需要root权限，因此本步跳过，改用在线postgresql版本(采用线上数 据库会存在访问慢的问题)。

本次测试中使用neon来创建postgresql数据库：

<https://console.neon.tech/app/projects/autumn-fog-52254869>



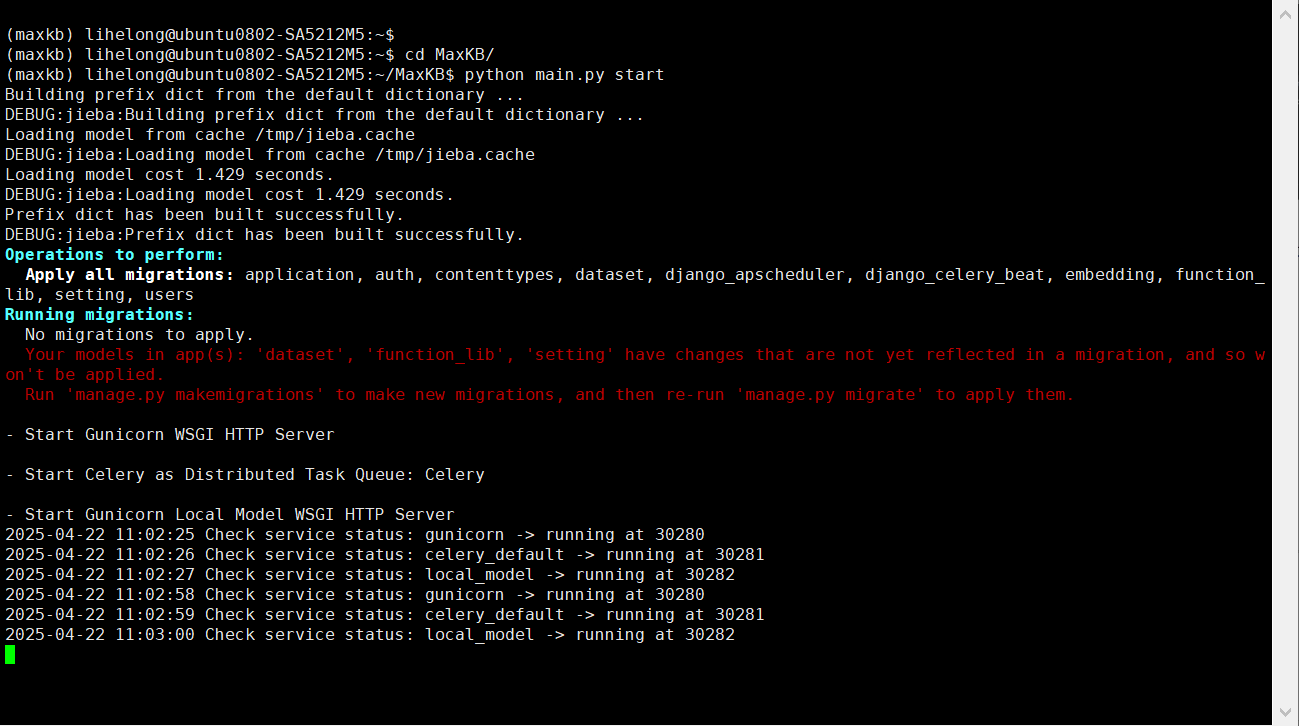
首次创建成功后会给一个url，该url中包括DB\_ENGINE,DB\_NAME,DB\_USER等信息，这 些信息需要填写到Maxkb项目中的config\_example.yml文件中



1. 启动项目

首先启动后端项目， 在Maxkb根目录下执行：

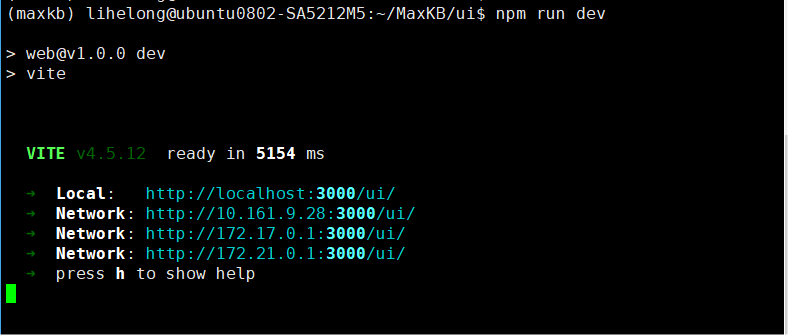
python main.py start



成功运行后在启动前端项目：

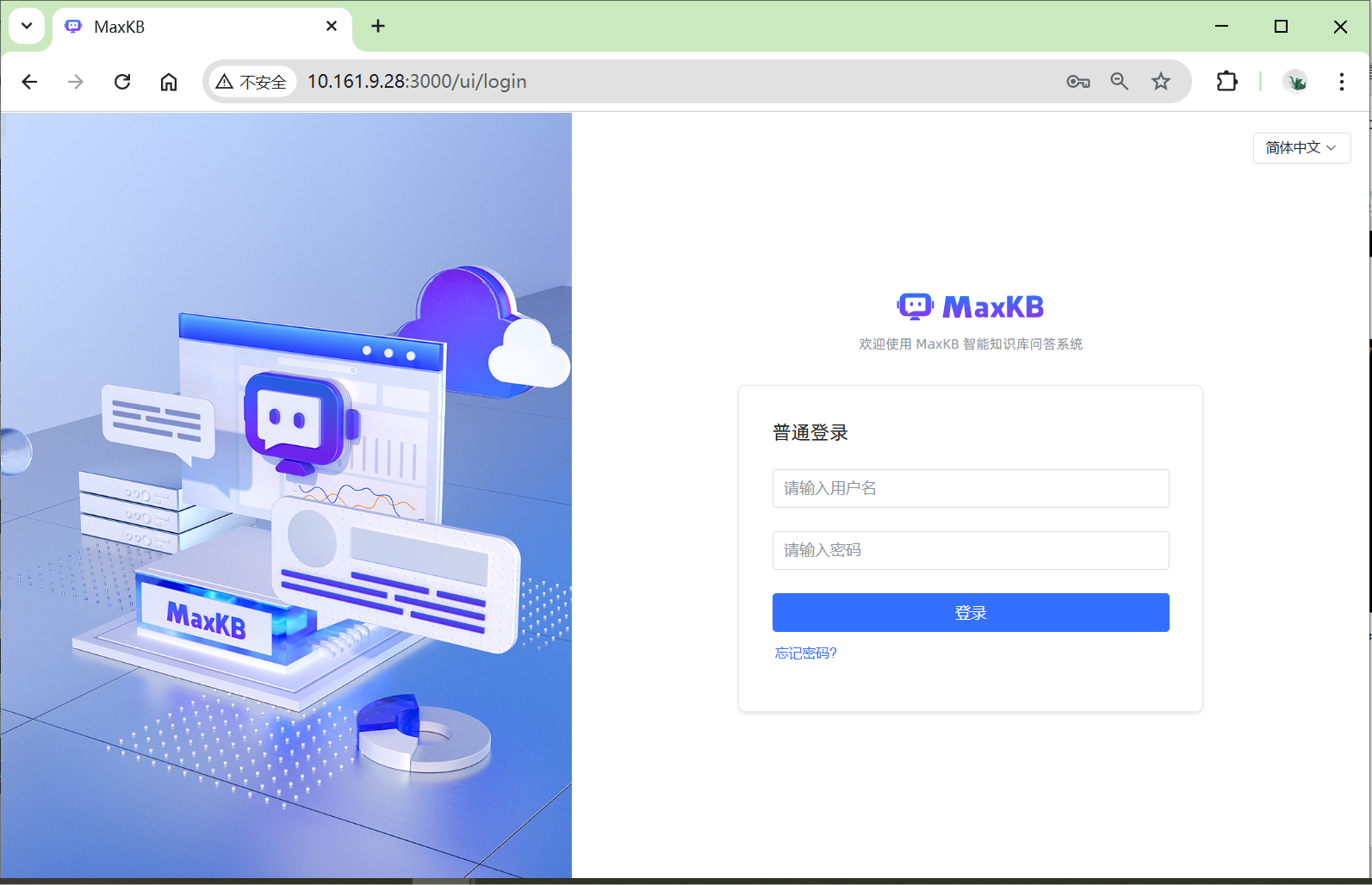
cd MaxKb/ui

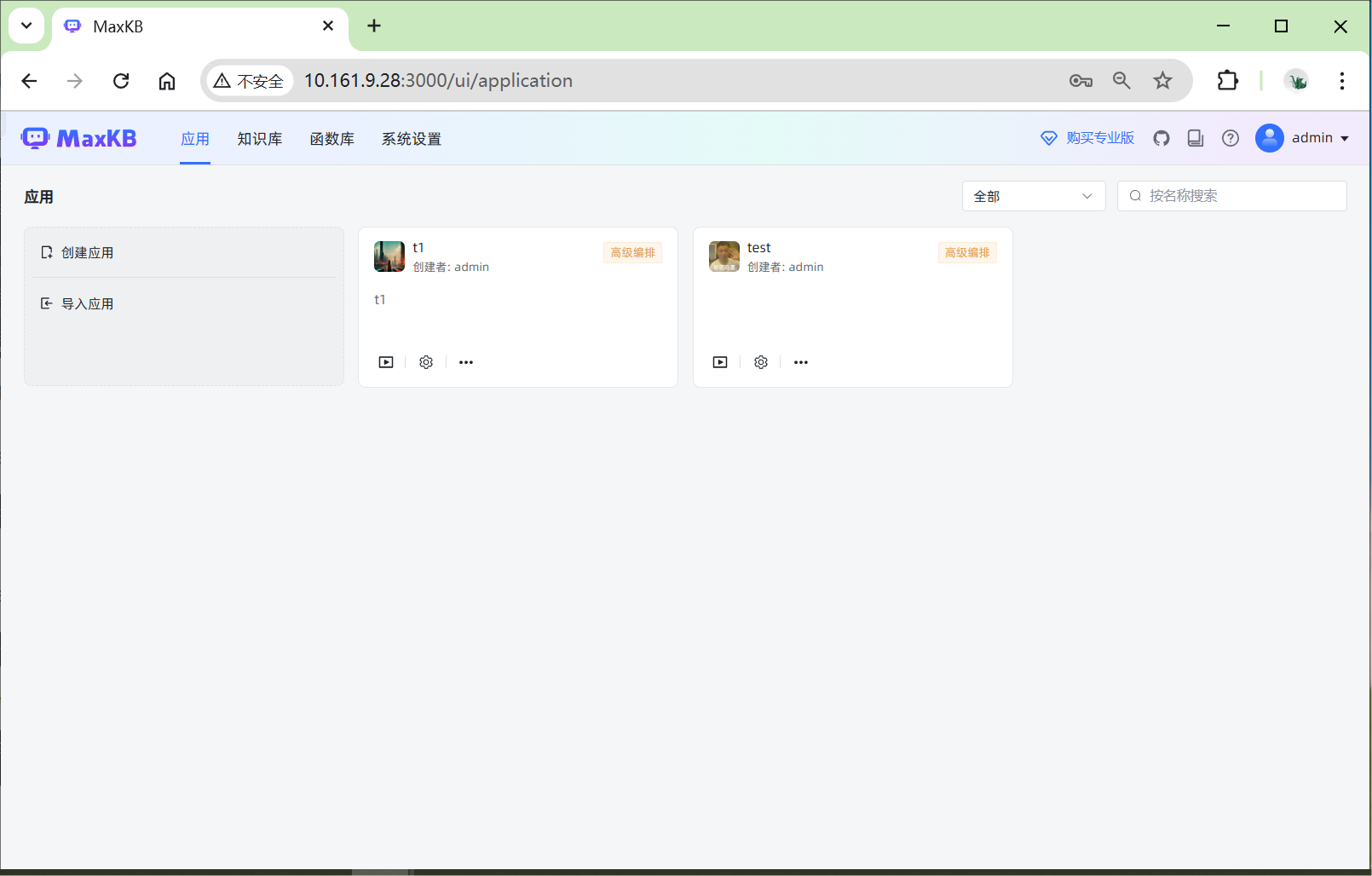
npm run dev



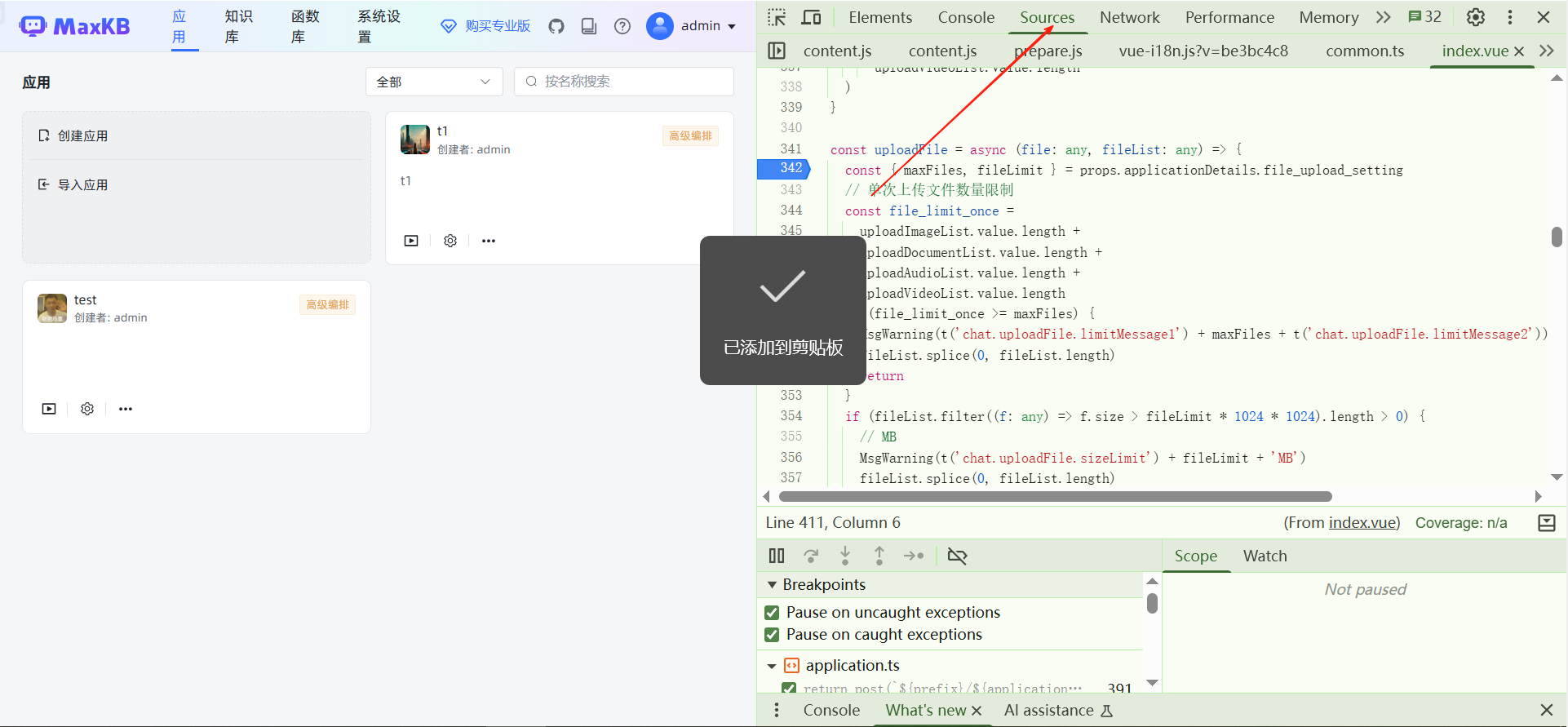
运行成功后则会给出链接地址，访问提送的链接地址即可访问MaxKb，

用户名密码:admin/MaxKB@123..





1. **前后端调试：**
2. 前端调试可直接打开F12--Sources找到对应的源文件添加断点



1. 后端调试
   1. 打开VSCode在左边点击【运行和调试】（快捷键Ctrl+Shift +D）
   2. 点击页面上“创建一个 launch.json 文件”或“运行和调试”
   3. 选择环境：Python
   4. VSCode 会自动帮你在 .vscode/ 文件夹下创建一个 launch.json
   5. 或者按 Ctrl+Shift+P → 输入 “Debug: Open launch.json”，添 加如下配置：

{

"version": "0.2.0",

"configurations": [

{

"name": "Django 后端调试",

"type": "debugpy",

"request": "launch",

"program": "${workspaceFolder}/apps/manage.py",//入口文件

"args": ["runserver", "8009"],

"env": {

"PATH": "/home/lihelong/miniconda3/envs/maxkb/bin:${env:PATH}"

},

"justMyCode": false,

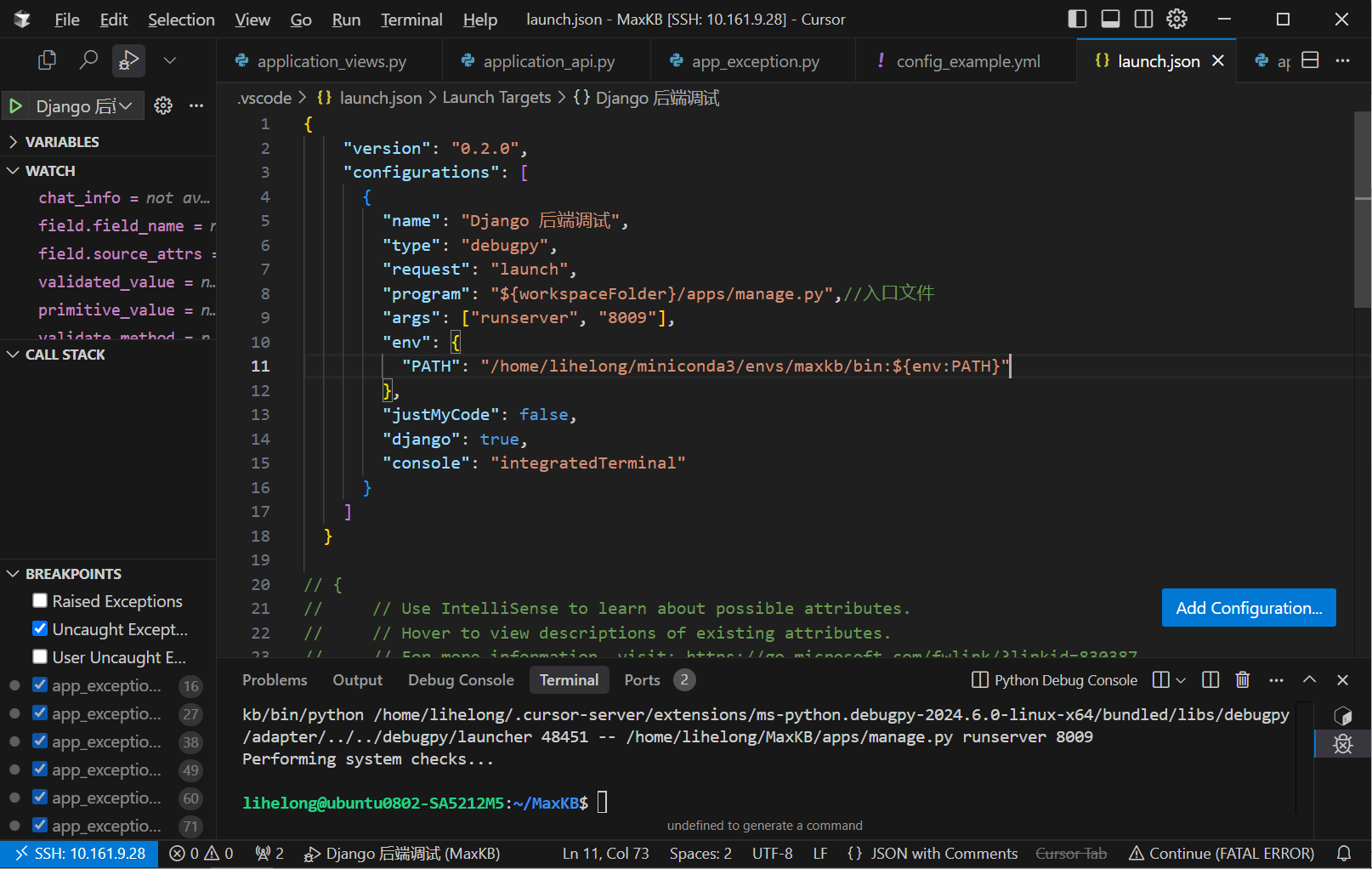
"django": true,

"console": "integratedTerminal"

}

]

}



* 1. 上述操作完成后则在vscode的debug界面就会出现一个绿色按钮 点击后就会启动后端服务
  2. 在对应文件中添加断点则可进入断点进行单步调试

1. **修改内容**
2. 修改社区版本应用限制5个限制
3. 文件上传增加eml文件
4. 文件上传增加提示语
5. 修改端口号以免出现占用

