

## Лабораторная работа №3. Markdown

---

Подготовил:

Королев Адам Маратович

Группа: НПИбд-02-21

Студенческий билет: № 1032217060

- Научиться оформлять отчёты с помощью легковесного языка разметки Markdown.

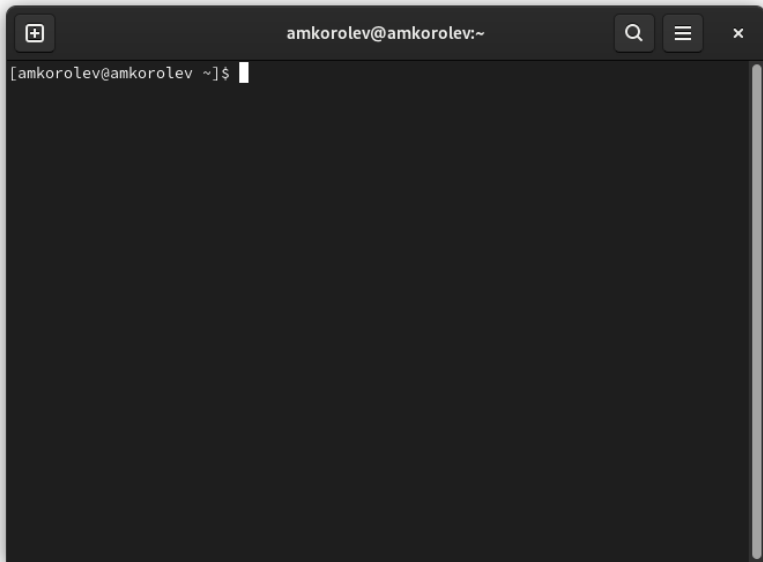
- Сделайте отчёт по предыдущей лабораторной работе в формате Markdown.
- В качестве отчёта предоставить отчёты в 3 форматах: pdf, docx и md

Markdown – это облегченный язык текстовой разметки документов. Его придумали в 2004 году блогер Джон Грубер и интернет-активист Аарон Шварц, чтобы быстро форматировать статьи. Требования к языку были следующими: - Чем проще — тем лучше. - Документы с этой разметкой можно перевести в красиво отформатированный вид, как на веб-странице. - Исходный текст материала должен оставаться читаемым даже без преобразования в веб-страницу.

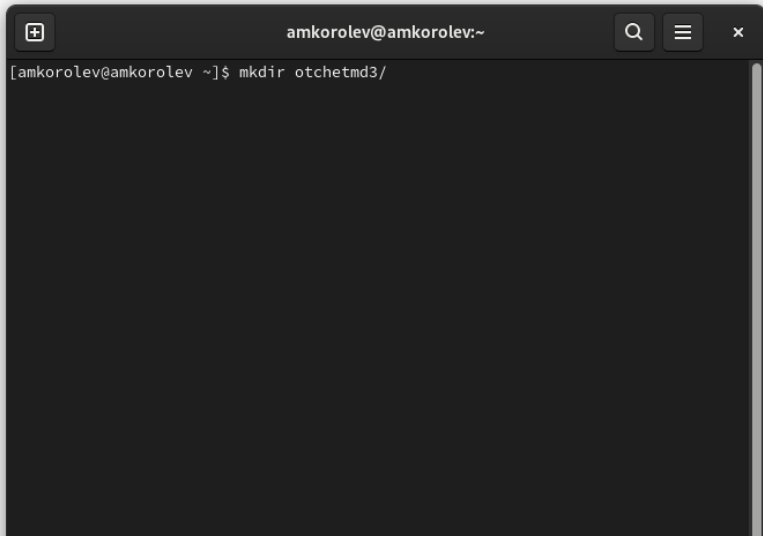
Выполнение лабораторной работы:

---

## Необходимо запустить терминал Linux

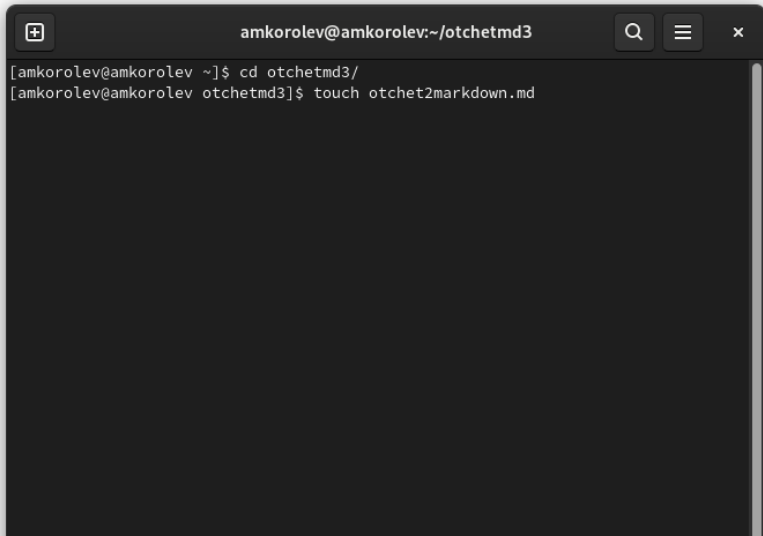


Создать папку, в которой в дальнейшем будет создан файл формата .md, а также изображения.

A terminal window with a dark background. The title bar at the top shows the username 'amkorolev@amkorolev:~' and standard window controls (search, menu, close). The terminal content shows a shell prompt '[amkorolev@amkorolev ~]\$' followed by the command 'mkdir otchetmd3/'.

```
amkorolev@amkorolev:~  
[amkorolev@amkorolev ~]$ mkdir otchetmd3/
```

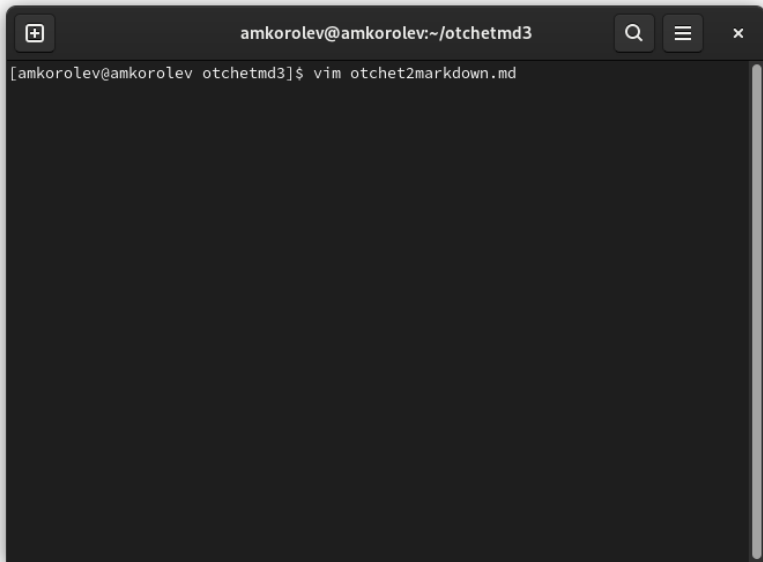
Перейдем в созданную папку. В ней создадим файл формата .md, в котором мы будем работать.

A terminal window with a dark background. The title bar shows the user 'amkorolev' at host 'amkorolev' in the directory '~/otchetmd3'. There are search, menu, and close buttons on the right. The terminal shows two commands: 'cd otchetmd3/' and 'touch otchet2markdown.md'.

```
amkorolev@amkorolev:~/otchetmd3
[amkorolev@amkorolev ~]$ cd otchetmd3/
[amkorolev@amkorolev otchetmd3]$ touch otchet2markdown.md
```

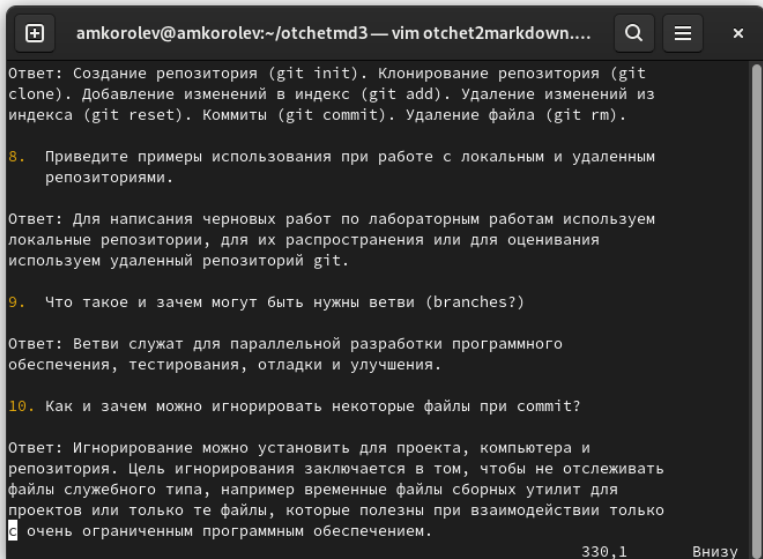


Используя редактор vim, отредактируем файл формата .md.

A terminal window with a dark background. The title bar at the top shows the user 'amkorolev' at host 'amkorolev' in the directory '~/otchetmd3'. On the right side of the title bar are icons for search, a menu, and a close button. The terminal text shows the command '[amkorolev@amkorolev otchetmd3]\$ vim otchet2markdown.md' being entered at the prompt. The rest of the terminal area is empty.

```
amkorolev@amkorolev:~/otchetmd3
[amkorolev@amkorolev otchetmd3]$ vim otchet2markdown.md
```

## Отредактируем файл формата .md.



The screenshot shows a vim editor window with the title bar "amkorolev@amkorolev:~/otchetmd3 — vim otchet2markdown....". The window contains a Q&A session about Git. The text is as follows:

```
amkorolev@amkorolev:~/otchetmd3 — vim otchet2markdown....
Ответ: Создание репозитория (git init). Клонирование репозитория (git
clone). Добавление изменений в индекс (git add). Удаление изменений из
индекса (git reset). Коммиты (git commit). Удаление файла (git rm).

8. Приведите примеры использования при работе с локальным и удаленным
репозиториями.

Ответ: Для написания черновых работ по лабораторным работам используем
локальные репозитории, для их распространения или для оценивания
используем удаленный репозиторий git.

9. Что такое и зачем могут быть нужны ветви (branches?)

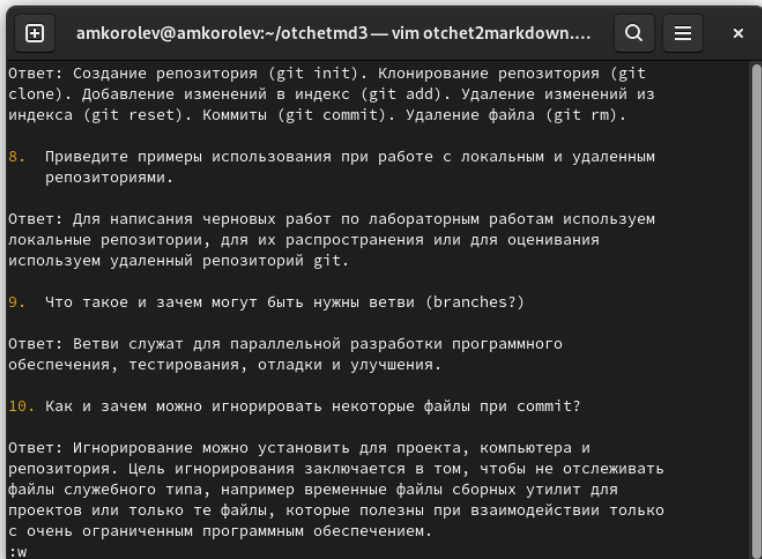
Ответ: Ветви служат для параллельной разработки программного
обеспечения, тестирования, отладки и улучшения.

10. Как и зачем можно игнорировать некоторые файлы при commit?

Ответ: Игнорирование можно установить для проекта, компьютера и
репозитория. Цель игнорирования заключается в том, чтобы не отслеживать
файлы служебного типа, например временные файлы сборных утилит для
проектов или только те файлы, которые полезны при взаимодействии только
с очень ограниченным программным обеспечением.
```

330,1 Внизу

Сохраним файл формата .md, используя команду :w.



```
amkorolev@amkorolev:~/otchetmd3 — vim otchet2markdown....
Ответ: Создание репозитория (git init). Клонирование репозитория (git
clone). Добавление изменений в индекс (git add). Удаление изменений из
индекса (git reset). Коммиты (git commit). Удаление файла (git rm).

8. Приведите примеры использования при работе с локальным и удаленным
репозиториями.

Ответ: Для написания черновых работ по лабораторным работам используем
локальные репозитории, для их распространения или для оценивания
используем удаленный репозиторий git.

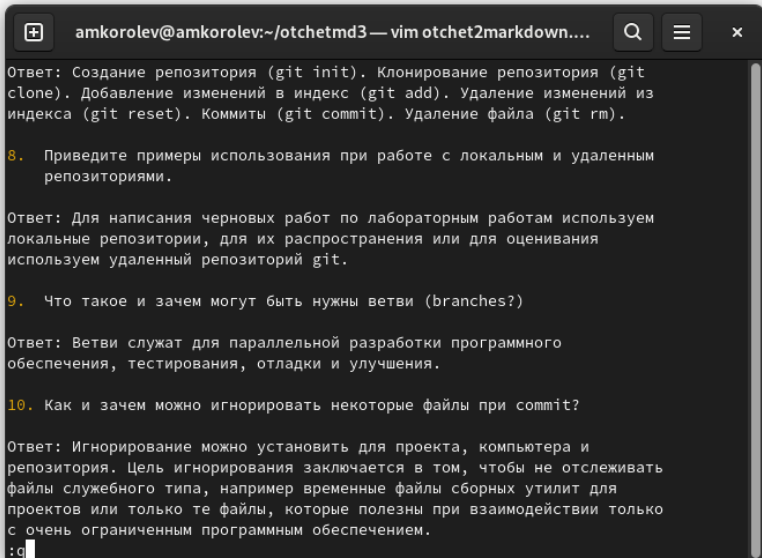
9. Что такое и зачем могут быть нужны ветви (branches?)

Ответ: Ветви служат для параллельной разработки программного
обеспечения, тестирования, отладки и улучшения.

10. Как и зачем можно игнорировать некоторые файлы при commit?

Ответ: Игнорирование можно установить для проекта, компьютера и
репозитория. Цель игнорирования заключается в том, чтобы не отслеживать
файлы служебного типа, например временные файлы сборных утилит для
проектов или только те файлы, которые полезны при взаимодействии только
с очень ограниченным программным обеспечением.
:w
```

## Выйдем из редактора vim, используя команду :q.



```
amkorolev@amkorolev:~/otchetmd3 — vim otchet2markdown....
Ответ: Создание репозитория (git init). Клонирование репозитория (git
clone). Добавление изменений в индекс (git add). Удаление изменений из
индекса (git reset). Коммиты (git commit). Удаление файла (git rm).

8. Приведите примеры использования при работе с локальным и удаленным
репозиториями.

Ответ: Для написания черновых работ по лабораторным работам используем
локальные репозитории, для их распространения или для оценивания
используем удаленный репозиторий git.

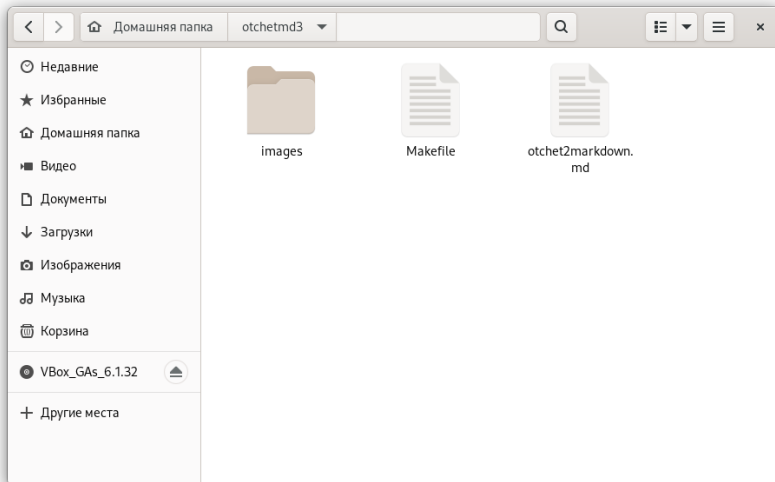
9. Что такое и зачем могут быть нужны ветви (branches?)

Ответ: Ветви служат для параллельной разработки программного
обеспечения, тестирования, отладки и улучшения.

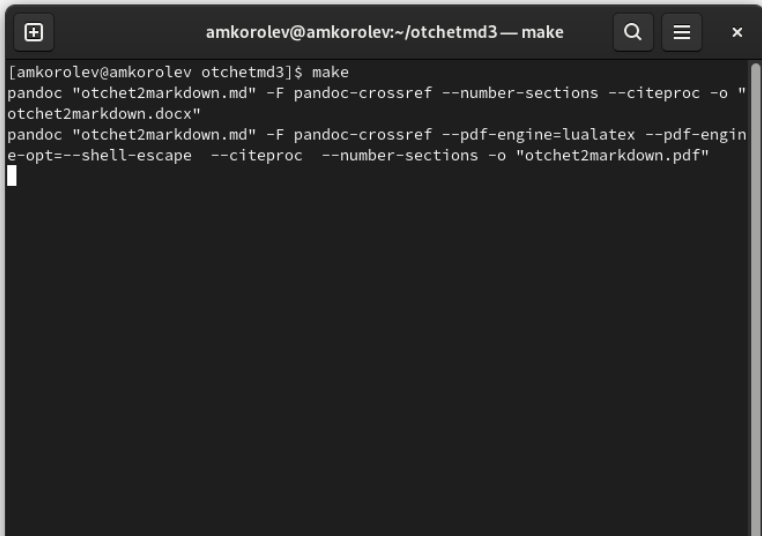
10. Как и зачем можно игнорировать некоторые файлы при commit?

Ответ: Игнорирование можно установить для проекта, компьютера и
репозитория. Цель игнорирования заключается в том, чтобы не отслеживать
файлы служебного типа, например временные файлы сборных утилит для
проектов или только те файлы, которые полезны при взаимодействии только
с очень ограниченным программным обеспечением.
:q
```

Перенесем в папку, где располагается файл формата .md, наши изображения и makefile для создания файлов формата docx. и pdf.



Используя имеющийся Makefile, используя команду make создадим файлы форматов docx. и pdf.



```
amkorolev@amkorolev:~/otchetmd3 — make
[amkorolev@amkorolev otchetmd3]$ make
pandoc "otchet2markdown.md" -F pandoc-crossref --number-sections --citeproc -o "otchet2markdown.docx"
pandoc "otchet2markdown.md" -F pandoc-crossref --pdf-engine=lualatex --pdf-engine-opt=--shell-escape --citeproc --number-sections -o "otchet2markdown.pdf"
```

Проверим, появились ли у нас файлы формата docx. и pdf.

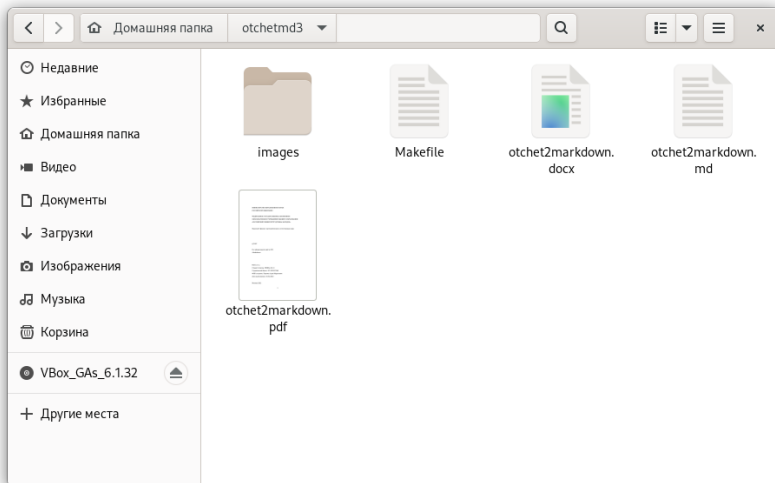


Figure 10: Проверяем наличие файлов

# Проверим полученные файлы на ошибки.

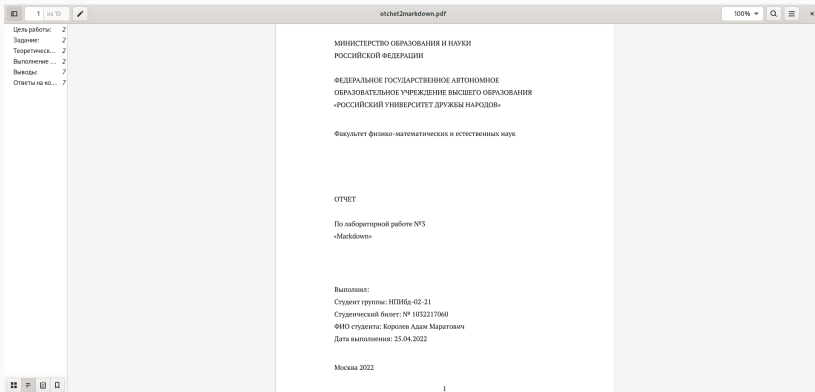


Figure 11: Проверяем работоспособность файлов



## Выводы:

В процессе выполнения задания были приобретены навыки по оформлению отчетов с помощью легковесного языка разметки Markdown.