

以下为 deep seek 总结的常见Gitbash命令行汇总

Git Bash 常用命令行手册

一、基础操作

初始化仓库

```
git init
```

查看状态

```
git status
```

添加文件到暂存区

```
git add 文件名          # 添加单个文件
```

```
git add .                # 添加所有修改的文件
```

提交更改

```
git commit -m "提交说明"
```

查看提交历史

```
git log                  # 详细日志
```

```
git log --oneline        # 简洁版日志
```

二、分支管理

创建分支

```
git branch 分支名
```

切换分支

```
git checkout 分支名      # 切换到已有分支
```

```
git checkout -b 新分支名 # 创建并切换到新分支
```

合并分支

```
git merge 分支名         # 将指定分支合并到当前分支
```

删除分支

```
git branch -d 分支名     # 删除已合并的分支
```

```
git branch -D 分支名     # 强制删除未合并的分支
```

查看所有分支

```
git branch               # 本地分支
```

```
git branch -a            # 包含远程分支
```

三、远程仓库操作

克隆远程仓库

```
git clone 仓库URL
```

关联远程仓库

```
git remote add origin 仓库URL # 首次关联  
git remote -v # 查看关联的远程仓库
```

推送到远程仓库

```
git push origin 分支名 # 推送到指定分支  
git push -u origin 分支名 # 首次推送并设置默认上游分支
```

拉取远程更新

```
git pull origin 分支名 # 拉取并合并远程分支  
git fetch # 仅获取远程更新（不自动合并）
```

解决冲突流程

```
git pull # 先拉取最新代码  
# 手动解决冲突文件中的 <<<<<< 标记  
git add .  
git commit -m "解决冲突"  
git push
```

四、实用技巧

撤销修改

```
git checkout -- 文件名 # 撤销工作区的修改（未 add）  
git reset HEAD 文件名 # 撤销暂存区的修改（已 add）  
git reset --hard commitID # 回退到指定提交（慎用！）
```

暂存当前工作

```
git stash # 临时保存未提交的修改  
git stash pop # 恢复暂存的修改
```

查看文件差异

```
git diff # 工作区与暂存区的差异  
git diff --cached # 暂存区与仓库的差异
```

重命名/删除文件

```
git mv 旧文件名 新文件名 # 重命名文件（Git 会跟踪）  
git rm 文件名 # 删除文件（Git 会跟踪）
```

忽略文件配置

1. 创建 .gitignore 文件
2. 添加要忽略的文件/文件夹（如 node_modules/, *.log）

五、常见问题处理

修改最后一次提交信息

```
git commit --amend -m "新提交信息"
```

恢复误删分支

```
git reflog                                # 查找commitID
git checkout -b 分支名 commitID          # 恢复分支
```

强制回退

```
git reset --hard HEAD                    # 放弃所有未提交的修改
```

命令速查表

场景	命令
初始化仓库	<code>git init</code>
提交代码	<code>git add . + git commit -m "说明"</code>
推送到远程	<code>git push origin 分支名</code>
拉取更新	<code>git pull origin 分支名</code>
切换分支	<code>git checkout 分支名</code>
合并分支	<code>git merge 分支名</code>
撤销工作区修改	<code>git checkout -- 文件名</code>