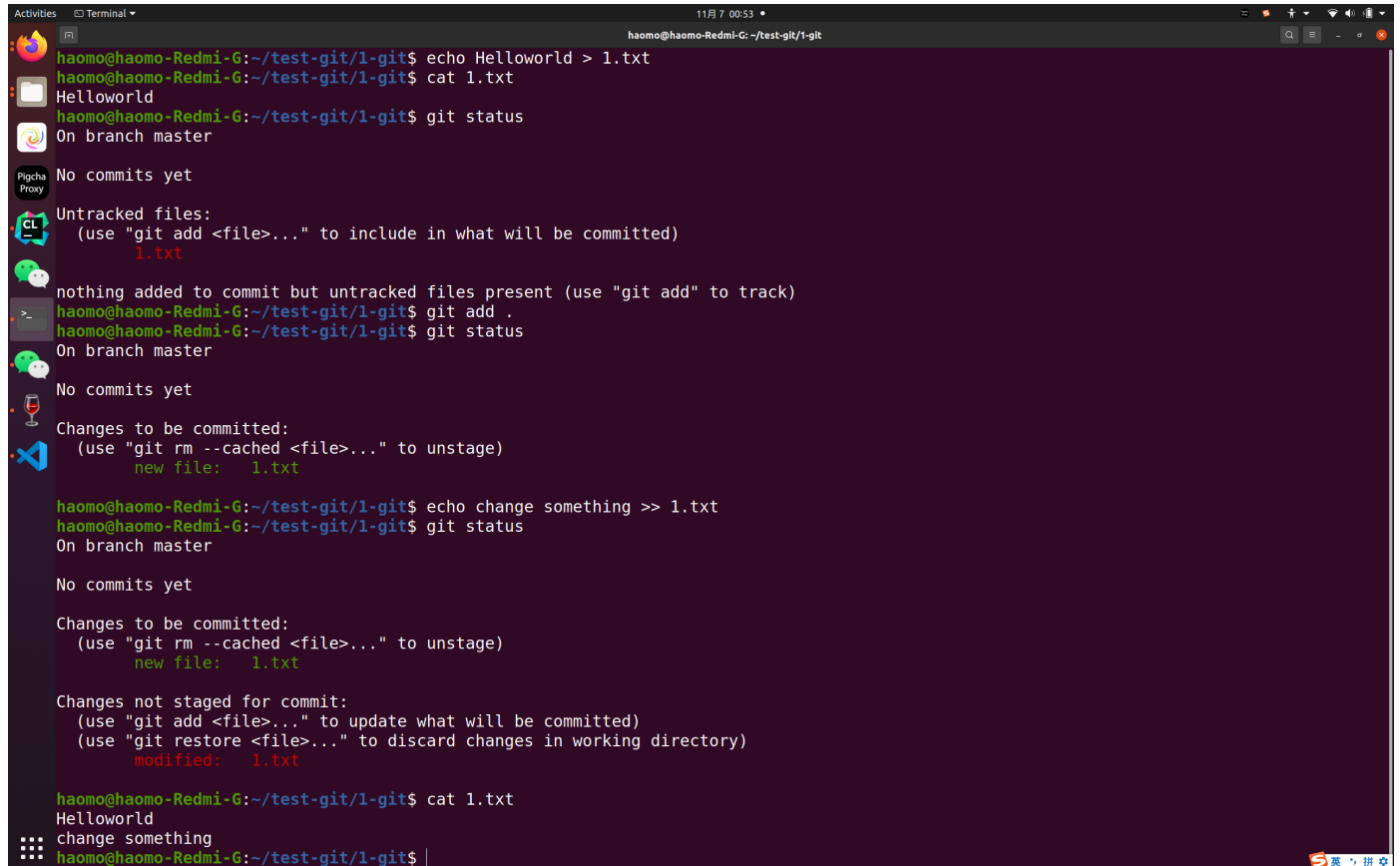


Git 应用

1. 关于撤销暂存区中的修改

- ①使用git reset将文件移出暂存区，此时工作区还保留修改
- ②使用git restore --staged也是将文件移出暂存区，此时工作区还保留修改
- ③使用图形化git操作（本质与命令行相同）

eg:



```
haomo@haomo-Redmi-6: ~/test-git/1-git$ echo Helloworld > 1.txt
haomo@haomo-Redmi-6:~/test-git/1-git$ cat 1.txt
Helloworld
haomo@haomo-Redmi-6:~/test-git/1-git$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        1.txt

nothing added to commit but untracked files present (use "git add" to track)
haomo@haomo-Redmi-6:~/test-git/1-git$ git add .
haomo@haomo-Redmi-6:~/test-git/1-git$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   1.txt

haomo@haomo-Redmi-6:~/test-git/1-git$ echo change something >> 1.txt
haomo@haomo-Redmi-6:~/test-git/1-git$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   1.txt

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:   1.txt

haomo@haomo-Redmi-6:~/test-git/1-git$ cat 1.txt
Helloworld
change something
haomo@haomo-Redmi-6:~/test-git/1-git$
```

```
Activities Terminal 11月7 00:54
haomo@haomo-Redmi-G: ~/test-git/1-git
haomo@haomo-Redmi-G:~/test-git/1-git$ git reset 1.txt
haomo@haomo-Redmi-G:~/test-git/1-git$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    1.txt

nothing added to commit but untracked files present (use "git add" to track)
haomo@haomo-Redmi-G:~/test-git/1-git$ 此时已将文件从暂存区移回工作区，可进行修改
```

```
Activities Terminal 11月7 18:08
haomo@haomo-Redmi-G: ~/test-git/1-git
haomo@haomo-Redmi-G:~/test-git/1-git$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   3.txt

haomo@haomo-Redmi-G:~/test-git/1-git$ echo change >> 3.txt
haomo@haomo-Redmi-G:~/test-git/1-git$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   3.txt

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:   3.txt

haomo@haomo-Redmi-G:~/test-git/1-git$ git restore --staged 3.txt
haomo@haomo-Redmi-G:~/test-git/1-git$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    3.txt

nothing added to commit but untracked files present (use "git add" to track)
haomo@haomo-Redmi-G:~/test-git/1-git$ echo 3.txt > 3.txt
haomo@haomo-Redmi-G:~/test-git/1-git$ 此时文件已移出暂存区至工作区，可进行修改了
```

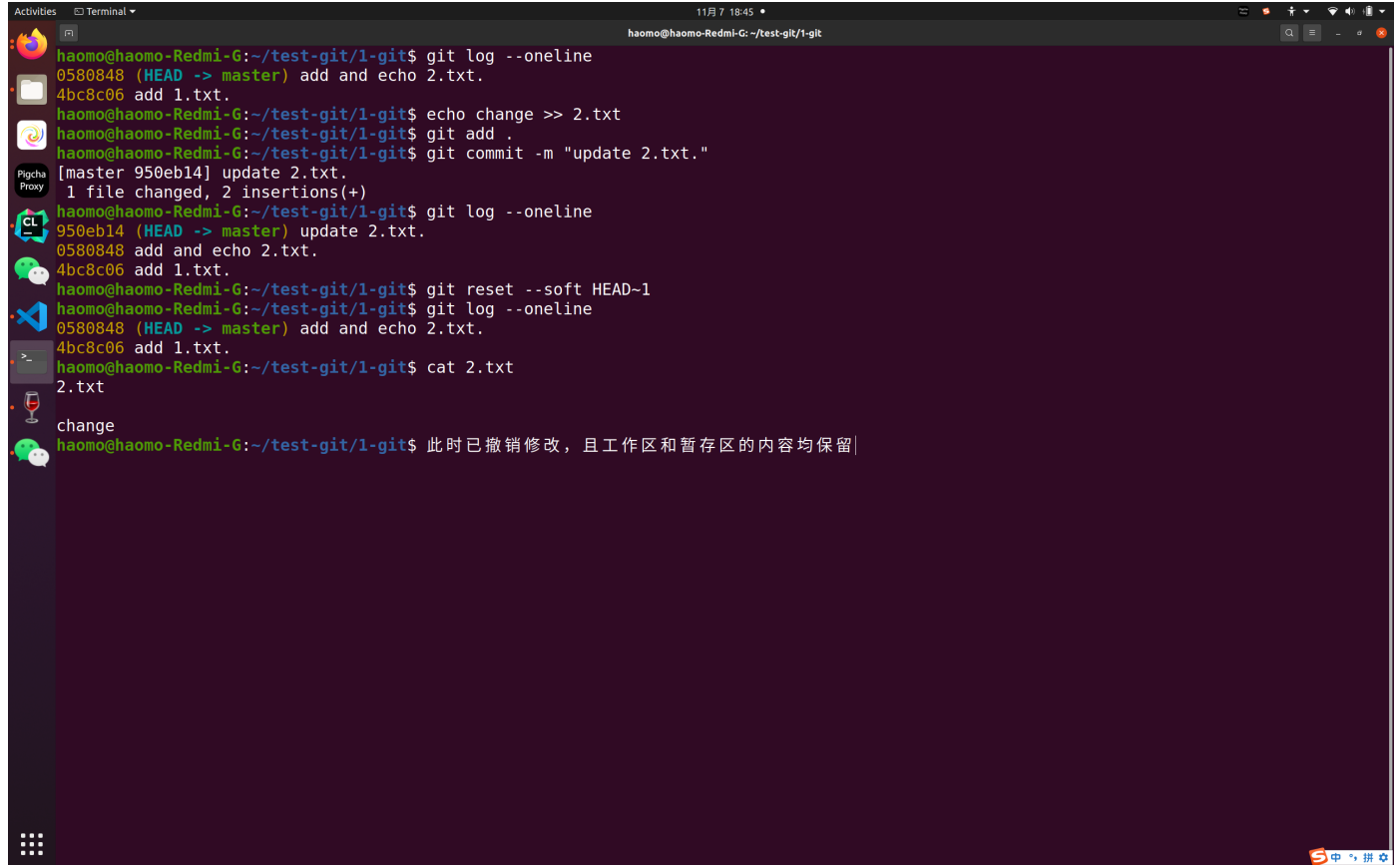
2. 关于撤销本地仓库中的修改

①不修改历史：使用git reset --soft HEAD~1来回退，此时工作区和暂存区中的修改都会保留（--mixed：只保留工作区；--hard：工作区暂存区都撤销）

Tips：如果使用reset，想推送到远程仓库需使用git push -f（因为少了个commit）

②修改历史：使用git revert HEAD~1新增一个commit，但这个commit是撤销修改（可理解为有一个修改change，git revert就是-change）

eg:

A terminal window on a Linux system showing a series of git commands and their outputs. The user is in a directory ~/test-git/1-git. The commands and outputs are as follows:

```
haomo@haomo-Redmi-G:~/test-git/1-git$ git log --oneline
0580848 (HEAD -> master) add and echo 2.txt.
4bc8c06 add 1.txt.
haomo@haomo-Redmi-G:~/test-git/1-git$ echo change >> 2.txt
haomo@haomo-Redmi-G:~/test-git/1-git$ git add .
haomo@haomo-Redmi-G:~/test-git/1-git$ git commit -m "update 2.txt."
[master 950eb14] update 2.txt.
1 file changed, 2 insertions(+)
haomo@haomo-Redmi-G:~/test-git/1-git$ git log --oneline
950eb14 (HEAD -> master) update 2.txt.
0580848 add and echo 2.txt.
4bc8c06 add 1.txt.
haomo@haomo-Redmi-G:~/test-git/1-git$ git reset --soft HEAD~1
haomo@haomo-Redmi-G:~/test-git/1-git$ git log --oneline
0580848 (HEAD -> master) add and echo 2.txt.
4bc8c06 add 1.txt.
haomo@haomo-Redmi-G:~/test-git/1-git$ cat 2.txt
2.txt
change
haomo@haomo-Redmi-G:~/test-git/1-git$ 此时已撤销修改，且工作区和暂存区的内容均保留|
```

```
haomo@haomo-Redmi-G: ~/test-git/1-git
haomo@haomo-Redmi-G:~/test-git/1-git$ git revert HEAD~1
Removing 2.txt
[master 3e82a2a] Revert "add 2.txt."
1 file changed, 0 insertions(+), 0 deletions(-)
delete mode 100644 2.txt
haomo@haomo-Redmi-G:~/test-git/1-git$ ls
1.txt 3.txt
haomo@haomo-Redmi-G:~/test-git/1-git$ git log
commit 3e82a2a29d4f80e3b6f6b4d60060120a43a5b430 (HEAD -> master)
Author: Haomogg <385371785@qq.com>
Date: Fri Nov 10 19:52:49 2023 +0800

    Revert "add 2.txt."

    This reverts commit b11414b7f86bcd5f0e51da8b7abba7583130838e.

commit 0bd4d91012dc06064abca40b2f3fc929ee395abb
Author: Haomogg <385371785@qq.com>
Date: Fri Nov 10 19:51:11 2023 +0800

    add 3.txt.

commit b11414b7f86bcd5f0e51da8b7abba7583130838e
Author: Haomogg <385371785@qq.com>
Date: Fri Nov 10 19:50:53 2023 +0800

    add 2.txt.

commit 4bc8c06eca971f66a6ec436a48085dd960becf4
Author: Haomogg <385371785@qq.com>
Date: Tue Nov 7 17:39:24 2023 +0800

    add 1.txt.
haomo@haomo-Redmi-G:~/test-git/1-git$ 此时已revert,增加了一个commit来撤销修改|
```

3. 关于合并分支的两种方式

①git merge：传统合并

②git rebase：将分支直接移动到主支上对应的节点（此节点为创立分支的祖先）

eg:

传统merge

```
Activities Terminal 11月 10 22:10
haomo@haomo-Redmi-G: ~/test-git/1-git
haomo@haomo-Redmi-G:~/test-git/1-git$ git branch dev
haomo@haomo-Redmi-G:~/test-git/1-git$ git checkout dev
Switched to branch 'dev'
haomo@haomo-Redmi-G:~/test-git/1-git$ ls
1.txt 2.txt 3.txt
haomo@haomo-Redmi-G:~/test-git/1-git$ echo dev1.txt > dev1.txt
haomo@haomo-Redmi-G:~/test-git/1-git$ git add .
haomo@haomo-Redmi-G:~/test-git/1-git$ git commit -m "add dev1.txt."
[dev a3994ca] add dev1.txt.
1 file changed, 1 insertion(+)
create mode 100644 dev1.txt
haomo@haomo-Redmi-G:~/test-git/1-git$ echo dev2.txt > dev2.txt
haomo@haomo-Redmi-G:~/test-git/1-git$ git add .
haomo@haomo-Redmi-G:~/test-git/1-git$ git commit -m "add dev2.txt."
[dev 4db4617] add dev2.txt.
1 file changed, 1 insertion(+)
create mode 100644 dev2.txt
haomo@haomo-Redmi-G:~/test-git/1-git$ git status
On branch dev
nothing to commit, working tree clean
haomo@haomo-Redmi-G:~/test-git/1-git$ git branch
* dev
master
haomo@haomo-Redmi-G:~/test-git/1-git$ git checkout master
Switched to branch 'master'
haomo@haomo-Redmi-G:~/test-git/1-git$ git merge dev
Updating 735ee74..4db4617
Fast-forward
 dev1.txt | 1 +
 dev2.txt | 1 +
 2 files changed, 2 insertions(+)
 create mode 100644 dev1.txt
 create mode 100644 dev2.txt
haomo@haomo-Redmi-G:~/test-git/1-git$ git log --oneline --graph --decorate --all
* 4db4617 (HEAD -> master, dev) add dev2.txt.
* a3994ca add dev1.txt.
* 735ee74 add 3.txt.
* ab9dfcc add 2.txt.
* 4bc8c06 add 1.txt.
```

```
Activities Terminal 11月 10 22:24
haomo@haomo-Redmi-G: ~/test-git/1-git
haomo@haomo-Redmi-G:~/test-git/1-git$ graph
* fabac5f (HEAD -> dev2) add 333.txt.
* b2a4628 add 222.txt.
* 4db4617 (master, dev) add dev2.txt.
* a3994ca add dev1.txt.
* 735ee74 add 3.txt.
* ab9dfcc add 2.txt.
* 4bc8c06 add 1.txt.
haomo@haomo-Redmi-G:~/test-git/1-git$ git branch
dev
* dev2
master
haomo@haomo-Redmi-G:~/test-git/1-git$ git rebase master
Current branch dev2 is up to date.
haomo@haomo-Redmi-G:~/test-git/1-git$ graph
* fabac5f (HEAD -> dev2) add 333.txt.
* b2a4628 add 222.txt.
* 4db4617 (master, dev) add dev2.txt.
* a3994ca add dev1.txt.
* 735ee74 add 3.txt.
* ab9dfcc add 2.txt.
* 4bc8c06 add 1.txt.
haomo@haomo-Redmi-G:~/test-git/1-git$ 此时已将分支移到对应主支上|
```