

Linux and more. Git Recap.

Artem Trunov for Ozon Masters



На этом занятии

- Повторяем git (немножко)
- Загружаем deploy key
- Добавляем коллаборатора
- Домашнее задание

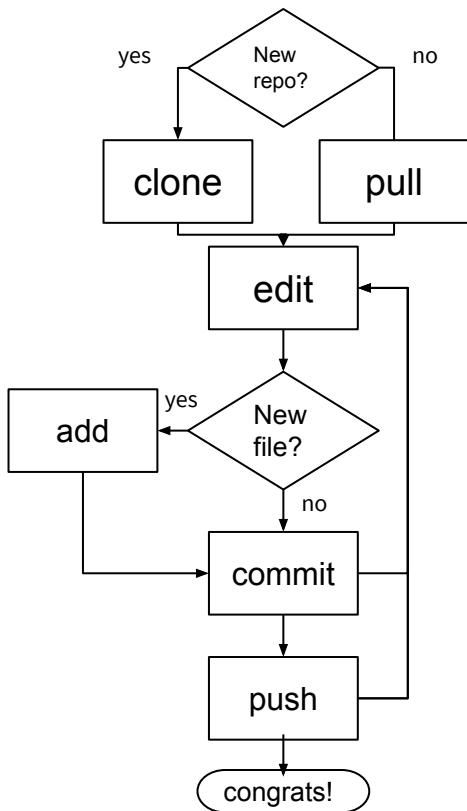
Git

- Распределенная система со слияниями изменений.

Терминология

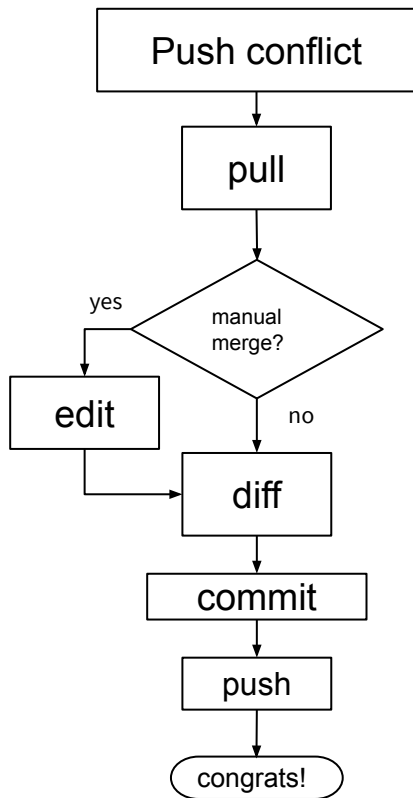
- Clone
 - Создание копии репозитория у себя
- Commit
 - Фиксация изменений в коде
- Tag, label, baseline
 - Слепок состояния репозитория
- Head
 - Слепок состояния с самым последний коммитом
- Branch
 - Ветка репозитория - копия файлов, над которой ведется независимая разработка. Изменения в ней позже могут быть слиты в “главную” ветку (trunk, master)
- Merge
 - Процесс слияния изменений в один файл/ветку
- Pull
 - Синхронизация изменений в другом репо с локальными (**push** - наоборот)

Алгоритм действий



- А на практике
 - Git pull перед
 - началом работы
 - Коммитом
 - пушем
 - git status перед каждым pull

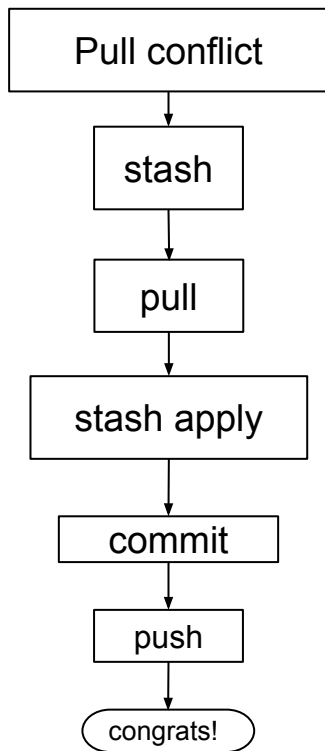
Conflict on push



- Кто-то отредактировал и запушил ваш файл перед вами.
- Делать нечего - надо сливать изменения, если хотите сохранить СВОИ КОММИТЫ
 - А еще вы можете их откатить или просто удалить репо и клонировать заново



Conflict on pull



- Кто-то отредактировал и запустил ваш файл перед вами, *НО ВЫ обнаружили это до коммита*
- Делать нечего - надо сливать изменения
 - stash - “складировать” свои незакомиченные изменения в надежное место
 - stash apply - если хотите вернуть свои изменения



Совместная работа - с коллабораторами

- Публичный репозиторий
 - Все могут видеть
 - Приглашенный коллаборатор может писать (push)
- Приватный
 - Только вы и приглашенный коллаборатор можете видеть и читать
 - Приглашенный коллаборатор может писать (push)
- Добавьте коллаборатора
 - repo->Settings->Manage Access->Add Collaborator

Deploy keys

- **Общий доступ на чтение к приватному репо**
 - Позволяет обойтись без приватных логина/пароля или ключа
 - Слишком много привилегий утекает!
- **Называется так из-за типичного сценария использования**
 - Загрузка приватного репо с помощью скриптов на сервере
- **Пара обычных ssh ключей**
 - Публичный загружается в гитхаб и ассоциируется только с одним репо
 - Приватный используется для скачивания репо (clone, pull)
- **Можно дать и доступ на запись в репо (push) по deploy key**
 - Но не очень хорошо с точки зрения безопасности.
- **Что делать**
 - repo->Settings->Deploy keys
 - <https://developer.github.com/v3/guides/managing-deploy-keys/#deploy-keys>