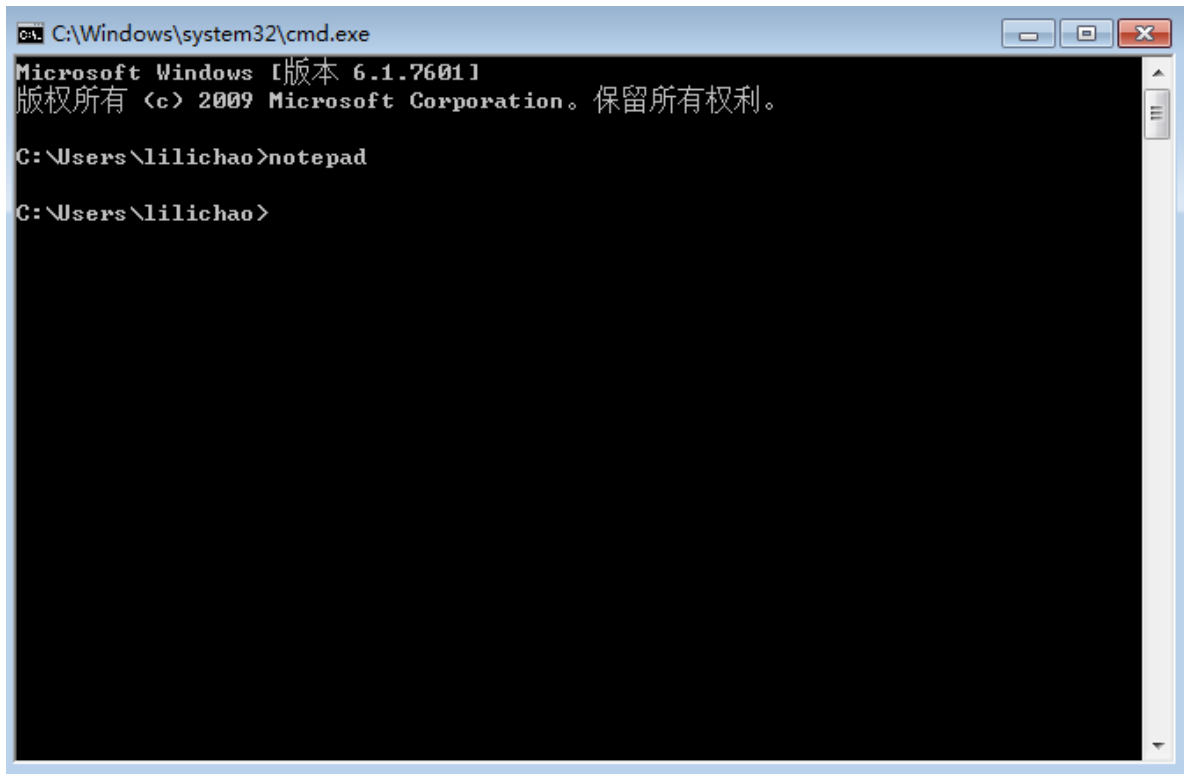


# 第一天

## 命令行

- 之前操作计算机通常都是通过图形化界面的形式来完成的（GUI）
- UI（User Interface）用户接口，用户通过接口来使用程序（设备）
- 命令行实际就是通过文字指令的来完成对计算机的操作（TUI）



- 命令行有很多的名字：命令行、CMD窗口、Terminal、终端、Shell ...

## 命令行的使用

### 1. 打开命令行

- 使用 win + r 输入 cmd 回车

### 2. 命令行窗口的结构

- 当前所在的路径

```
C:\Users\lilichao
```

- 命令提示符

```
>
```

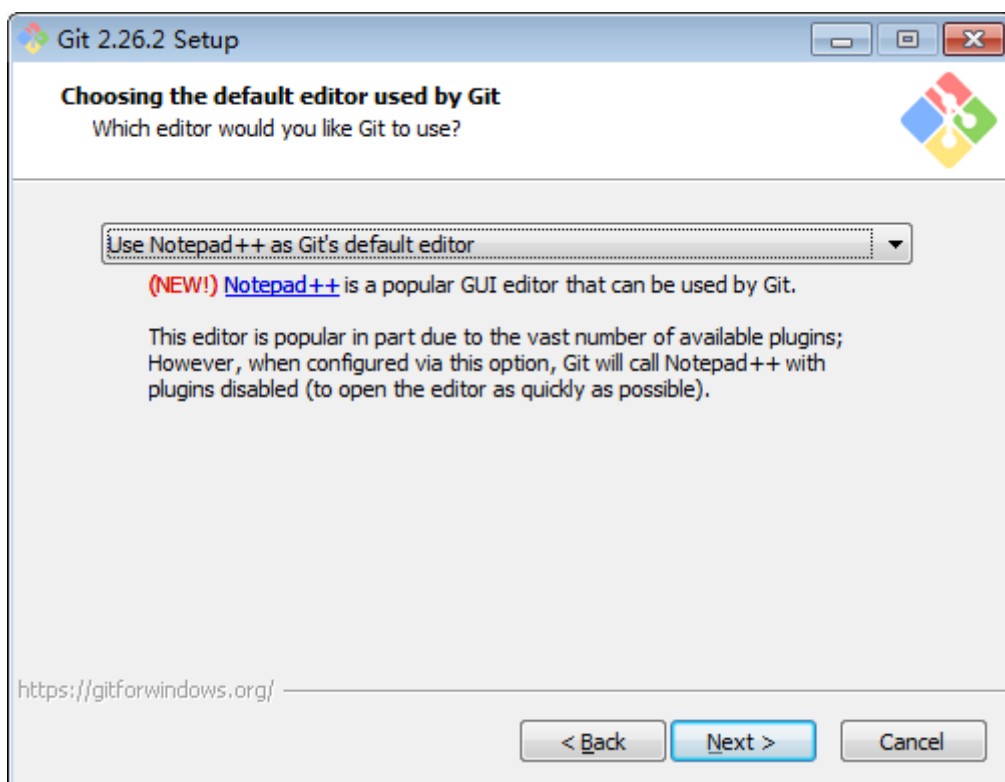
### 3. 常用的命令

1. dir，显示当前目录下的内容
2. cd，跳转到指定的目录
  - 在路径中，.表示当前当前目录，..表示上一级目录
3. mkdir，创建一个新的文件夹
4. rmdir，删除一个文件夹

5. del，删除一个文件（慎用）

## 版本控制（Version Control）

- 版本控制是指对软件开发过程中各种代码，配置文件及说明文档等文件变更的管理。
- 版本控制有点类似于游戏的存储进度，通过版本控制工具可以存储代码的编写的进度，可以根据需要恢复代码到指定的状态。
- 版本控制工具有很多种，今天要讲的一种版本控制工具叫做Git
- git的安装：
  - 下载，双击安装
  - 注意：



- git的使用：
  1. 基本的配置（通常只需配置一次）
    - 配置用户名

```
git config --global user.name "用户名"
```

- 配置电子邮件

```
git config --global user.email "电子邮件"
```

### 2. 使用

- 初始化项目，将项目交给git维护
  1. 进入到项目目录
  2. 初始化项目，使git对代码进行管理

```
git init
```

- 查看代码的状态

```
git status
```

- 使git对文件进行追踪

```
git add 文件名
```

- 追踪当前目录下所有文件

```
git add .
```

- 将代码存储到仓库中（存储进度）

```
git commit
```

- 如果使用的是vim编辑器，输入内容，然后Esc，然后输入:wq回车
- 将代码存储到仓库中，并直接设置描述信息

```
git commit -m '设置描述信息'
```

- 提交所有的已被追踪的文件

```
git commit -a -m '描述的信息'
```

- 查看提交的日志

```
git log
```

- 加载指定版本的代码

```
git checkout 版本
```

- 远程的Git库

- 在真正的开发项目时，Git仓库一定不是仅仅存在于本地计算机，一定会有一个公共的服务器用来存储代码，公共服务器可以满足多人开发的需求。
- 在公司时，一般公司都会有自己的Git的服务器，用来存储代码。但是我们现在处于学习阶段，所以我们可以直接使用一些公共的Git仓库来学习远程Git的操作
- 这几天我们使用国内的代码托管平台，来学习git的操作 <https://gitee.com/>
-

仓库名称

class200422

归属

ymhold

/

路径

class200422

仓库地址：<https://gitee.com/ymhold/class200422>

仓库介绍 非必填

class200422的JS代码

是否开源

☐ 私有 ☒ 公开

任何人都可以访问该仓库的代码和其他任何形式的资源

选择语言

JavaScript

添加 .gitignore

JetBrains

添加开源许可证

点此快速选择许可证

请选择开源许可证

○ 远程库的操作

- 复制远程的仓库：

```
git clone 仓库的地址
```

- 向远程仓库推送代码（上传）
  - 向服务器推送代码前，先用拉取一下服务器中的最新代码

```
git push
```

- 从远程仓库中拉取最新代码（下载）

```
git pull
```

- 记住用户的权限信息

```
git config --global credential.helper wincred
```

- 取消记录用户的权限信息

```
git credential-manager uninstall
```