

Отчёт по лабораторной работе №3

Система контроля версий Git

Мулин Иван Владимирович

Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
2.1	Базовая настройка git	5
2.2	Создание ключа SSH	5
2.3	Создание рабочего пространства	6
2.4	Настройка каталога курса	6
2.5	Выполнение заданий для самостоятельной работы	8
3	Заключение	9

Список иллюстраций

2.1	Конфигурация пользователя	5
2.2	Стандартная настройка git	5
2.3	Создание рабочего пространства	6
2.4	Клонирование нового репозитория (Снимок обрезан.)	6
2.5	Конфигурация каталога курса	6
2.6	Добавление файлов в коммит (Снимок обрезан.)	6
2.7	Отправка файлов на сервер	7
2.8	Проверка соблюдения иерархии рабочего пространства (Снимок обрезан.)	7
2.9	Копирование отчётов в каталог курса (Снимок обрезан.)	8
2.10	Добавление отчётов в коммит (Снимок обрезан.)	8

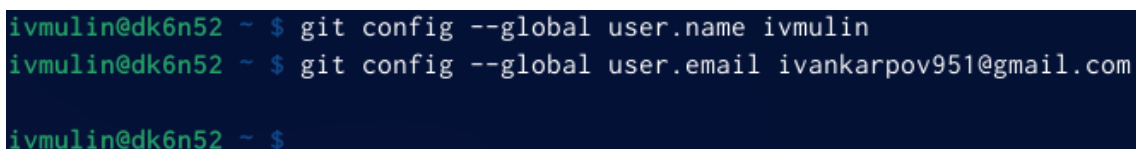
1 Цель работы

Цель данной лабораторной работы состоит в изучении применения средств контроля версий и приобретении практических умений по работе с системой git. В ходе выполнения работы будет создан репозиторий, который можно найти по адресу https://github.com/ivmulin/study_2022-2023_arch-ps.

2 Выполнение лабораторной работы

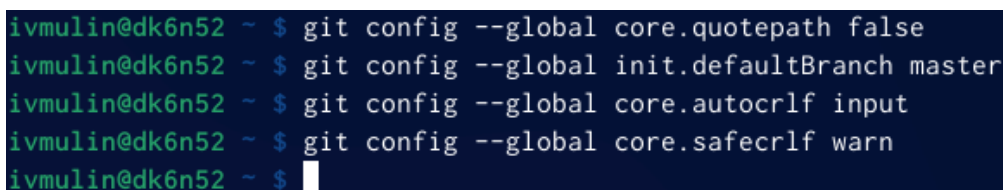
2.1 Базовая настройка git

Прежде чем создать репозиторий, необходимо настроить git:



```
ivmulin@dk6n52 ~ $ git config --global user.name ivmulin
ivmulin@dk6n52 ~ $ git config --global user.email ivankarpov951@gmail.com
ivmulin@dk6n52 ~ $
```

Рис. 2.1: Конфигурация пользователя



```
ivmulin@dk6n52 ~ $ git config --global core.quotePath false
ivmulin@dk6n52 ~ $ git config --global init.defaultBranch master
ivmulin@dk6n52 ~ $ git config --global core.autocrlf input
ivmulin@dk6n52 ~ $ git config --global core.safecrlf warn
ivmulin@dk6n52 ~ $
```

Рис. 2.2: Стандартная настройка git

2.2 Создание ключа SSH

SSH-ключ есть пара ключей, которая необходима при подключения к серверу по протоколу SSH. Создаём ключ командой `ssh-keygen` (в качестве комментария можно добавить к команде опцию `-C "..."`). Теперь необходимо установить соединение клиента с сервером github. Для этого копируем только что сгенерированный ключ, вставляем в окне создания нового SSH-ключа и нажимаем Add SSH key.

2.3 Создание рабочего пространства

```
ivmulin@dk6n52 ~ $ mkdir -p ~/work/study/2022-2023/"Архитектура компьютера"  
ivmulin@dk6n52 ~ $
```

Рис. 2.3: Создание рабочего пространства

На этом шаге необходимо лишь создать каталог рабочего пространства для соответствующего предмета.

2.4 Настройка каталога курса

С помощью шаблона создаём репозиторий и клонируем его в рабочее пространство:

```
ivmulin@dk2n26 ~/work/study/2022-2023/Архитектура компьютера $ git clone --recursive git@github.com:ivmulin/study_2022-2023_arch-pc.git arch-pc  
Клонирование в «arch-pc»...
```

Рис. 2.4: Клонирование нового репозитория (Снимок обрезан.)

В созданном каталоге удаляем лишние файлы и добавляем необходимые директории:

```
ivmulin@dk2n26 ~/work/study/2022-2023/Архитектура компьютера $ cd arch-pc  
ivmulin@dk2n26 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $ rm package.json  
ivmulin@dk2n26 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $ echo arch-pc > COURSE  
ivmulin@dk2n26 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $ make  
ivmulin@dk2n26 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $
```

Рис. 2.5: Конфигурация каталога курса

```
ivmulin@dk2n26 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $ git add .  
ivmulin@dk2n26 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $ git commit -am 'feat(main): make course structure'  
[master f33d6d0] feat(main): make course structure
```

Рис. 2.6: Добавление файлов в коммит (Снимок обрезан.)

После этого необходимо загрузить все файлы на сервер командой `git push`.

```
ivmulin@dk2n26 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $ git push
Перечисление объектов: 22, готово.
Подсчет объектов: 100% (22/22), готово.
При сжатии изменений используется до 6 потоков
Сжатие объектов: 100% (16/16), готово.
Запись объектов: 100% (20/20), 310.94 КиБ | 7.77 МиБ/с, готово.
Всего 20 (изменений 1), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:ivmulin/study_2022-2023_arch-pc.git
   3a39c1e..f33d6d0  master -> master
ivmulin@dk2n26 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $
```

Рис. 2.7: Отправка файлов на сервер

Проверить правильность создания иерархии рабочего пространства в локальном репозитории можно с помощью команды

```
tree -d /work/study
```

(Команда печатает всю иерархию выбранной либо текущей директории, а в сочетании с ключом `-d` выводит только каталоги.)

```
ivmulin@dk2n26 ~ $ tree -d ~/work/study
/afs/.dk.sci.pfu.edu.ru/home/i/v/ivmulin/work/study
├── 2022-2023
│   ├── Архитектура компьютера
│   │   ├── arch-pc
│   │   │   ├── config
│   │   │   │   ├── course
│   │   │   │   └── script
│   │   ├── labs
│   │   │   └── lab01
```

Рис. 2.8: Проверка соблюдения иерархии рабочего пространства (Снимок обр-зан.)

2.5 Выполнение заданий для самостоятельной работы

```
labs $ cp ~/Документы/"Отчёт 1.pdf" lab01/report/"Отчёт 1.pdf"  
labs $ cp ~/Документы/"Отчёт 2.pdf" lab02/report/"Отчёт 2.pdf"  
labs $ cp ~/Документы/"Отчёт 3.pdf" lab03/report/"Отчёт 3.pdf"  
labs $
```

Рис. 2.9: Копирование отчётов в каталог курса (Снимок обрезан.)

Задания для самостоятельной работы предполагают добавление к созданному репозиторию отчётов по лабораторным работам № 1, 2 и 3. Для этого достаточно скопировать отчёты по предыдущим (первой и второй) и текущей лабораторным работам в соответствующие им папки в каталоге курса: Следом необходимо отправить все изменённые файлы в репозиторий командой `git push`.

```
labs $ git add .  
labs $ git commit -am "add reports to lab01, lab02, lab03"
```

Рис. 2.10: Добавление отчётов в коммит (Снимок обрезан.)

3 Заключение

В результате выполнения лабораторной и самостоятельной работ были получены прикладные навыки работы с системой контроля версий git, а значит, цель работы была достигнута.