

git 初学者

以下是将您学习的 Git 内容转化为 Markdown 格式的初学者笔记：



程式與網頁開發者必備技能！Git 和 GitHub 零基礎...

◆ 內容綱要00:00 開場白 & Git 安裝02:03 Git 基本設定與初始化
03:36 Git 檔案狀態06:46 檢視提交記錄與檔案還原10:06 忽略檔案
清單10:34 GitHub 註冊 & 同步儲存庫12:11 加入協作者至專案

www.youtube.com



十分钟学会正确的github工作流，和开源作者们使...

这期视频分享一下github工作流，也就是究竟应该如何使用github来维护代码。

www.youtube.com

Git 学习笔记

什么是 Git?

- Git 是一个版本控制工具，用于管理文件的版本。
- 类似于 Google 文档或 Word 的历史记录功能，但更加强大。
- 主要用于代码管理和追踪，但也适合非程序员使用，例如网页设计师、数据分析师或学生。

Git 的功能与好处

- 版本控制：记录文件的修改历史。
- 多人协作：支持多人同时参与项目开发。
- 数据备份：安全地保存项目进度和历史。

安装前准备

- 推荐安装 [VS Code](#)，一款免费开源的编辑器。
- 安装完成后，选择一个用于存放项目的文件夹，并在 VS Code 中打开。

安装 Git

Windows 系统

1. 下载 [Git 安装包](#)。
2. 双击安装文件，按照提示点击“下一步”完成安装。

macOS 系统

1. 按下 `Command + 空白键` 调出 Spotlight，输入 `Terminal` 打开终端。
2. 输入以下指令安装：

```
xcode-select --install
```

3. 按照提示点击“安装”并等待完成。

检查安装是否成功

1. 打开 VS Code 的终端（上方菜单 -> 终端 -> 新增终端）。
2. 输入以下指令：

```
```bash
git --version
```
```

3. 若输出 Git 的版本号，则表示安装成功。

Git 基本配置

设置用户名与邮箱

Git 需要记录每次变更的作者信息：

```
```bash
git config --global user.name "你的姓名"
git config --global user.email "你的邮箱地址"
```

---

## 初始化 Git 仓库

1. 打开 VS Code 的终端。
2. 在项目文件夹内输入以下指令：

```
git init
```

3. 此指令会在当前文件夹生成一个隐藏的 `.git` 文件夹，用于存储版本记录。

---

## Git 文件状态

### 文件的几种状态

1. **未追踪 (Untracked)** :
  - Git 不记录文件的历史。
2. **已暂存 (Staged)** :
  - 文件已被 Git 标记为需要记录。
3. **已提交 (Committed)** :
  - 文件的状态被永久存储。

---

## Git 常用指令

### 查看文件状态

```
git status
```

### 添加文件到暂存区

```
git add 文件名
或添加所有文件
git add .
```

### 提交文件

```
git commit -m "提交说明"
```

### 查看提交历史

```
git log
简化为单行显示
git log --oneline
```

## 清空终端屏幕

```
clear
```

---

## 文件变更与版本管理

### 修改文件

1. 修改文件内容并保存。
2. 使用以下指令更新版本：

```
git add .
git commit -m "修改说明"
```

### 查看文件差异

```
git diff
```

### 恢复到某个版本

```
git checkout 版本号 文件名
```

### 重置到某个版本（不可逆）

```
git reset --hard 版本号
```

---

## 远程仓库与 GitHub

### 推送本地代码到 GitHub

1. 注册 [GitHub 账号](#)。
2. 创建远程仓库（Repository）。
3. 按 GitHub 提供的指令操作：

```
git remote add origin 仓库地址
git branch -M main
git push -u origin main
```

## 拉取远程代码

```
git pull
```

---

## 团队协作与分支管理

### 创建分支

```
git checkout -b 分支名称
```

### 切换分支

```
git checkout 分支名称
```

### 合并分支

1. 提交 Pull Request 请求合并。
  2. 审核通过后点击 Merge Pull Request 合并到主分支。
- 

## 忽略文件

### 创建 .gitignore 文件

1. 在项目文件夹内新建 `.gitignore` 文件。
2. 添加需要忽略的文件或文件类型，例如：

```
*.log
node_modules/
```

---

### 小贴士

- 使用 `git help` 指令 查看具体指令的帮助文档。
- 在进行不可逆操作前，务必备份重要数据。

---

希望这份笔记能帮助你轻松上手 Git! 🎉

保存为 ``git_notes.md`` 即可! 有什么需要调整的可以随时告诉我。 😊