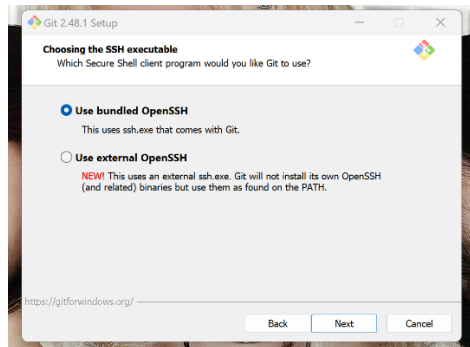


第一周学习报告—王艳-22024110137

1. **ssh:** SSH 是 Secure Shell 的缩写，它是一种网络协议，主要用于在不安全的网络环境中提供安全的远程登录和其他网络服务。ssh 是互联网上加密通讯的工具。

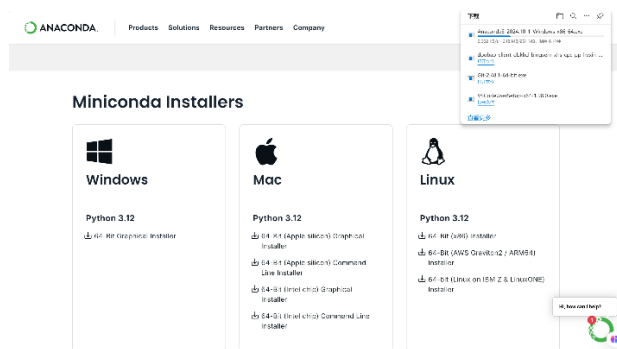


2. **miniconda 和 anaconda 的区别:**

Miniconda 和 Anaconda 都是用于管理 Python 环境和包的工具，由同一团队开发。

Anaconda: 功能非常全面，适合初学者以及需要快速搭建数据科学、机器学习等工作环境的用户。它预装了众多数据科学和机器学习相关的库和工具，用户可以直接使用，无需再单独安装，大大节省了配置环境的时间。

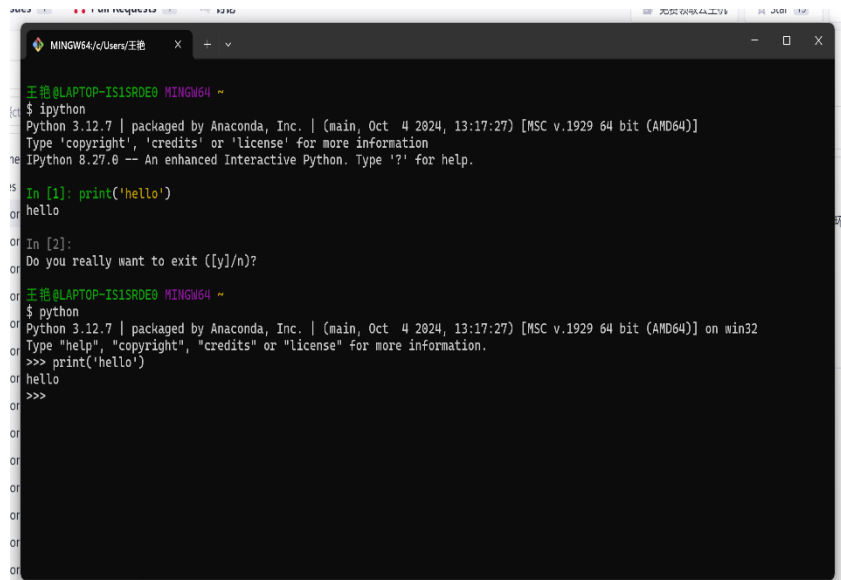
Miniconda: 相对来说功能较为基础，只提供了 conda 的核心功能和 Python 的基础环境。用户需要根据自己的需求，使用 conda 命令手动安装所需的库和工具，但这种方式更加灵活，用户可以根据自己的实际需求定制环境，避免安装大量不需要的软件包，从而减少系统资源的占用。



3. **ipython 和 python 区别:**

Python: Python 提供了一个简单的命令行界面，用户输入 Python 代码，按下回车键后立即执行，并显示执行结果。但它的历史记录查看、代码编辑等功能不够便捷，只能通过简单的上下箭头键来查看之前输入的命令。例如，在 Python 解释器中，你输入 `print("Hello, World!")` 并回车，会直接输出结果，但如果要修改上一条命令，操作起来不太方便。

IPython: IPython 着重提升了交互性，拥有更强大的命令行编辑功能。它支持使用类似 Unix shell 的快捷键，如 Ctrl + A 移动到行首、Ctrl + E 移动到行尾，还能使用 Ctrl + R 进行历史命令的反向搜索。此外，IPython 会以不同颜色区分代码的不同部分，提高可读性，并且能对输入的代码进行语法高亮显示。



```
MINGW64/c/Users/王艳 X + -
王艳@LAPTOP-IS1SRDE0 MINGW64 ~
$ ipython
Python 3.12.7 | packaged by Anaconda, Inc. | (main, Oct 4 2024, 13:17:27) [MSC v.1929 64 bit (AMD64)]
Type 'copyright', 'credits' or 'license' for more information
IPython 8.27.0 -- An enhanced Interactive Python. Type '?' for help.

In [1]: print('hello')
hello

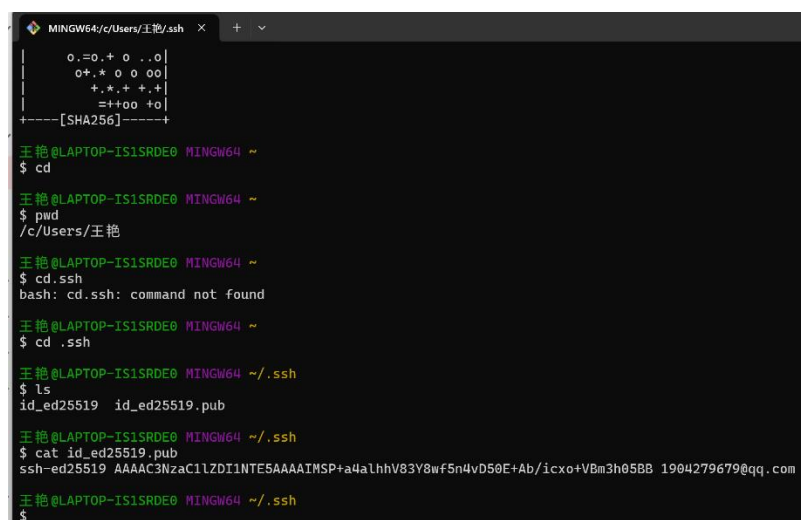
In [2]:
Do you really want to exit ([y]/n)?

王艳@LAPTOP-IS1SRDE0 MINGW64 ~
$ python
Python 3.12.7 | packaged by Anaconda, Inc. | (main, Oct 4 2024, 13:17:27) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print('hello')
hello
>>>
```

4. 公钥私钥加密

公钥：是公开的密钥，任何人都可以获取。它用于对数据进行加密，确保只有拥有相应私钥的接收者才能解密并读取数据。

私钥：是与公钥配对的、属于用户自己的密钥，需要严格保密。私钥用于对使用相应公钥加密的数据进行解密，以及对数据进行数字签名，以证明数据的来源和完整性。



```
MINGW64/c/Users/王艳/.ssh X + -
| 0.=0.+ 0 ..0 |
| 0+.* 0 0 00 |
| +.*+ +.+ |
| =+00 +0 |
+-----[SHA256]-----+

王艳@LAPTOP-IS1SRDE0 MINGW64 ~
$ cd

王艳@LAPTOP-IS1SRDE0 MINGW64 ~
$ pwd
/c/Users/王艳

王艳@LAPTOP-IS1SRDE0 MINGW64 ~
$ cd .ssh
bash: cd: .ssh: command not found

王艳@LAPTOP-IS1SRDE0 MINGW64 ~
$ cd .ssh

王艳@LAPTOP-IS1SRDE0 MINGW64 ~/.ssh
$ ls
id_ed25519 id_ed25519.pub

王艳@LAPTOP-IS1SRDE0 MINGW64 ~/.ssh
$ cat id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIMSP+a4a1hhV83Y8wf5n4vD50E+Ab/ixco+VBm3h05BB 1904279679@qq.com

王艳@LAPTOP-IS1SRDE0 MINGW64 ~/.ssh
$
```

5. 完成 Git 的全局配置

```
MINGW64/c/Users/王艳/.ssh
王艳@LAPTOP-IS1SRDE0 MINGW64 ~
$ cd .ssh

王艳@LAPTOP-IS1SRDE0 MINGW64 ~/.ssh
$ ls
id_ed25519 id_ed25519.pub

王艳@LAPTOP-IS1SRDE0 MINGW64 ~/.ssh
$ cat id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIMSP+a4alhhV83Y8wF5n4vD50E+Ab/icxo+VBm3h05BB 1904279679@qq.com

王艳@LAPTOP-IS1SRDE0 MINGW64 ~/.ssh
$ ssh -T git@gitcode.com
The authenticity of host 'gitcode.com (116.205.2.91)' can't be established.
RSA key fingerprint is SHA256:aTlSy+4ARMC7nWyy5eKIqUkotk8yv7Jd+XXoP4EXj1Y.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'gitcode.com' (RSA) to the list of known hosts.
remote: Welcome to GitCode, bunny12345

王艳@LAPTOP-IS1SRDE0 MINGW64 ~/.ssh
$ ssh -T git@gitcode.com
remote: Welcome to GitCode, bunny12345

王艳@LAPTOP-IS1SRDE0 MINGW64 ~/.ssh
$ git config --global user.name bunny12345
git config --global user.email bunny12345@noreply.gitcode.com

王艳@LAPTOP-IS1SRDE0 MINGW64 ~/.ssh
$
```