# Семинар №3. Git

Клиент: <u>https://github.com/git-for-windows/git/releases/download/v2.10.0.windows.1/PortableGit-</u><u>2.10.0-64-bit.7z.exe</u>

#### Запускать: bin/git.exe

## Информация о рабочем репозитории

git clone http://gitlab.dev.ccfit.nsu.ru/students2016/[project name].git Используйте свою фамилию в качестве имени проекта: bobrov dymova example fink golovin ivanov kapustina kondyrev kushner morskiy muhin nechahin sklyarov zharkova В Git всегда origin – имя вашего оригинального репозитория Чтение не требует прав доступа. Для вливания изменений:

User: student

Password: password

## Работа через командный интерфейс

- 1. Проверить работу с файлами и с commit/push/fetch/pull:
  - Сделать изменения в файле своего проекта
  - Закоммитить в локальный репозиторий
  - Посмотреть лог изменений по файлу
  - Сравнить 2 версии файла через diff
  - Вытащить предыдущую версию файла
  - Залить изменения на сервер
  - Вытащить проект с сервера в другую директорию
  - Поправить файл в другой директории и залить на сервер
  - Получить измененный файл с сервера в первой директории
- 2. Работа с ветками
  - Сделать тестовую ветку
  - Переключиться в нее
  - Сделать изменения в ветке и залить на сервер
  - Зайти в другую директорию и вытащить новую ветку
  - Сделать изменения в ветке в другой директории и залить на сервер
  - Зайти в первую директорию и вытащить изменения
  - Влить ветку в master и залить на сервер

- Зайти в другую директорию, переключиться в master и вытащить изменения
- 3. Подключение второго remote репозиторию
  - Подключить другой репозиотрий: git remote add example <u>http://gitlab.dev.ccfit.nsu.ru/students2016/example.git</u> Теперь example – имя вашего второго внешнего репозитория
  - Вытащите ветку master из example
  - Создайте новую ветку в example (используйте свою фамилию) на основе master
  - Залейте на сервер изменения
  - Вернитесь в orgin ветку master
  - Влейте в нее ветку из remote
  - Залейте на сервер изменения в origin/master

## Работа через TortoiseGit

- 1. Посмотреть структуру вашего репозитория, ветки и коммиты. Посмотрите как выглядит example
- 2. Проверить работу с конфликтами:
  - Поменять файл в master
  - Поменять файл в том же месте в другой ветке
  - разрешить конфликт при merge ветки в мастер
- 3. Проверить работу с rebase:
  - Поменять файл в master
  - Поменять другой файл в другой ветке
  - Влить ветку в master через rebase
  - Посмотреть графически как выглядит ветка после rebase
  - Сделайте 3 изменения в ветке
  - Запустите interactive rebase и объедините все изменения в одно
  - Залейте в мастер получившийся коммит как merge (--no-ff)