Настройка

* Указать глобальный username**:$ git config --global user.name "John Doe"**
* Указать глобальный user email**:$ git config --global user.email**[**johndoe@example.com**](mailto:johndoe@example.com)
* Указать редактор, который будет использоваться, когда нужно ввести сообщение в Git**:$ git config --global core.editor emacs**
* Указать утилиту сравнения, которая будет использоваться для разрешения конфликтов слияния**:$ git config --global merge.tool vimdiff**
* Просмотреть используемые настройки**:$ git config --list**

Работа с репозиторием

* Инициализировать репозиторий**:$ git init**
* Показать статус репозитория**:$ git status**
* Коммитит с указанием комментария**:$ git commit -m 'some entry script fixes'**
* Коммитит с указанием комментария и автоматической индексацией**:$git commit -a -m 'some entry script fixes'**
* Добавить файл в индекс**:$ git add index.php**
* Посмотреть изменения между рабочим каталогом и индексом**:$ git diff**
* Посмотреть изменения между последним коммитом и индексом**:$ git diff --cached**
* Удалить файл из индекса и из рабочего каталога**:$ git rm readme.txt**
* Удалить файл из индекса, оставив его при этом в рабочем каталоге**:$ git rm --cached readme.txt**
* Исключить файл (который находится в индексе, но еще не за коммитен) из индексации**:$ git reset HEAD readme.txt**
* Просмотр истории коммитов**:$ git log**
* Просмотр истории коммитов в GUI**:$ gitk**

Работа с удаленным сервером

* Просмотреть какие удалённые серверы уже настроены, параметр -v отображает так же url сервера:$ **git remote -v**
* Добавить новый удалённый репозиторий под именем pb**:$ git remote add pb** **git://github.com/paulboone/ticgit.git**
* Посмотреть список коммитов репозитория pb ветки master, которые были выполнены после последнего **pull-а:git log pb/master**
* Извлечь (fetch) всю информацию, которая есть в репозитории pb:$ **git fetch pb**
* Слить (merge) информацию, которую получили через fetch с рабочим каталогом:$ **git merge pb/master**
* Отправить (push) код в ветку master удаленного репозитория pb:$ **git push pb master**
* Переименовать удаленный репозиторий из pb на paul:$ **git remote rename pb paul**
* Удалить удаленный репозиторий paul:$ **git remote rm paul**

Работа с метками

* Просмотр имеющихся меток:$ **git tag**
* Создание аннотированной метки:$ **git tag -a v1.4 -m 'my version 1.4'**
* Создание легковесной метки:$ **git tag v1.4**
* Просмотр данных метки:$ **git show v1.4**
* Создание метки для какого-либо существующего коммита (9fceb02 - контрольная сумма коммита, или ее часть):$ **git tag -a v1.2 9fceb02**
* Отправить метку на удалённый сервер:$ **git push pb v1.5**

Работа с ветками

* Отобразить список веток проекта:$ **git branch**
* Отобразить список веток с последними коммитами:$ **git branch -v**
* Создать новую ветку с названием "iss53":$ **git branch iss53**
* Переключиться на ветку iss53**:$ git checkout iss53**
* Создать ветку с названием "iss53" и переключиться на нее:$ **git checkout -b iss53**
* Создать ветку с названием "articles" на основе ветки "articles" из удаленного сервера pb и переключиться на нее:**git checkout -b articles pb/articles**
* Слить текущую ветку с веткой hotfix:$ **git merge hotfix**
* Переименовать ветку "articles" в "content":$ **git branch -m articles content**
* Удалить ветку hotfix:$ **git branch -d hotfix**
* Удалить ветку hotfix на удаленном сервере pb:$ **git push pb :hotfix**
* Запустить графический инструмент для отображения конфликтных ситуаций:$ **git mergetool**
* Посмотреть список веток, которые уже слиты с текущей:$ **git branch --merged**
* Посмотреть список веток, которые содержат наработки, но еще не слиты с текущей:$ **git branch --no-merged**