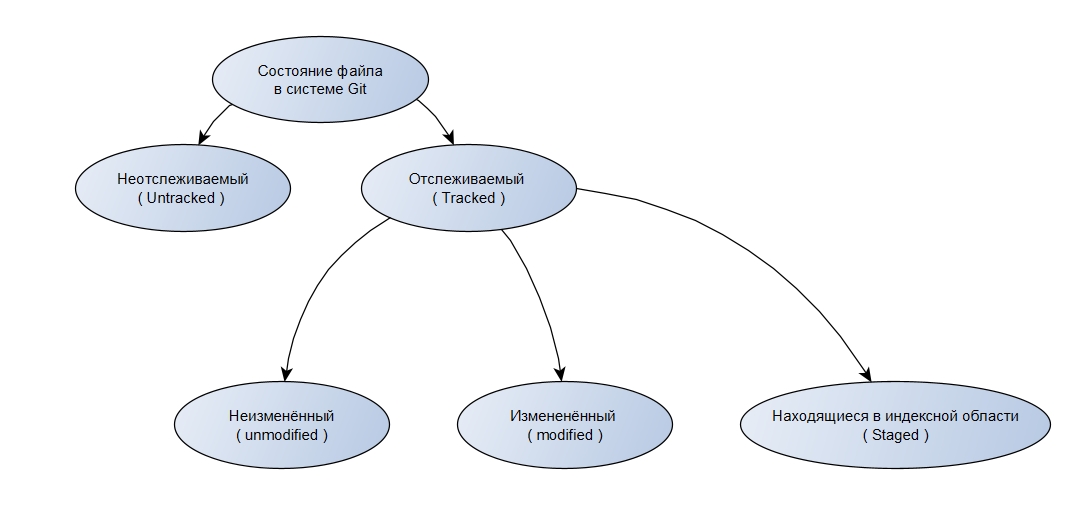
Git init – создание нового репозитория ( создаётся подпапка .git )

Git add – добавить файлы под версионный контроль и переместить в индексовую область

Git commit – Закоммитить файлы из индексной области в область HEAD ( в репозиторий )

Git clone – Используется для получения уже существующего удаленного репозитория ( для клонирования )

В каких состояниях может находиться файл в Git

 Для определения в каком состоянии находитятся файлы используется команда git status

Для добавления файла под версионный контроль и записи в индексную область ( staged ) используется команда git add

**Игнорирование файлов**

Часто бывает необходимо скрыть группу файлов от версионного контроля, но так чтобы не наблюдать их в списки неотслеживаемых. Для этого используется команда git ignore для более подробного изучения как работать с ней используйте справочное руководство.

**Коммит изменений ( commit - фиксировать )**

Теперь когда ваш индексный файл содержит все нужные вам изменения, вы можете зафиксировать все изменения. Для коммита используется команда git commit, часто в качестве аргумента функции может использоваться параметр –m после которого следует название коммита. Следует отметить, в область HEAD ( репозиторий ) переходят только файлы находящиеся в индексной области.

Git log – просмотр истории коммитов.

**Операции отмены**

Допустим случилось так, что вы слишком рано закоммитили ваши файлы и остались небольшие правки которые необходимо внести в проект, для этого внестите в проект необходимые правки, проиндексируйте нужные файлы и заккомитьте необходимые файлы с параметром --amend

Пример: git commit --amend Данная команда дополнит предыдущий коммит и обьеденит результат ( результирующий коммит будет равен сумме двух последних коммитов ).

**Отмена индексации файла**

Допустим в индексную область попал файл по ошибке, и вы не хотите чтобы он попал в будущий коммит, для исключения файла из индексной области используется команда git reset HEAD <file\_name>