**РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ**

**Факультет физико-математических и естественных наук**

**Кафедра прикладной информатики и теории вероятностей**

**ОТЧЕТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ № 3**

Дисциплина: Архитектура компьютера

Студент: Пашутина Анна Алексеевна

Группа: НПИбд-02-25

Ст. билет: 1032253642

**МОСКВА**

**2025 г.Цель работы**

Целью работы является освоение процедуры оформления отчетов с помощью легковесного языка разметки Markdown.

Перед началом всех действий зайдем в терминал и узнаем, какая у нас архитектура компьютера(с помощью dpkg –print-architecture), чтобы установить нужный установочный файл из github и введем команду.

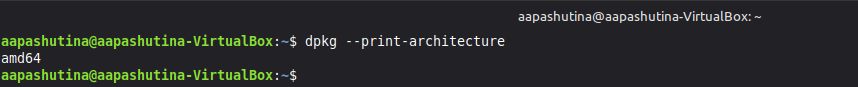


Рис. 1 Узнали архитектуру нашего компьютера.

Далее зайдем на github по ссылке, которую нам предоставили в лекции (<https://github.com/quarto-dev/quarto-cli/releases/tag/v1.7.34>)

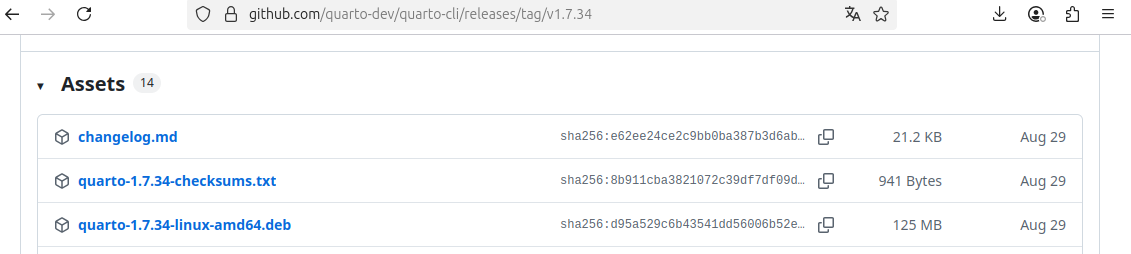
Опираясь на вывод из Рис.1 ищем пакет с нашим расширением - amd64.deb. 

Рис. 2 Перешли по ссылке и нашли нужный пакет (quarto-1.7.34-linux-amd64.deb)

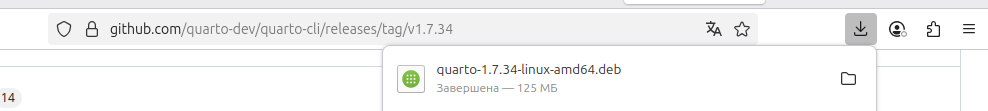


Рис. 3 Установили необходимое ПО.

После завершения установки переходим в терминал и пишем эту команду, чтобы установились все нужные зависимости. Потом проверяем, что Quarto установлен.

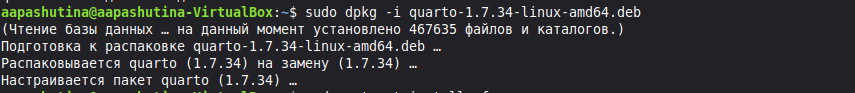


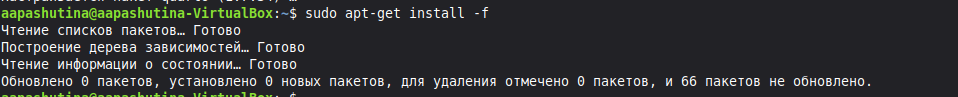
Рис.4 Распаковываем скачанные пакеты. 

Рис.5 Скачиваем необходимые зависимости.

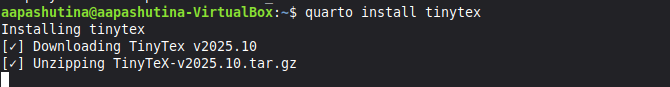


Рис.6 Устанавливаем Tinytex.

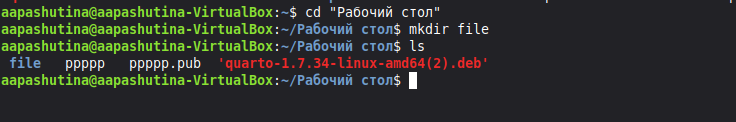


Рис.7 Создали папку под названием file на рабочем столе и проверим его наличие.

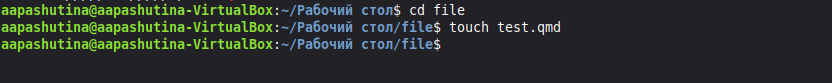


Рис.8 Перешли в нашу созданную папку и создали там файл с расширением .qmd.

Теперь мы готовы начать работу в Markdown.(с учетом того, что у меня по умолчанию настроен Makefile)

**3.4. Порядок выполнения лабораторной работы**

1. Откроем терминал.

2. Перейдем в каталог курса, сформированный при выполнении лабораторной работы № 2 с помощью команды cd, указав полный пути до каталога

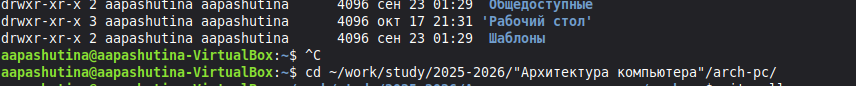


Рис. 3.4.1 Перешли в каталог курса, который сформировали во время выполнения лаб.работы номер 2.

Обновим локальный репозиторий, скачав изменения из удаленного репозитория с помощью команды git pull.

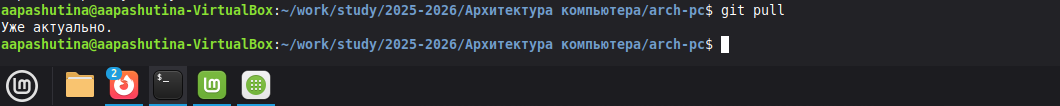


Рис. 3.4.2 Обновили локальный репозиторий.

3.Перейдем в каталог с шаблоном отчета по лабораторной работе № 3 с помощью команды cd, указав полный путь до этого каталога.

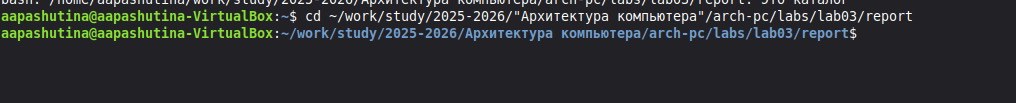


Рис. 3.4.3 Перешли в каталог с шаблоном отчета по лабораторной работе № 3.

4. Проведем компиляцию шаблона с использованием Makefile.

Для начала проверим, что настройки Makefile верны.

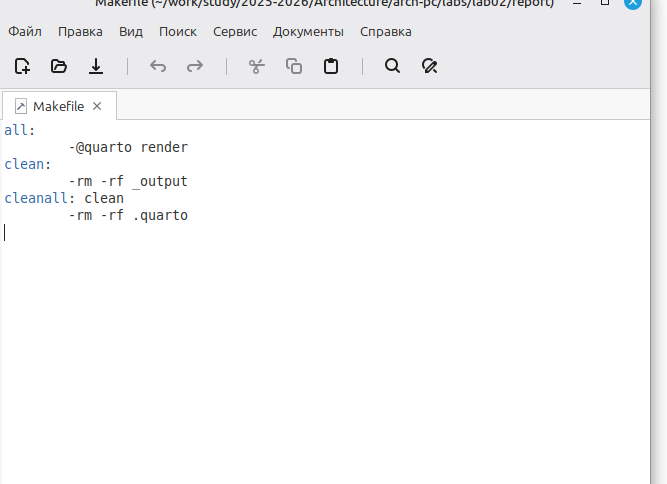
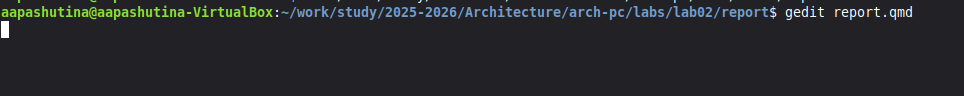


Рис. 4.0

Для этого введем команду make.

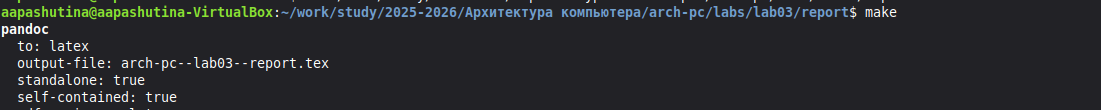


Рис. 4.1 Проведем компиляцию шаблона с использованием Makefile.

При успешной компиляции должны сгенерироваться файлы report.pdf и report.docx. Откройем и проверем корректность полученных файлов.

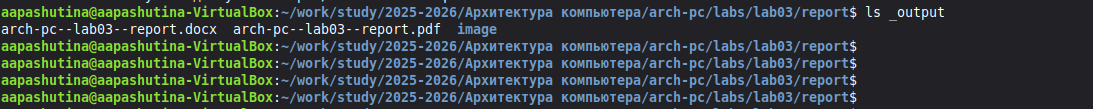


Рис. 4.2 Проверили наличие созданных файлов.

5. Удалим полученный файлы с использованием Makefile. Для этого введем команду make clean.

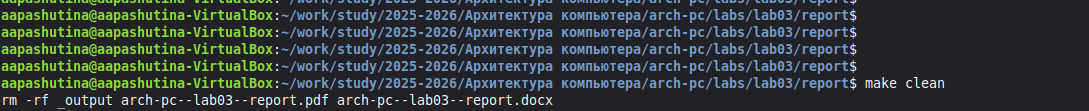


Рис. 5.1 Применили команду make clean.

Проверим, что после этой команды файлы report.pdf и report.docx были удалены, для этого выведем созданный с помощью команды make каталог \_output(Как мы видим, теперь его нет).

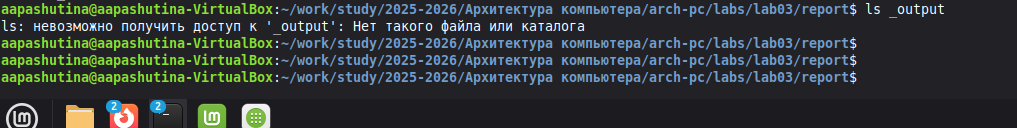


Рис. 5.2 Проверили, удалился ли подкаталог \_output(где и были файлы с расширениями .docx и .pdf).

6. Откроем файл report.md c помощью любого текстового редактора, например gedit с помощью команды gedit report.qmd.

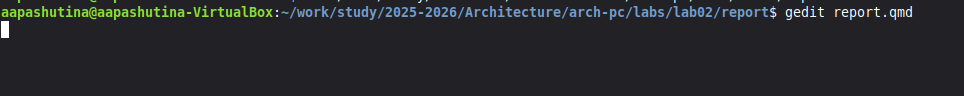


Рис.6.1 Открыли файл report.qmd в текстовом редакторе gedit.

Теперь мы готовы выполнить отчет по 2 и 3 лабораторным работам.

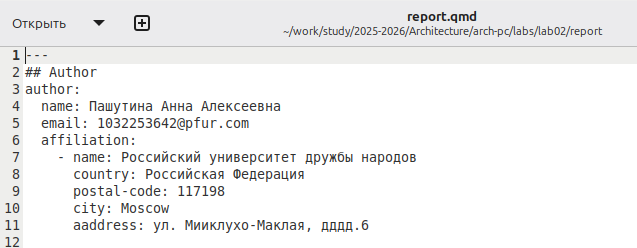


Рис.6.2 Создаем отчет в файле с расширением .qmd.

После написания отчетов, с помощью make конвертируем их в файлы с расширениями .pdf и .docx.

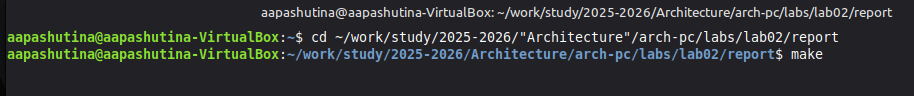


Рис. 6.3 Создали отчеты в файлах с требуемыми расширениями.

Теперь проверим, что файлы действительно появились.

Рис.

