

۱- در صورتی که فایل هایی را به اشتباه استیج کنیم که مربوط به آن کامیت نیستند میتوان از دستور زیر برای خارج کردن آن فایل از استیج استفاده کرد:

نام فایل `git reset`

با استفاده از دستور فوق میتوانیم فایل تغییر یافته را از استیج به دایرکتوری کاری انتقال دهیم.

۲- در صورتی که کامنت کامیت انجام شده در ریپوزیتوری لوکال دارای مشکل باشد با استفاده از دستور زیر میتوان ویرایش را انجام داد :

`git commit --amend`

در صورتیکه مایل به ویرایش در ادیتور ترمینال نباشیم میتوانیم از دستور زیر هم استفاده کنیم :

`git commit --amend -m "updated commit message"`

۳- اگر بخواهیم علاوه بر تغییر دادن پیام کامیت فایل هایی که در کامیت قرار داشتند در لوکال رو بگردانیم باید دو حالت رو در نظر بگیریم :

اگر کامیت اشتباه اولین کامیت باشد از دستور زیر استفاده میکنیم :

`git update-ref -d HEAD`

با این دستور اولین کامیت برگشت داده میشه و فایل های درون اون به حالت استیج درمی آیند و میتوانیم اونها رو مجددا ویرایش کرده و دوباره کامیت کرد.

اگر کامیت مورد نظر کامیت اول نباشه از ستور زیر استفاده میکنیم :

`git reset HEAD~`

زمانی که از این دستور استفاده میکنیم تغییراتی که در فایل های مورد نظر دادیم به `working tree` برگشت داده میشه و به حالت `unstage` در میان و تغییراتی که از قبل بر روی فایل ها در `working tree` انجام داده بودیم سر جاش باقی میمونه و با دستور `git log` میبینیم که کامیت مورد نظر دیگه وجود نداره.

اگر کامیت رو پوش کرده باشیم از دستور زیر استفاده میکنیم :

`git push remote <branch> --force`

۴- در صورتی که مرج دارای مشکل بود با دستور زیر میتوانیم اون رو بگردانیم:

`git reset --hard <commit-before-merge>`

در صورتی که هش کامیت قبل از مرج رو نداشتیم میتوانیم از دستور زیر استفاده کنیم:

`git reset --hard HEAD~1`

نکته ای که وجود دارد اینه که با این دستورات اگه تغییرات کامیت نشده ای داشته باشیم از بین میره.

برای برگرداندن مرج در ریموت از دستور زیر استفاده میکنیم:

`git revert -m 1 <merge-commit-hash>`

۵- در صورتی که به `merge conflict` برخورد کردیم میتوانیم به فایلی که قصد مرج داشتیم رفته و

یکی از کدها رو انتخاب کنیم و دیگری رو به صورت دستی حذف کنیم.

در صورتی که از ویرایش گر `Visual studio code` استفاده میکنیم میتوانیم از گزینه هایی مثل

`Accept current change`:

با این گزینه تغییری که در شاخه مستر یا فعلی هست انتخاب میشه

`Accept incoming change` :

با این گزینه تغییری که در شاخه بعدی هست انتخاب میشه

`Accept both changes`:

با انتخاب این گزینه هر دو تغییر به صورت همزمان نگه داشته میشن

۶- میتوانیم از دستور `Git checkout` استفاده کنیم و در انتها هش کامیتی که میخواهیم به اون

برگردیم رو وارد کنیم.

دستور `Git revert` که در انتهای اون باید هش کامیت رو بنویسیم برای زمان هایی که به آخرین

ورژن برگردیم مناسب تر هست چون زمان هایی که به کامیت های چند ورژن قبل برگردیم امکان

ایجاد `conflict` وجود دارد.