**Git基本操作（一）**

**1、基本操作练习**

**（1）初始化本地版本库**

要对现有的某个项目开始使用Git管理，只需到此项目所在的根目录，执行git init命令即可。

准备一个新的Git本地版本库：

|  |
| --- |
| Bash L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit $ git init Initialized empty Git repository in J:/git-repository/learngit/.git/ |

说明：初始化Git仓库后，在当前目录下会出现一个名为.git的目录，所有Git需要的数据和资源都存放在这个目录中。

不过目前，仅仅是按照既有的结构框架，初始化好了Git仓库中所有的文件和目录，但我们还没有开始跟踪管理项目中的任何一个文件。

**（2）查看文件的状态**

使用git status命令进行查看。

进入本地版本库查看工作区、暂存区中文件的状态。

|  |
| --- |
| Bash # 执行`git status`命令 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status # 在主分支上工作 On branch master   # 尚无提交文件，指的是本地库中没有提交过任何文件。 No commits yet   # 无需提交（可创建/复制文件并使用“git add”进行跟踪） # 无需提交指的是，暂存区中没有任何可提交的文件 # 追踪文件，就是让Git管理该文件。 nothing to commit (create/copy files and use "git add" to track) |

**（3）把文件添加到暂存区**

**1）创建文件后查看工作区、暂存区中文件的状态**

我们在仓库目录中创建一个readme.txt文件后，在执行git status命令。

|  |
| --- |
| Bash # 创建readme.txt文件 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ touch readme.txt  L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ ll total 0 -rw-r--r-- 1 L 197121 0 4月 4 00:38 readme.txt  # 查看工作区、暂存区状态 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status On branch master  # 尚无提交文件，指的是本地库中没有提交过任何文件。 No commits yet  Untracked files:  (use "git add <file>..." to include in what will be committed)  readme.txt # 文件名为红色  nothing added to commit but untracked files present (use "git add" to track) |

说明：

1. Untracked files:readme.txt 表示发现未追踪的文件readme.txt
2. use "git add <file>..." to include in what will be committed 表示对readme.txt文件，你可以使用git add <file>命令， 将新建文件添加到暂存区。
3. nothing added to commit but untracked files present (use "git add" to track) 表示你没有添加任何内容到暂存区，但是存在未追踪的文件， 可使用“git add”进行追踪。

**2）将工作区的文件添加到暂存区**

执行git add命令，将readme.txt文件添加到暂存区。

|  |
| --- |
| Bash L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git add readme.txt warning: LF will be replaced by CRLF in readme.txt. The file will have its original line endings in your working directory |

|  |
| --- |
| 说明：  warning: LF will be replaced by CRLF in readme.txt.  The file will have its original line endings in your working directory   * 警告：readme.txt文件中的换行符格式会被替换，CRLF替换LF。 * 在我们安装Git的第8步时，所选择的选项。这是Git底层的东西，不用我们控制的。 * 但是该文件在你的工作目录中，还是保存原始文件的换行符格式。 * LF是Linux系统下的换行符，而CRLF是windows系统下的换行符。由于我们的文件创建于Linux系统下（Git Bash中创建），而保存中windows系统中，所以文件中的行结束符要使用windows下的CRLF格式换行。 * 以上两行就是一个提示作用，对我们的操作没有实际影响。 |

**3）再次查看工作区、暂存区状态**

执行git status命令查看工作区、暂存区状态。

|  |
| --- |
| Bash L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status On branch master  No commits yet  Changes to be committed:  (use "git rm --cached <file>..." to unstage)  new file: readme.txt # 文件名为绿色 |

说明：

1. Changes to be committed: new file: readme.txt
2. 所做更改：新建了readme.txt文件
3. use "git rm --cached <file>..." to unstage
4. 提示你可以适用使“git rm --cached <file> ...”命令，
5. 把文件从暂存区中撤回到工作区。

总结：

只要在"Changes to be committed"这行下面显示的文件，就说明是已暂存状态。

如果此时提交，那么该文件此时此刻的版本，将被留存在历史记录中。

在git add命令后面可以指明要跟踪的文件或目录路径。

如果是目录的话，就说明要递归跟踪该目录下的所有文件。（其实git add命令的潜台词就是把目标文件快照放入暂存区域，同时未曾跟踪过的文件标记为已跟踪。）

**4）将文件从暂存区撤回到工作区**

执行git rm --cached命令，将readme.txt文件从暂存区撤回到工作区。

并执行git status命令查看工作区、暂存区状态。

|  |
| --- |
| Bash L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git rm --cached readme.txt rm 'readme.txt'  L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status On branch master  No commits yet  Untracked files:  (use "git add <file>..." to include in what will be committed)  readme.txt # 文件名为红色  nothing added to commit but untracked files present (use "git add" to track) |

可以看到结果，readme.txt文件又成为了一个未被Git追踪的文件。

作者：繁华似锦Fighting 链接：https://www.jianshu.com/p/455135b4b8f6 来源：简书 著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。