

# Лабораторная работа № 2:

## Система контроля версий Git

ФИО: Аккур Имад

Студенческий Билет : 1032239342

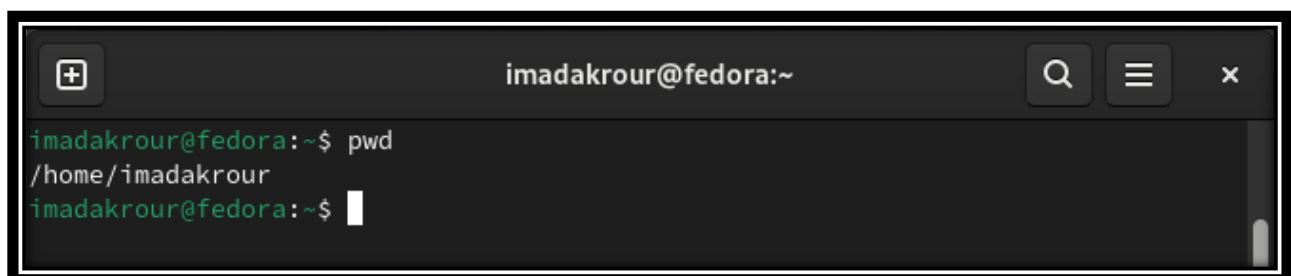
### 1.1 Цель лабораторной работы:

Целью работы является изучить идеологию и применение средств контроля версий. Приобрести практические навыки по работе с системой **git**.

### 1.3 выполнения заданий лабораторной работы и результаты:

#### 1.4.1. Создание учетной записи на GitHub:

**Описание задания:** Для начала необходимо создать учетную запись на сайте **GitHub**. Это позволит получить доступ к управлению репозиториями для работы над проектами.


A terminal window with a dark background. The title bar shows 'imadakrour@fedora:~'. The prompt is 'imadakrour@fedora:~\$'. The command 'pwd' has been entered, and the output is '/home/imadakrour'. The prompt is now 'imadakrour@fedora:~\$' followed by a cursor.

```
imadakrour@fedora:~$ pwd
/home/imadakrour
imadakrour@fedora:~$
```

*в моем случае учетная запись уже существует, поэтому этот шаг был пропущен.*

### 1.4.2. Базовая настройка Git:

1. Указание имени пользователя и email для создания коммитов в репозиториях. Эти данные будут добавляться к каждому коммиту, чтобы идентифицировать автора изменений.:

A terminal window titled 'imadakrour@fedora:~' with search, menu, and close icons in the title bar. The terminal shows a sequence of six 'git config' commands being executed. The prompt 'imadakrour@fedora:~\$' is repeated before each command. The commands are: 'git config --global user.name "imadakrour"', 'git config --global user.email "1032239342@pfur.ru"', 'git config --global core.quotepath false', 'git config --global init.defaultBranch master', 'git config --global core.autocrlf input', and 'git config --global core.safecrlf warn'. The final prompt is 'imadakrour@fedora:~\$'.

```
imadakrour@fedora:~$ git config --global user.name "imadakrour"
imadakrour@fedora:~$ git config --global user.email "1032239342@pfur.ru"
imadakrour@fedora:~$ git config --global core.quotepath false
imadakrour@fedora:~$ git config --global init.defaultBranch master
imadakrour@fedora:~$ git config --global core.autocrlf input
imadakrour@fedora:~$ git config --global core.safecrlf warn
imadakrour@fedora:~$
```

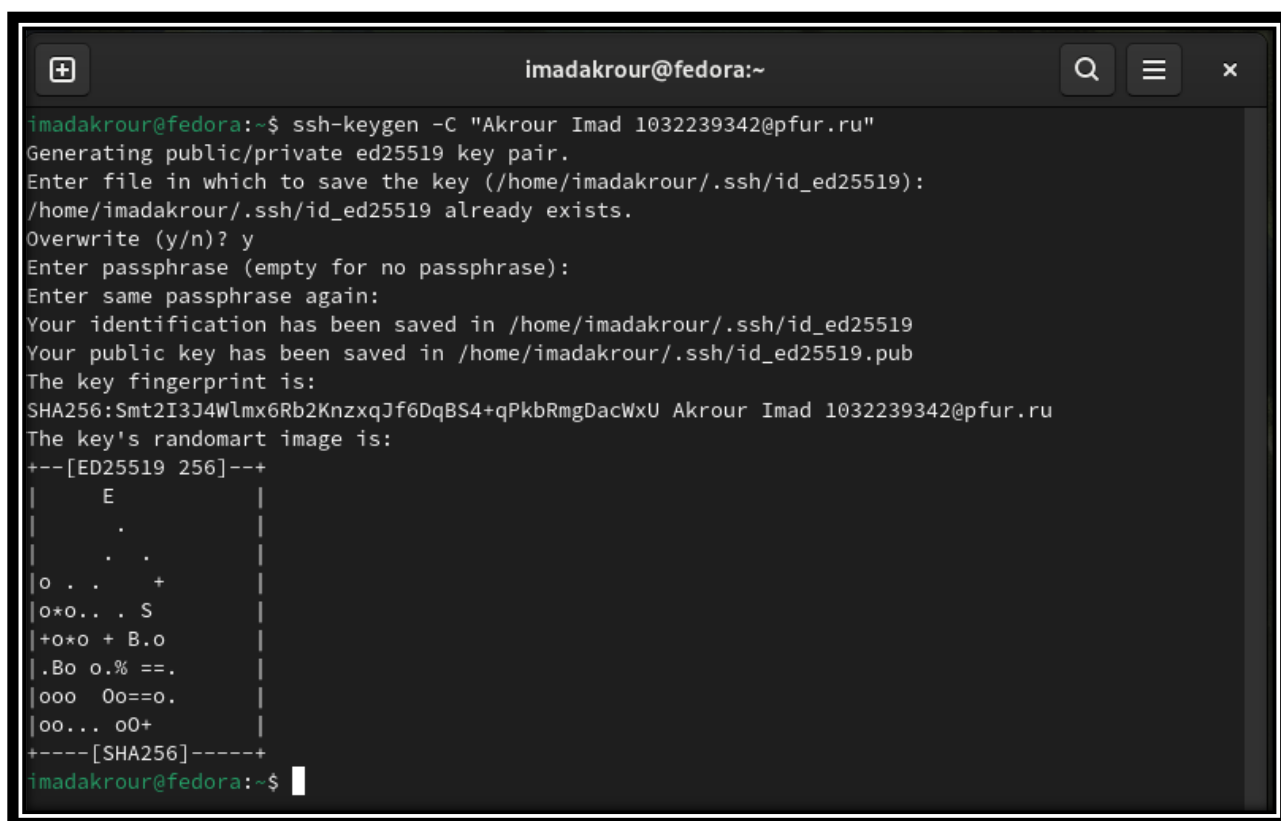
*Эта последовательность команд настраивает имя пользователя, email, кодировку UTF-8, задает начальную ветку master, устанавливает обработку концов строк (autocrlf), и включает предупреждения о несоответствиях концов строк (safecrlf). Выполнение всех команд в одной строке позволяет быстрее закончить настройку и сразу перейти к работе с Git. Все параметры будут применены глобально для всех будущих репозиториев.*

### 1.4.3. Создание SSH-ключа:

#### Описание выполняемого задания:

Для безопасного подключения к репозиториям на GitHub, нужно сгенерировать SSH-ключи (публичный и приватный). Это позволит вам работать с репозиториями, не вводя каждый раз логин и пароль.

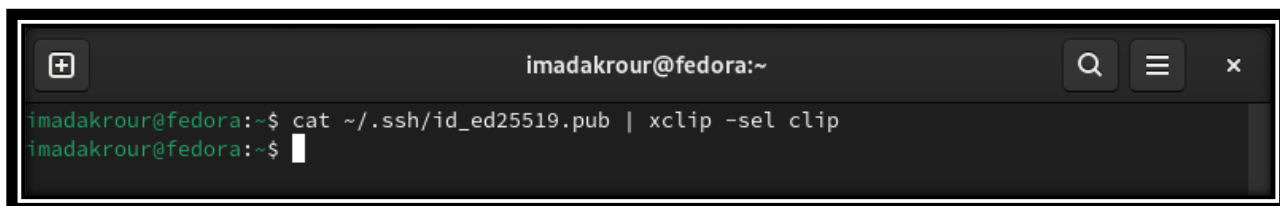
#### 1. Генерация SSH-ключа:



```
imadakrour@fedora:~$ ssh-keygen -C "Akrour Imad 1032239342@pfur.ru"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/imadakrour/.ssh/id_ed25519):
/home/imadakrour/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/imadakrour/.ssh/id_ed25519
Your public key has been saved in /home/imadakrour/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:Smt2I3J4Wlmx6Rb2KnzqxJf6DqBS4+qPkbRmgDacWxU Akrou r Imad 1032239342@pfur.ru
The key's randomart image is:
+--[ED25519 256]--+
|      E          |
|      .          |
|      . .        |
| o . . . +       |
|o+o.. . S        |
|+o*o + B.o       |
|.Bo o.% ==.      |
|ooo 0o==o.       |
|oo... o0+        |
+-----[SHA256]-----+
imadakrour@fedora:~$
```

Каталог `~/ .ssh/` — это стандартное место для хранения SSH-ключей. Не изменяйте путь, если вы не хотите использовать другое место для хранения.

#### Копирование публичного ключа в буфер обмена:



```
imadakrour@fedora:~$ cat ~/.ssh/id_ed25519.pub | xclip -sel clip
imadakrour@fedora:~$
```

Команда `cat ~/.ssh/id_rsa.pub` выводит содержимое публичного ключа, а команда `xclip -sel clip` копирует это содержимое в буфер обмена

## Add new SSH Key

**Title**

SSH for Архитектура лабы


**Key type**

Authentication Key

**Key**

```
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAILStSaFs3NKkC0DwZqNnoGTrInScQE9LOOdYTW+rmJjM Akroure Imad 1032239342@pfur.ru
```

Add SSH key

**Имад Акруп** (imadakroure)  
Your personal account

[Public profile](#)  
[Account](#)  
[Appearance](#)  
[Accessibility](#)  
[Notifications](#)  
  
**Access**  
[Billing and plans](#)  
[Emails](#)  
[Password and authentication](#)  
[Sessions](#)  
**[SSH and GPG keys](#)**  
[Organizations](#)  
[Enterprises](#)  
[Moderation](#)

[Go to your personal profile](#)

## Add new SSH

**Title**

**Key type**

Authentication Key

**Key**

```
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAILStSaFs3NKkC0DwZqNnoGTrInScQE9LOOdYTW+rmJjM Akroure Imad 1032239342@pfur.ru
```

Add SSH key

Copy Ctrl+C

Paste Ctrl+V

Paste as plain text Ctrl+Shift+V

Select all Ctrl+A

Open in sidebar

Writing direction

Block element...

Inspect

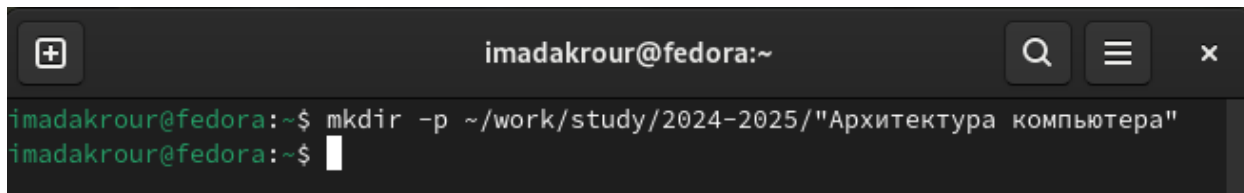
вставить ключ на сайт GitHub.

#### 1.4.4. Создание рабочего пространства и репозитория курса на основе шаблона:

Для правильной организации рабочих файлов и проектов в рамках курса необходимо создать рабочее пространство по определенной структуре. В этом шаге будет выполнено создание директории для предмета «Архитектура компьютера», а также будет продемонстрировано, как структурировать папки для лабораторных работ.

**Описание выполняемого задания:**

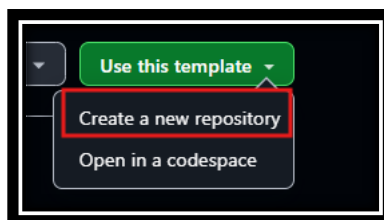
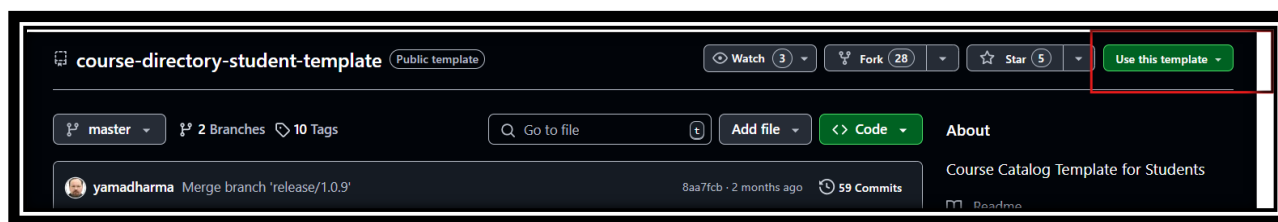
**Создание структуры рабочего пространства :**




```
imadakrour@fedora:~$ mkdir -p ~/work/study/2024-2025/"Архитектура компьютера"
imadakrour@fedora:~$
```

*Эта команда создаст нужные каталоги по иерархии. Опция -p создает промежуточные каталоги, если они еще не существуют*

#### 1.4.5. Сознание репозитория курса на основе шаблона:



Owner \*

 imadakraour

Repository name \*

study\_2024-2025\_arch-pc

study\_2024-2025\_arch-pc is available.

Great repository names are short and memorable. Need inspiration? How about **jubilant-enigma** ?


Description (optional)

☒ Public

Anyone on the internet can see this repository. You choose who can commit.

☐ Private

You choose who can see and commit to this repository.

 You are creating a public repository in your personal account.

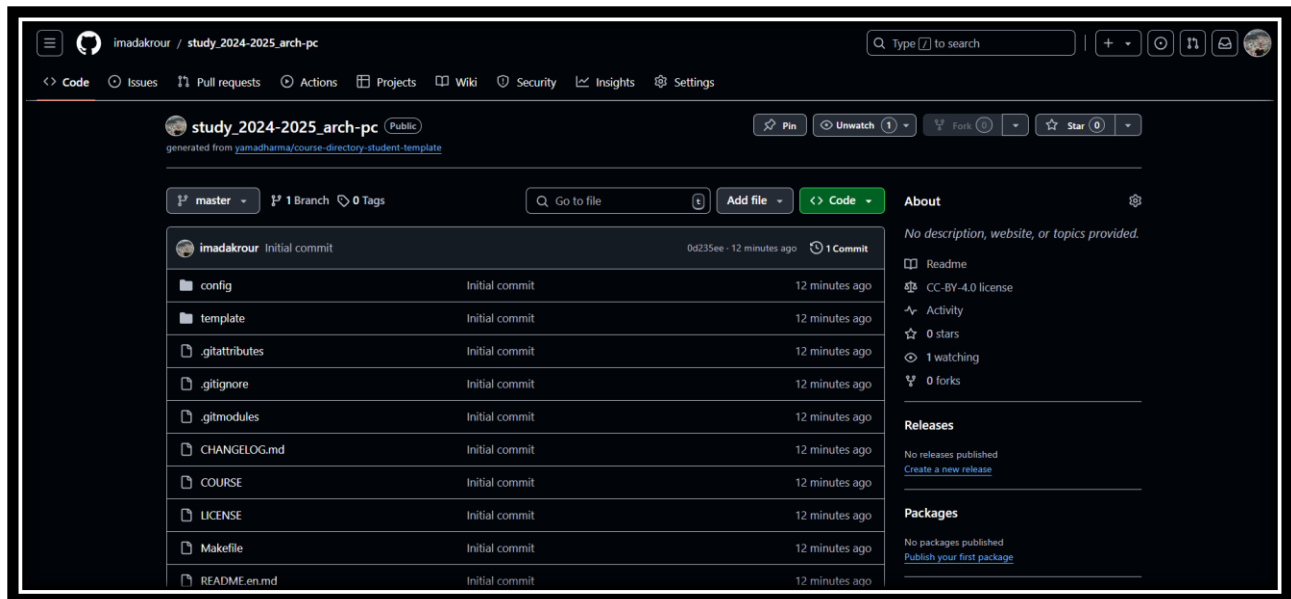
Create repository

Эти действия создадут новый репозиторий на основе предоставленного шаблона, который содержит структуру и необходимые файлы для работы по курсу.

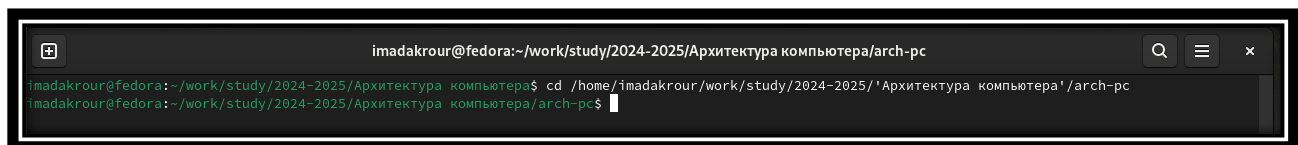
#### Клонирование репозитория на локальный компьютер :

```
imadakraour@fedora:~/work/study/2024-2025/Архитектура компьютера
imadakraour@fedora:~/work/study$ cd /home/imadakraour/work/study/2024-2025/Архитектура компьютера
imadakraour@fedora:~/work/study/2024-2025/Архитектура компьютера$ git clone --recursive git@github.com:imadakraour/study_2024-2025_arch-pc.git arch-pc
Cloning into 'arch-pc'...
remote: Enumerating objects: 33, done.
remote: Counting objects: 100% (33/33), done.
remote: Compressing objects: 100% (32/32), done.
remote: Total 33 (delta 1), reused 18 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (33/33), 18.85 KiB | 448.00 KiB/s, done.
Resolving deltas: 100% (1/1), done.
Submodule 'template/presentation' (https://github.com/yamadharma/academic-presentation-markdown-template.git) registered for path 'template/presentation'
Submodule 'template/report' (https://github.com/yamadharma/academic-laboratory-report-template.git) registered for path 'template/report'
Cloning into '/home/imadakraour/work/study/2024-2025/Архитектура компьютера/arch-pc/template/presentation'...
remote: Enumerating objects: 111, done.
remote: Counting objects: 100% (111/111), done.
remote: Compressing objects: 100% (77/77), done.
remote: Total 111 (delta 42), reused 100 (delta 31), pack-reused 0 (from 0)
Receiving objects: 100% (111/111), 102.17 KiB | 968.00 KiB/s, done.
Resolving deltas: 100% (42/42), done.
Cloning into '/home/imadakraour/work/study/2024-2025/Архитектура компьютера/arch-pc/template/report'...
remote: Enumerating objects: 142, done.
remote: Counting objects: 100% (142/142), done.
remote: Compressing objects: 100% (97/97), done.
remote: Total 142 (delta 60), reused 121 (delta 39), pack-reused 0 (from 0)
Receiving objects: 100% (142/142), 341.09 KiB | 1.27 MiB/s, done.
Resolving deltas: 100% (60/60), done.
Submodule path 'template/presentation': checked out 'c9b2712b4b2d431ad5086c9c72a02bd2fca1d4a6'
Submodule path 'template/report': checked out 'c26e22effe7b3e0495707d82ef561ab185f5c748'
imadakraour@fedora:~/work/study/2024-2025/Архитектура компьютера$
```

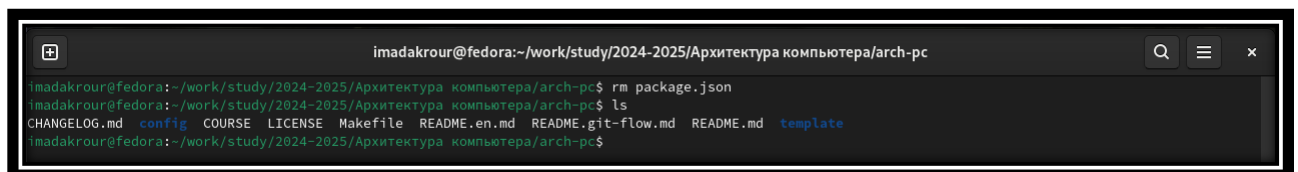
Команда `git clone --recursive` позволяет загрузить все файлы из удаленного репозитория в папку `arch-pc`.



Настройка каталога курса :

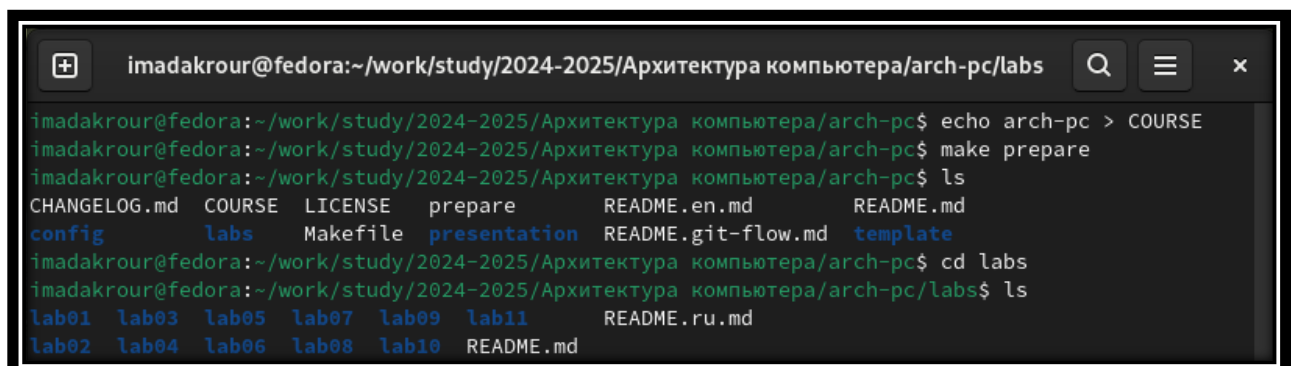


Удаление ненужного файла package.json :

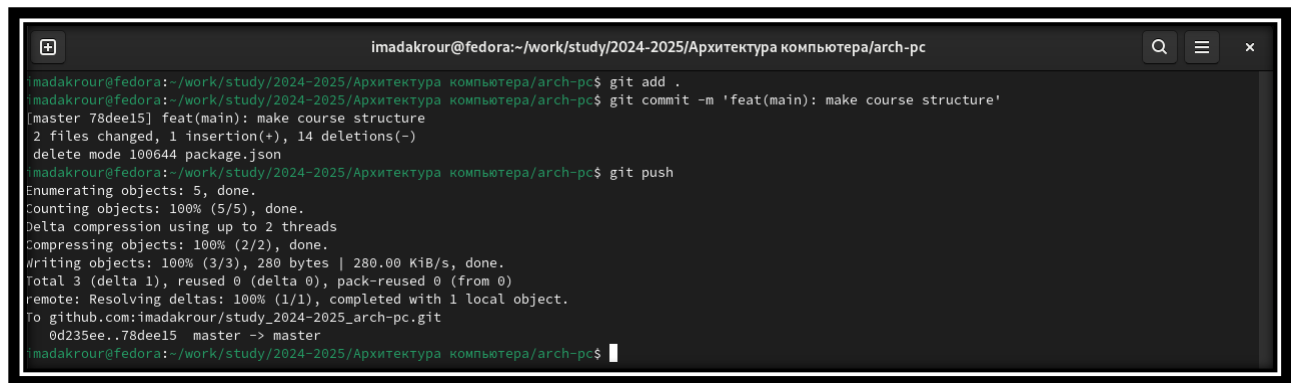


Удаление лишних файлов помогает избежать путаницы и оставить только необходимые для курса файлы.

Создание файла COURSE с названием курса:



**Отправка изменений на сервер :**



```
imadakrour@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc
imadakrour@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git add .
imadakrour@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git commit -m 'feat(main): make course structure'
[master 78dee15] feat(main): make course structure
2 files changed, 1 insertion(+), 14 deletions(-)
delete mode 100644 package.json
imadakrour@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 280 bytes | 280.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:imadakrour/study_2024-2025_arch-pc.git
   0d235ee..78dee15  master -> master
imadakrour@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$
```

*Эти команды добавляют изменения в локальный репозиторий, создают коммит с описанием и отправляют изменения в удаленный репозиторий на GitHub.*

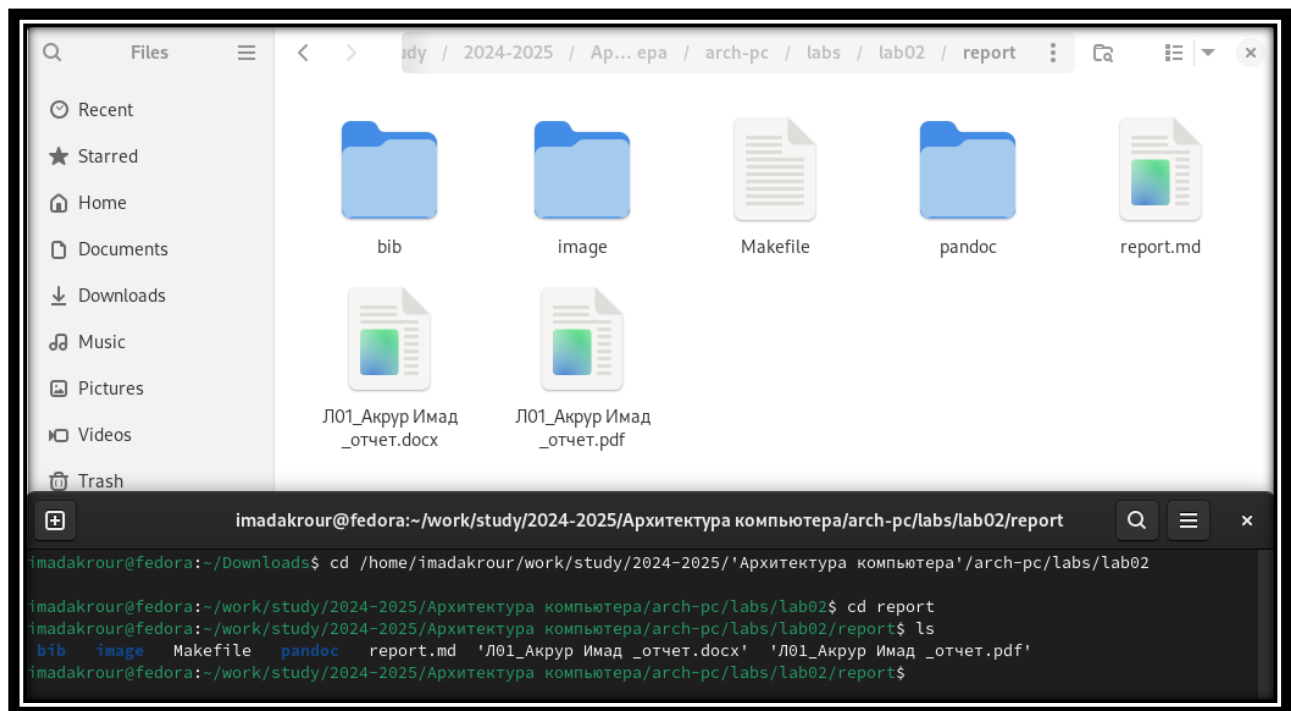


## 1.5 Описание результатов выполнения заданий для самостоятельной работы:

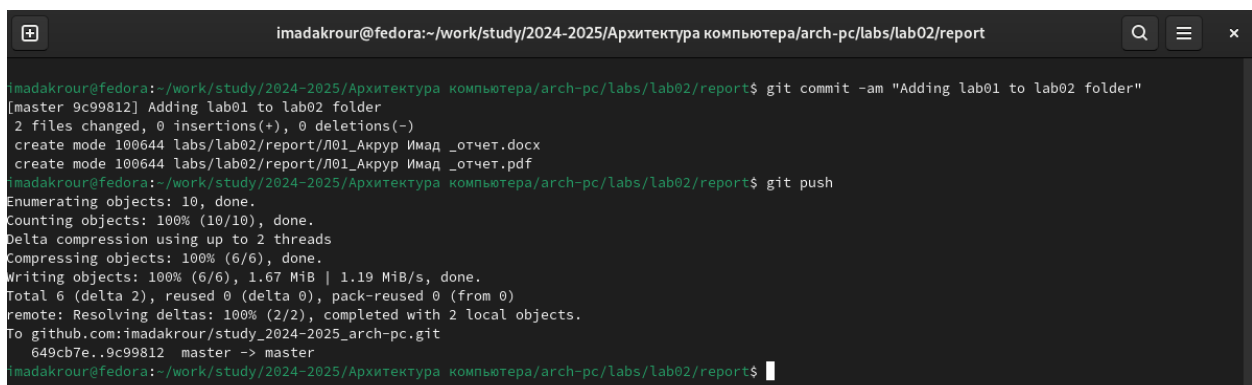
### 1. Создание отчета в соответствующем каталоге рабочего пространства:

#### Описание задания:

Данное задание включает в себя создание отчета о выполнении лабораторной работы, копирование предыдущих отчетов и загрузку файлов на GitHub.



*Я скопировал файлы предыдущего отчета в папку `lab02`, используя команду `cp`. Сначала я перешел в каталог с файлами отчета с помощью команды `cd`, а затем выполнил команду для копирования всех файлов*



*После копирования файлов я выполнил команды для отправки их на GitHub*

The screenshot shows a GitHub repository interface. At the top, the path is 'study\_2024-2025\_arch-pc / labs / lab02 / report /'. Below this, a commit by 'imadakrour' is shown with the message 'Adding lab01 to lab02 folder' and a timestamp of '9c99812 · 1 minute ago'. A table lists the files included in this commit:

Name	Last commit message	Last commit date
..		
bib	feat(main): make course structure	10 hours ago
image	feat(main): make course structure	10 hours ago
pandoc	feat(main): make course structure	10 hours ago
Makefile	feat(main): make course structure	10 hours ago
report.md	feat(main): make course structure	10 hours ago
Л01_Акрур Имад _отчет.docx	<a href="#">Adding lab01 to lab02 folder</a>	1 minute ago
Л01_Акрур Имад _отчет.pdf	Adding lab01 to lab02 folder	1 minute ago

*Затем я сделал скриншот в GitHub, чтобы показать, что файлы были обновлены и теперь отображаются в репозитории.*

## 1.7 Вопросы для самопроверки:

### 1. Что такое системы контроля версий (VCS) и для чего они предназначены?

Системы контроля версий (VCS) — это инструменты для управления изменениями в коде и файлах проекта, позволяющие отслеживать историю изменений, работать с разными версиями и сотрудничать в команде.

### 2. Объясните понятия VCS: хранилище, commit, история, рабочая копия

- Хранилище (repository) место, где хранятся все версии проекта.
- Commit зафиксированные изменения с комментарием, представляющие версию проекта.
- История последовательность всех commit
- Рабочая копия (working copy) локальная версия проекта, с которой работает разработчик.

### 3. Чем отличаются централизованные и децентрализованные VCS? Примеры.

- Централизованные VCS (например, SVN): одно центральное хранилище, доступное для всех.
- Децентрализованные VCS (например, Git): каждый пользователь имеет своё полное хранилище, с возможностью синхронизации.

### 4. Действия с VCS при единоличной работе с хранилищем.

- Инициализация репозитория.
- Добавление файлов и создание commit

- Проверка состояния (git status), просмотр истории (git log).

## **5. Порядок работы с общим хранилищем VCS.**

- Клонирование репозитория.
- Создание ветки, внесение изменений.
- Commit изменений.
- Слияние с основной веткой через pull request.

## **6. Основные задачи, решаемые Git.**

Управление версиями, создание веток, слияние изменений, разрешение конфликтов, работа с удалёнными репозиториями.

## **7. Команды Git: краткая характеристика.**

- `git init`: создание репозитория.
- `git clone`: клонирование удалённого репозитория.
- `git add`: добавление файлов для отслеживания.
- `git commit`: фиксация изменений.
- `git push`: отправка изменений в удалённый репозиторий.
- `git pull`: получение изменений из удалённого репозитория.

## **8. Примеры работы с локальными и удалёнными репозиториями.**

- Локальные : инициализация репозитория, создание commit , проверка состояния.
- Удалённые: клонирование, отправка изменений (git push), получение обновлений (git pull).

## **Комментарии и выводы по результатам выполнения заданий:**

Выполнение задания помогло закрепить навыки работы с системами контроля версий, организацией файлов в рабочем пространстве, а также загрузкой проектов на GitHub. Эти навыки важны для поддержания эффективной командной работы, обеспечения безопасности данных и удобства доступа к прошлым версиям проектов.