**《Git和GitHub简介》学习笔记**

这份文件主要讲述了Git的基本概念、基本操作命令、工作流程、配置方法、分支管理，以及如何使用GitHub进行远程仓库操作、项目贡献和团队协作。

它提供了Git安装、配置用户信息、文件状态管理、常用命令等相关知识的简单介绍。

通过阅读这个文件，我简单地了解到GitHub的主要功能以及我如何去使用它。

以下是文件内的一些主要内容的记录：

Git：由Linus Torvalds于2005年开发，是一个分布式版本控制系统。

GitHub：基于Git的代码托管平台，用于项目代码的存储和协作。

**1.Git基本操作**：

基本命令：git clone、git pull、git commit&git push。

文件状态：committed（已提交）、modified（已修改）、staged（已暂存）。

**2.Git工作流程：**

修改文件：在工作目录中进行。

暂存文件：使用git add命令。

提交文件：使用git commit -m "说明"。

推送更新：使用git push。

**3.初次运行Git前的配置：**

配置用户信息：git config --global user.name和user.email。

设置文本编辑器和差异分析工具。

**4.Git常用命令：**

创建仓库：git init或git clone [url]。

检查文件状态：git status。

提交文件：git add和git commit。

移除文件：git rm。

比较差异：git diff。

移动或重命名文件：git mv。

查看提交历史：git log。

回退版本：git reset --hard 版本号。

**5.远程库操作：**

创建SSH key并添加到GitHub。

添加远程库：git remote add origin。

推送到远程仓库：git push -u origin master。

**6.Git分支管理：**

创建分支：git branch和git checkout -b。

切换分支：git checkout。

合并分支：git merge。

删除分支：git branch -d。

**7.冲突管理和远程分支：**

解决冲突：手动编辑冲突文件后再次提交。

推送远程分支：git push origin 分支名称。

与远程分支关联：git branch --set-upstream-to。

忽略特殊文件

创建.gitignore文件，添加需要忽略的文件类型。

**8.GitHub基本概念：**

Repository：项目代码仓库。

Star：收藏项目。

Fork：复制项目。

Pull Request：发起合并请求。

Watch：关注项目更新。

Issue：提出问题或讨论。

gist：粘贴代码数据。

**9.开源项目贡献流程：**

提交Issue：提出问题或建议。

Pull Request：提交代码修改请求。

**10.团队合作开发：**

使用Collaborators进行协作。

Fork & Pull Request：复制项目、提交修改、发起合并请求。

一开始刚注册GitHub的时候，我对里面的一些操作都不是很理解，前期使用的时候只能自己摸索。通过学习这份文件，我对GitHub这个网站有了更具体的认识，我更加理解了GitHub这个网站的特点，跟我以前使用的一些团队协作的app（比如飞书）有区别的是，我认为他更加注重个人，我可以从很多方面完善个人主页，做项目的时候可以是个人项目，也可以是团队项目。此外，GitHub是针对开发者的协作平台，而飞书是适用于任何团队的。

总的来说使用GitHub于我而言是一种很新奇的体验。希望在接下来的学习可以继续深入使用它，也为我以后的学习与工作做好积累。