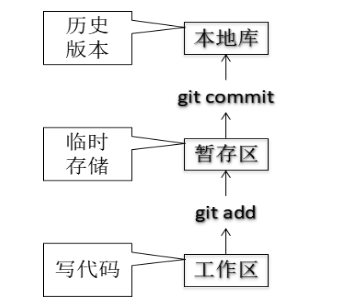
**Git操作 - Git基本操作补充**

**1、Git本地版本库结构**



1. 工作区(Working Directory) 添加、编辑、修改、删除文件等操作。
2. 暂存区(Stage) 打算提交，但还没提交的内容。最后可以统一提交到Git仓库中。也可以不提交，撤销回来。
3. Git仓库(Git Repository)
4. 实实在在的项目存储的每个一历史的版本。

**2、Git常用操作方法**

Git的专属命令都是以git开始的，然后是索要执行的操作，最后还可以加上一些参数。

以下命令都需在仓库中执行。

**（1）状态查看**

命令：git status

命令作用：查看工作区、暂存区状态。

**（2）添加文件到暂存区**

命令：git add [file name]

命令作用：

* 该命令的作用是告诉Git系统，将指定文件的当前快照写入到版本库暂存区。即，将文件交给Git进行版本管理。
* 提交到暂存区，并且转换文件中的换行符。
* 被Git追踪的暂存区中的文件可以被提交到本地版本库。

**（3）文件从暂存区撤回到工作区**

命令：git rm --cached [file name]

命令作用：把文件从暂存区撤回到工作区。

**（4）提交文件**

命令：git commit -m '本次提交的说明'

命令作用：提交操作就通过命令将Git暂存区中的文件快照永久性地写入到本地版本库中。

**3、补充：添加多个文件到暂存区**

有两种方式：

* git add命令后添加多个文件，文件之间使用空格分隔。
* git add命令后使用通配符\*指定多个文件。

示例：

|  |
| --- |
| Bash # 1.查看工作区、暂存区状态 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status On branch master Untracked files:  (use "git add <file>..." to include in what will be committed)  hello.java # 文件名为红色  test.java # 文件名为红色  xyj-sha\_hs.py # 文件名为红色  xyj-sun\_wk.py # 文件名为红色  xyj-zhu\_bj.py # 文件名为红色  nothing added to commit but untracked files present (use "git add" to track)  # 2.添加多个文件到暂存区 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git add hello.java test.java  L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status On branch master Changes to be committed:  (use "git restore --staged <file>..." to unstage)  new file: hello.java # 文件名为绿色  new file: test.java # 文件名为绿色  Untracked files:  (use "git add <file>..." to include in what will be committed)  xyj-sha\_hs.py # 文件名为红色  xyj-sun\_wk.py # 文件名为红色  xyj-zhu\_bj.py # 文件名为红色  # 3.使用通配符添加多个文件到暂存区 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git add xyj\*.py  L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status On branch master Changes to be committed:  (use "git restore --staged <file>..." to unstage)  new file: hello.java # 文件名为绿色  new file: test.java # 文件名为绿色  new file: xyj-sha\_hs.py # 文件名为绿色  new file: xyj-sun\_wk.py # 文件名为绿色  new file: xyj-zhu\_bj.py # 文件名为绿色 |

**4、补充：提交未写备注**

如果你在执行git conmit提交命令的时候，并没有写-m信息，这时会启动文本编辑器，以便输入本次提交的说明。

默认的提交消息包含最后一次运行git status的输出，放在注释行里，如下图：

另外开头还有一空行，供你输入提交说明。你完全可以去掉这些注释行，不过留着也没关系，多少能帮你回想起这次更新的内容有哪些。

如下：

会弹出一个窗口，一个vim编辑器窗口：

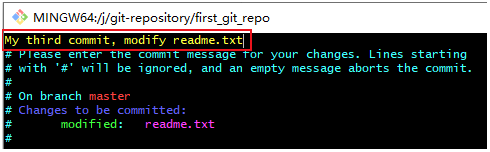


说明：

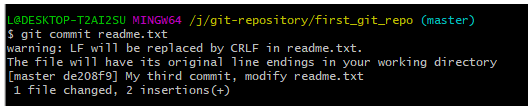
|  |
| --- |
| Bash # Please enter the commit message for your changes. Lines starting # with '#' will be ignored, and an empty message aborts the commit. # # On branch master # Changes to be committed: # modified: readme.txt # |

1. Please enter the commit message for your changes. Lines starting with '#' will be ignored, and an empty message aborts the commit. 对于你这次提交中修改的内容要进行说明， 以'＃'开头的行将被忽略，并且在第一行填写说明信息。
2. Changes to be committed: modified: readme.txt readme.txt的修改已被Git追踪到

进行提交说明的补充：



保存并退出后，提交成功，如下图：



|  |
| --- |
| 总结：当提交操作的说明内容比较多，或者需要写的比较详细的时候，可以使用这种方式提交。 |

**5、从工作区直接提交到版本库**

尽管使用暂存区域的方式，可以准备好要提交的细节，但有时候这么做略显繁琐。

Git提供了一个跳过使用暂存区域的方式，只要在提交的时候，给git commit命令加上-a选项，Git就会自动把所有已经跟踪过的文件，暂存起来一并提交，从而跳过git add步骤。

命令：git commit -a

示例：

**1）先查看当前工作目录中的文件状态**

|  |
| --- |
| Bash L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status On branch master nothing to commit, working tree clean |

可以看到非常的干净。

**2）我们创建两个文件**

一个文件是新增文件test.txt，此前没有被Git追踪过。

两个文件是readme.txt，已被Git追踪，我们将该文件变成已修改状态。

|  |
| --- |
| Bash # 1.新建test.txt L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ echo "hello test" > test.txt  # 2.修改readme.txt文件 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ echo "hello git v666" >> readme.txt  # 3.查看工作目录中的文件状态 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status On branch master Changes not staged for commit:  (use "git add <file>..." to update what will be committed)  (use "git restore <file>..." to discard changes in working directory)  modified: readme.txt # 已修改状态，未在暂存区  Untracked files:  (use "git add <file>..." to include in what will be committed)  test.txt # 未追踪状态  no changes added to commit (use "git add" and/or "git commit -a") |

**3）提交到Git版本库中**

我们直接使用git commit -a -m命令直接执行提交操作，看看会发生什么情况。

|  |
| --- |
| Bash # 提交 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git commit -a -m 'test -a' [master 1b0de31] test -a  1 file changed, 1 insertion(+)  # 查看工作目录文件状态 L@DESKTOP-T2AI2SU MINGW64 /j/git-repository/learngit (master) $ git status On branch master Untracked files:  (use "git add <file>..." to include in what will be committed)  test.txt # 未追踪状态  nothing added to commit but untracked files present (use "git add" to track) |

我们可以看到，未被追踪状态的文件，不能直接从工作区直接提交到版本库，使用-a选择也不可以。

作者：繁华似锦Fighting 链接：https://www.jianshu.com/p/df914459c6a0 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。