5、初次运行git前的配置

（1）git config后面接相应命令，用来配置或读取相应的工作环境变量。这些变量可以存放在以下三个不同的地方：

1. **/etc/gitconfig**（windows系统在git安装目录下）：系统中对所有用户都普遍适用的配置。

配置命令**git config --system ...**读写的就是这个文件。

1. **~/.gitconfig**（一般在用户主目录下，主目录即 $HOME 变量指定的目录，windows系统一般在这C:\Users\..）：配置命令**git config --global ...**读写的就是这个文件。
2. 当前项目的git目录中的配置文件（也就是工作目录中的**.git/config** 文件）：这里的配置仅仅针对当前项目有效。配置命令为**git config ...**。配置过程是找到项目所在目录下的 .git，进入.git文件夹执行配置命令。
3. 每一个级别的配置都会覆盖上层的相同配置，所以.git/config里的配置会覆盖 /etc/gitconfig和~/.gitconfig中的同名变量。
4. 用户信息

第一个要配置的是你个人的用户名称和电子邮件地址。

例：

用户名称：git config --global user.name "John Doe"

电子邮件：git config --global user.email [johndoe@example.com](mailto:johndoe@example.com)

1. 文本编辑器

Git需要你输入一些额外消息的时候，会自动调用一个外部文本编辑器给你用。默认会使用操作系统指定的默认编辑器，一般可能会是 Vi 或者 Vim。如果你有其他偏好，比如 vscode 的话，可以重新设置：

git config --global core.editor "D:\\Microsoft VS Code\\Code.exe"

1. 差异分析工具

一个比较常用的解决合并冲突时使用哪种差异分析工具，比如要改用vimdiff的话：

git config --global merge.tool vimdiff

Git 可以理解kdiff3，tkdiff，meld，xxdiff，emerge，vimdiff，gvimdiff，ecmerge和opendiff 等合并工具的输出信息。当然，你也可以指定使用自己开发的工具。

1. 查看配置信息

A、检查已有的配置信息，可以使用 git config --list 命令。有时候会看到重复的变量名，那就说明它们来自不同的配置文件（比如/etc/gitconfig 和 ~/.gitconfig），不过最终Git实际采用的是最后一个。

B、也可以直接查阅某个环境变量的设定，只要把特定的名字跟在后面即可，像这样：

git config user.name

1. 获取帮助

想了解 Git 的各式工具该怎么用，可以阅读它们的使用帮助，方法有三：

（1）git help <verb>

（2）git <verb> --help

（3）man git-<verb>

比如，要学习config命令可以怎么用，运行：git help config或git config --help或man git-config。