

1 闯关任务

完成SSH连接与端口映射并运行hello_world.py

1.1 ssh连接

1. 申请开发机并启动



2. 创建ssh连接

- win下已有key在C:\Users\用户名\.ssh\
○ [新生成](#)

配置免密连接

3. win下cmd进行远程连接



```
C:\Users\dell>ssh -p 36407 root@ssh.intern-ai.org.cn -o StrictHostKeyChecking=no -o UserKnownHostsFile=/dev/null
Warning: Permanently added '[ssh.intern-ai.org.cn]:36407' (ED25519) to the list of known hosts.
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.10.134-13.al8.x86_64 x86_64)

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

----- 欢迎使用 InternStudio 开发机 -----
```

目录	名称	简介
/	系统目录	每次停止开发机会将其恢复至系统（镜像）初始状态。不建议存储数据。
/root	用户家目录	您的所有开发机共享此目录，不受开发机的启停影响。强烈建议将 conda 环境、代码仓库等所有数据存储在此目录下。 【注意】该目录有存储限额，超过限额后新写入的数据会被静默删除！
/share	共享目录	常用微调数据集、模型仓库、教程、xtuner 配置文件都存放在此。

Tips:

- 快速从本地上传文件：
`scp -o StrictHostKeyChecking=no -r -P {端口} {本地目录} root@ssh.intern-ai.org.cn:{开发机目录}`
 *注：在开发机 SSH 连接功能查看端口号

5. [vscode远程连接](#)（后续debug使用）

REMOTE EXPLORER Remotes (Tunnels/SSH) ...

RECENT REMOTES (TUNNELS/SSH)

- SSH
- ssh.intern-ai.org.cn connected

Start

- New File...
- Open File...
- Open Folder...
- Clone Git Repository...
- Connect to...

目录	名称	简介
/	系统目录	每次停止开发机会将其恢复至系统（镜像）初始状态。不建议存储数据。
/root	用户家目录	您的所有开发机共享此目录，不受开发机的启停影响。强烈建议将 conda 环境、代码仓库等所有数据存储在此目录下。 【注意】该目录有存储限额，超过限额后新写入的数据会被静默删除！
/share	共享目录	常用微调数据集、模型仓库、教程、xtuner 配置文件都存放在此。

Tips:

- 快速从本地上传文件：
`scp -o StrictHostKeyChecking=no -r -P {端口} {本地目录} root@ssh.intern-ai.org.cn:{开发机目录}`
 *注：在开发机 SSH 连接功能查看端口号
- 避免因终端关闭或 SSH 连接断开导致任务终止，强烈建议使用 <https://www.ruanyifeng.com/blog/2019/10/tmux.html>
- 查看 GPU 显存和算力使用率：studio-smi
- 使用InternStudio开箱即用的conda环境：
`studio-conda -h`
- 将conda环境一键添加到jupyterlab：
`lab add {YOUR_CONDA_ENV_NAME}`

Terminal: (base) root@intern-studio-50105460:~#

1.2 端口映射

- 创建示例web服务hello_world.py

```

# ssh连接开发机
# vim hello_world.py
import socket
import re
import gradio as gr

# 获取主机名
def get_hostname():
    hostname = socket.gethostname()
    match = re.search(r'-(\d+)$', hostname)
    name = match.group(1)

    return name

# 创建 Gradio 界面
with gr.Blocks(gr.themes.Soft()) as demo:
    html_code = f"""
        <p align="center">
            <a href="https://intern-ai.org.cn/home">
                
            </a>
        </p>
        <h1 style="text-align: center;">👋 welcome {get_hostname()} user, welcome to the ShuSheng LLM Practical Camp Course!</h1>
        <h2 style="text-align: center;">🚀 Let's go on a journey through Shusheng Island together.</h2>
        <p align="center">
            <a href="https://github.com/InternLM/Tutorial/blob/camp3">
                
            </a>
        </p>
    """
    gr.Markdown(html_code)

demo.launch()

```

- 安装必要依赖 `pip install gradio==4.29.0`
- 运行服务 `python hello_world.py`
- 另起一个cmd，执行映射命令

```

ssh -p 36407 root@ssh.intern-ai.org.cn -CNg -L 7860:127.0.0.1:7860 -o
StrictHostKeyChecking=no

#ssh -p 37367 root@ssh.intern-ai.org.cn -CNg -L {本地机器_PORT}:127.0.0.1:{开发机_PORT} -o StrictHostKeyChecking=no

```

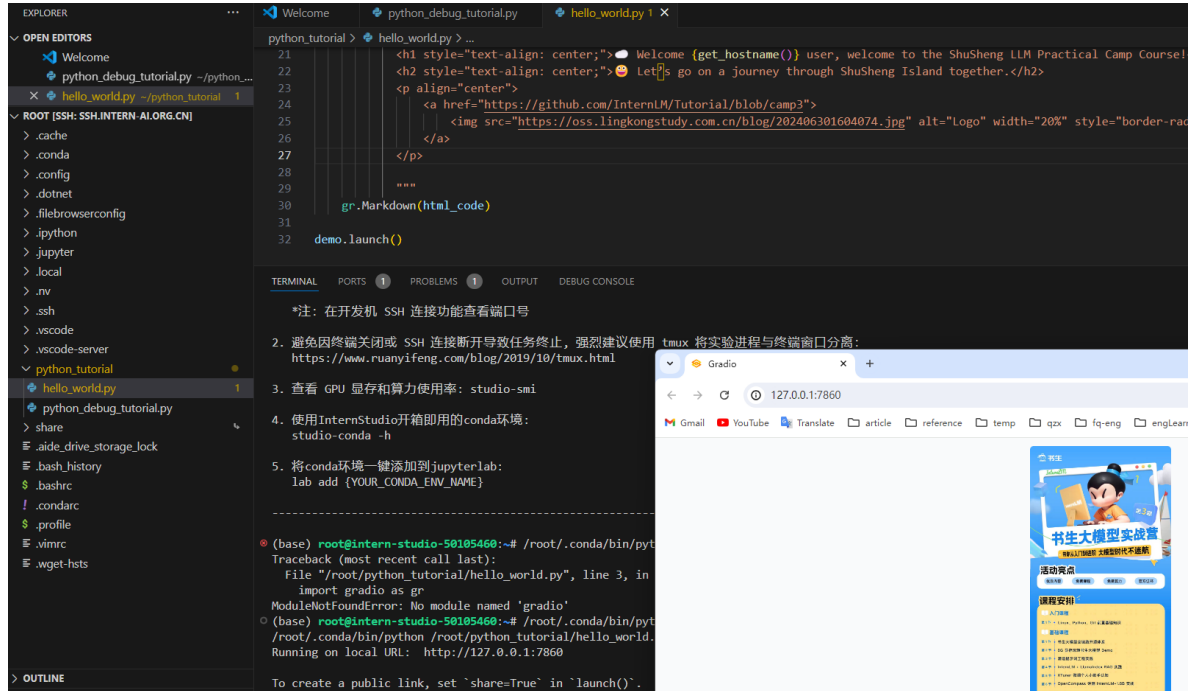
命令各部分的含义：

- `-p 37367`：是指定 SSH 连接的端口为 37367，**注意修改为自己开发机的。**
- `root@ssh.intern-ai.org.cn`：表示要以 `root` 用户身份连接到 `ssh.intern-ai.org.cn` 这个主机。

-Cng

:

- `-C` 通常用于启用压缩。
- `-N` 表示不执行远程命令，仅建立连接用于端口转发等。
- `-g` 允许远程主机连接到本地转发的端口。
- `-L {本地机器_PORT}:127.0.0.1:{开发机_PORT}`：这是设置本地端口转发，将本地机器的指定端口（由 `{本地机器_PORT}` 表示）转发到远程主机（这里即 `ssh.intern-ai.org.cn`）的 `127.0.0.1`（即本地回环地址）和指定的开发机端口（由 `{开发机_PORT}` 表示）。
- `-o StrictHostKeyChecking=no`：关闭严格的主机密钥检查，这样可以避免第一次连接时因为未知主机密钥而产生的提示或错误。



PS: 不知道是不是bug。我这里并没有端口映射操作。仅就配置了免密登录和vscode连接

2 可选任务

2.1 可选任务 1

将Linux基础命令在开发机上完成一遍

2.2 可选任务 2

使用 VSCODE 远程连接开发机并创建一个conda环境

2.3 可选任务 3

创建并运行test.sh文件