

Python

06

Git

Git - распределенная система контроля версий (VCS)

Git — это набор консольных утилит, которые отслеживают и фиксируют изменения в файлах.

Настройка Git

```
git config --global user.name "My Name"
```

```
git config --global user.email myEmail@example.com
```

Создание(Инициализация) нового репозитория

```
mkdir tms-21
```

```
cd tms-21
```

```
git init
```

Определение текущего состояния

`git status`

Состояния файлов

1. Зафиксированное(файл уже сохранен в локальной базе)
2. Измененное(файлы были изменены, но еще не были зафиксированы)
3. Подготовленное(измененные файлы, отмеченные для включения в следующий коммит)

Подготовка файлов

```
git add filename.txt
```

Фиксация изменений

```
git commit -m 'Your comment'
```


Создание репозитория

<https://github.com/>

Работа с удаленными репозиториями

```
git remote add origin git@github.com:alexVarkalov/tms-21-pupils.git
```

Отправка изменений на сервер

```
git push origin master
```

Клонирования репозитория

```
git clone https://github.com/alexVarkalov/tms-21-pupils.git
```

Запрос изменений с сервера

```
git pull origin master
```

Работа с ветками

`git branch my_new_branch` - создание новой ветки

`git branch` - показать все ветки

`git checkout my_new_branch` - переключение на другую ветку

`git checkout -b my_new_branch_2` - создание новой ветки и переключение на нее

`git merge my_new_branch` - слияние веток

`git branch -d my_new_branch` - удаление ветки

`git log` - отображение информации о коммитах

`git show commit_id` - отображение информации об изменении в коммите

`git diff commit1_id commit2_id` - отображение разницы между двумя коммитами

Файл .gitignore

<https://github.com/github/gitignore/blob/master/Python.gitignore>

Инструкция по работе с git при выполнении ДЗ

1. `git checkout master`
2. `git checkout -b 'hw99'`
3. Сделать первое задание
4. `git add task_99_01.py`
5. `git commit -m 'task_99_01.py'`
6. Сделать второе задание
7. ...
8. ...
9. `git push origin hw99`