1.Как отказаться от изменений в файле (файлах) находящийся в рабочем каталоге (work directory)?

Нужно использовать команду **git checkout -- <file>** для каждого файла.

**//////////////////////////////////////////////**

2.Как вернуть файл из индекса в рабочий каталог?

**git reset HEAD <file>**

**//////////////////////////////////////////////**

3.Как отменить ошибочный commit ?

**git revert <commit>** - Это создаст новый коммит, который отменяет изменения, внесенные в указанном коммите.

**//////////////////////////////////////////////**

4.Как исправить коммит отправленный на сервер GitHub?

**git commit --amend** - Позволит внести изменения в последний коммит

**//////////////////////////////////////////////**

5.Как создать новую ветку и переключится в нее?

**git checkout -b <branchname>**

**//////////////////////////////////////////////**

6.В чем отличение между fast-forward merge and non fast-forfard merge?

**Fast-forward merge** и **non fast-forward merge** - это два различных способа объединения веток в Git

**//////////////////////////////////////////////**

7.Когда может возникнуть конфликт слияния веток, как его разрешить?

*Конфликт слияния веток может возникнуть, когда две ветки, которые вы пытаетесь объединить, внесли различные изменения в один и тот же участок кода.*

*Это может произойти, например, когда два разработчика работают над одним и тем же файлом и пытаются объединить свои изменения.*

*Чтобы разрешить конфликт слияния веток, вы можете следовать этим шагам:*

*Запустите команду git status для проверки наличия конфликта слияния и определения файлов, в которых он возник.*

*Откройте конфликтные файлы в редакторе кода. В этих файлах будут помечены участки кода, которые были изменены в каждой из веток.*

*Проанализируйте изменения в конфликтных участках кода и определите, какие изменения следует сохранить. Вы можете редактировать файлы,*

*чтобы объединить изменения или выбрать, какие изменения следует сохранить.*

*Удалите маркеры конфликтов (<<<<<<<, ======= и >>>>>>>) и сохраните изменения.*

*Запустите команду git add <file> для каждого конфликтного файла, чтобы пометить его как разрешенный конфликт.*

**Запустите команду git commit для завершения слияния веток после разрешения всех конфликтов. При этом будет создан новый коммит, объединяющий изменения из обеих веток.**

**//////////////////////////////////////////////**

8.Можно ли удалить ветку? Как?

**git branch -d <branchname>**  -Это удалит указанную ветку, если она уже слита с текущей веткой.

**//////////////////////////////////////////////**