

IT РЕШЕНИЯ



www.mmtr.ru

Программирование на языках высокого уровня

Гутман Александр Сергеевич

IT РЕШЕНИЯ

www.mmtr.ru

Что требуется
для работы

- **Git**
- **Git extension / TortoiseGit**

<https://gitforwindows.org/>

<http://gitextensions.github.io/>

<https://tortoisegit.org/>

Проверка

git –version

git version 2.19.0.windows.1

Что это?

Git — распределённая система контроля версий, которая даёт возможность разработчикам отслеживать изменения в файлах и работать совместно с другими разработчиками

Что это?

- Системы контроля версий (СКВ, VCS, Version Control Systems) позволяют разработчикам сохранять все изменения, внесённые в код
- Существует три типа СКВ: локальная, централизованная и распределённая

Что это?

- Локальные системы контроля версий (ЛСКВ)

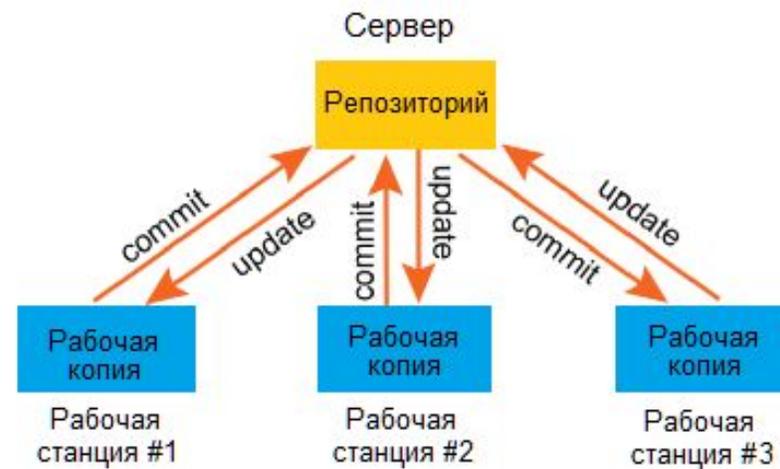
Локальный компьютер



Что это?

- Централизованные системы контроля версий (ЦСКВ)

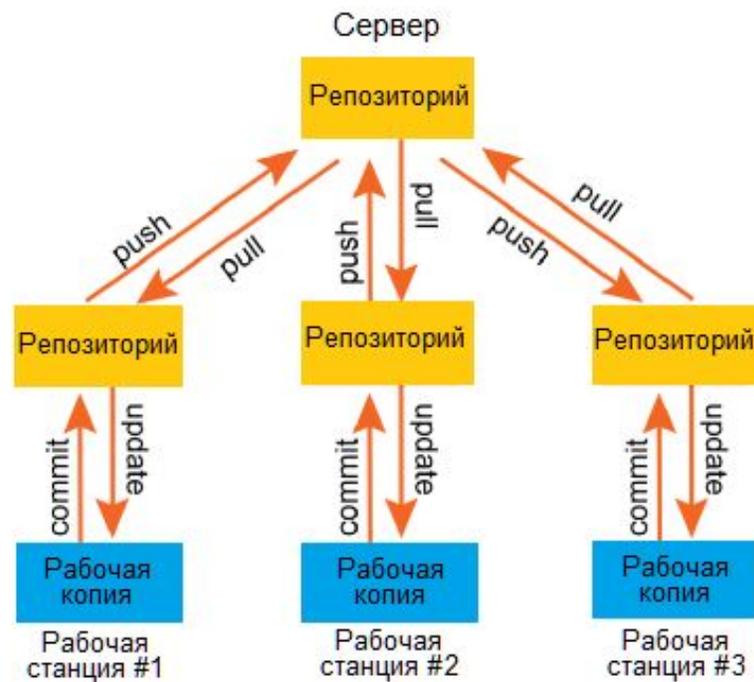
Централизованная система контроля версий



Что это?

- Распределённые системы контроля версий (РСКВ)

Распределённая система контроля версий



Что это?

- **Преимущества Git**
 - **Бесплатный и open-source**
 - **Небольшой и быстрый**
 - **Резервное копирование**
 - **Простое ветвление**

Что это?

- **КОМАНДА GIT**
 - \$ git опции команда аргументы

Что это?

Опции

- **-C** — использовать указанную папку репозитория вместо текущей папки;
- **-c параметр=значение** — использовать указанное значение параметра конфигурации;
- **-p** — прокручивать весь вывод с помощью less;

Команды

Что это?

- **add** — добавить файл или папку в репозиторий git;
- **am** — применить все патчи из email;
- **archive** — создать архив файлов;
- **bisect** — использовать бинарный поиск для поиска нужного коммита;
- **branch** — управление ветками проекта;
- **bundle** — перемещение объектов и ссылок в архиве;
- **checkout** — переключение между ветками;
- **cherry-pick** — внести изменения в уже существующие коммиты;
- **clean** — удалить все неотслеживаемые файлы и папки проекта;
- **clone** — создать копию удаленного репозитория в папку;
- **commit** — сохранить изменения в репозиторий;
- **diff** — посмотреть изменения между коммитами;
- **fetch** — скачать удаленный репозиторий;
- **init** — создать репозиторий;
- **merge** — объединить две ветви;
- **pull** — интегрировать удаленный репозиторий с локальным;
- **push** — отправить изменения в удаленный репозиторий;
- **tag** — управление тегами;
- **worktree** — управление деревьями разработки.

Инструкция

1. Создать папку для проекта
2. `git init`
3. `git add .`
4. `git commit -m "<commit>" -a`
5. `git remote add origin`
6. `git push origin master`

Инструкция

Создать ветку

–git checkout –b <name>

Переключаться между ветками

–git checkout <name>

Посмотреть ветки

–git branch

Просмотр текущего статуса

–git status

Объединить (слить) ветки

–git merge <name>

Итого:

- **Git — это очень популярная система контроля версий и совместной разработки проектов с открытым исходным кодом. С помощью Git вы можете отслеживать изменения в исходном коде своих проектов, возвращать предыдущие версии в случае критических ошибок, а также делиться своим кодом со всеми желающими и принимать от них исправления**

Задачи:

1. Написать веб-приложение:
 - По нажатию на кнопку отправляет данные с формы по электронной почте
 - По нажатию на кнопку на форме получает или отправляет данные в БД
2. Написать чат для обмена текстовыми сообщениями (сервер + клиент)
3. Задача БОНУС
4. Тест