

Федеральное государственное автономное образовательное учреждение  
высшего образования «Северо-Кавказский федеральный университет»  
Институт цифрового развития

ОТЧЕТ ПО ДИСЦИПЛИНЕ  
ОСНОВЫ ПРОГРАММНОЙ ИНЖЕНЕРИИ

Выполнила:

Мирзаева Камилла Мирзаевна

2 курс, группа ПИЖ-б-о-20-1

Принял:

Воронкин Роман Александрович

Ставрополь, 2021 г.

```
Git CMD

C:\Users\Камилла>cd moon
C:\Users\Камилла\moon>explorer .
C:\Users\Камилла\moon>git tag
C:\Users\Камилла\moon>git tag -a v1.4 -m "my version 1.4"
C:\Users\Камилла\moon>git tag
v1.4
C:\Users\Камилла\moon>git tag -a v1.5 -m "my version 1.5"
C:\Users\Камилла\moon>git tag
v1.4
v1.5
C:\Users\Камилла\moon>git tag -a v1.6 -m "my version 1.6"
C:\Users\Камилла\moon>git tag
v1.4
v1.5
v1.6
C:\Users\Камилла\moon>git push origin --tags
Enumerating objects: 35, done.
Counting objects: 100% (35/35), done.
Delta compression using up to 12 threads
Compressing objects: 100% (34/34), done.
Writing objects: 100% (35/35), 769.85 KiB | 25.66 MiB/s, done.
Total 35 (delta 12), reused 4 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (12/12), done.
To https://github.com/mirzaevakamilla/moon.git
 * [new tag]          v1.4 -> v1.4
 * [new tag]          v1.5 -> v1.5
 * [new tag]          v1.6 -> v1.6

C:\Users\Камилла\moon>
```

Рисунок 1 - скриншот терминала командами Git

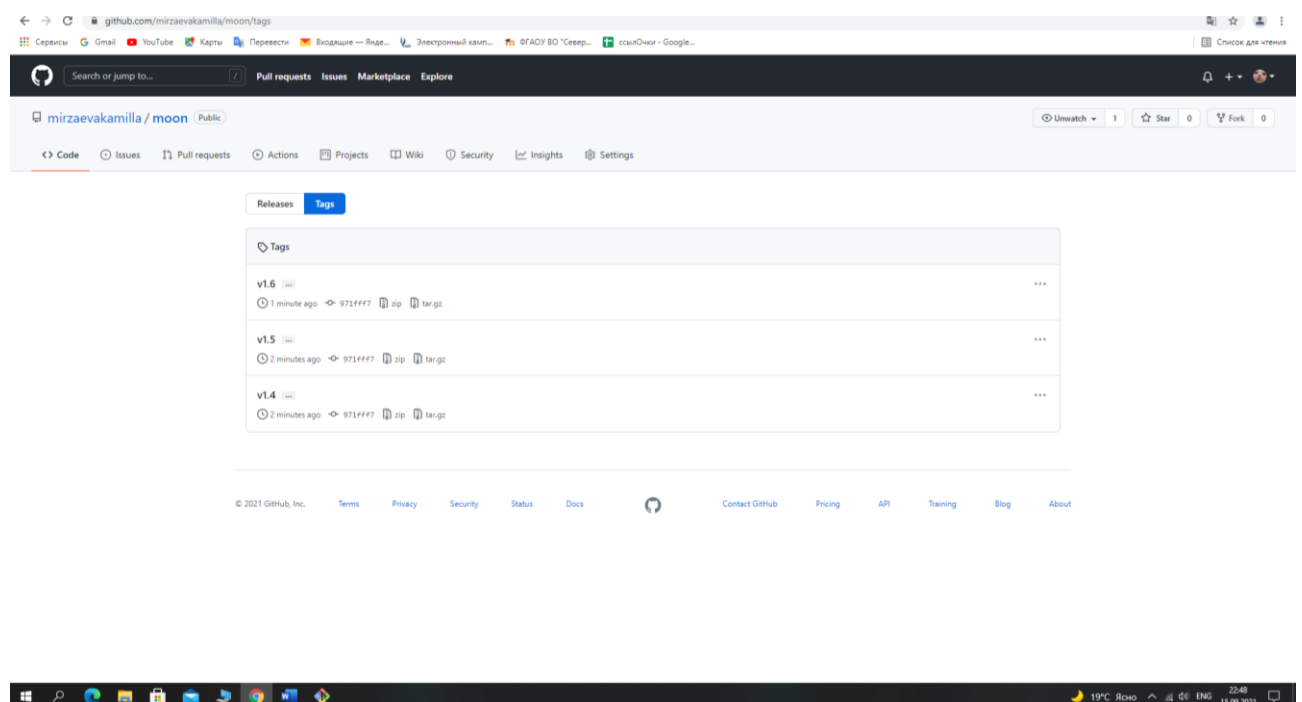


Рисунок 2 - скриншот репозитория после соответствующих изменений

```
Git CMD
C:\Users\Камилла\moon>git push origin --tags
Enumerating objects: 35, done.
Counting objects: 100% (35/35), done.
Delta compression using up to 12 threads
Compressing objects: 100% (34/34), done.
Writing objects: 100% (35/35), 769.85 KiB | 25.66 MiB/s, done.
Total 35 (delta 12), reused 4 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (12/12), done.
To https://github.com/mirzaevakamilla/moon.git
 * [new tag]          v1.4 -> v1.4
 * [new tag]          v1.5 -> v1.5
 * [new tag]          v1.6 -> v1.6

C:\Users\Камилла\moon>git log --pretty=format:"%h %s" --graph
* 971fff7 добавление файла
* 40ff82b change
* d5af1bf change
* c8ceafa изменение ридми
* dbd71d9 change readme
* 37870f8 change evrything
* 20e4cdf change
* 91115ff изменения
* db311c9 change
* 6672718 Initial commit

C:\Users\Камилла\moon>
```

Рисунок 3 - скриншот терминала командами Git

```
Git CMD - "C:\Program Files\Git\cmd\git.exe" show HEAD
C:\Users\Камилла\moon>git show HEAD
commit 971fff77cebfd52f743d1bae241abe794d6fb8d (HEAD -> main, tag: v1.6, tag: v1.5, tag: v1.4, origin/main, origin/HEAD)
Author: mirzaevakamilla <mirzaeva.ncfu>
Date: Thu Sep 9 20:07:43 2021 +0300

    добавление файла

diff --git "a/\320\273\320\260\320\261\320\260 1.pdf" "b/\320\273\320\260\320\261\320\260 1.pdf"
new file mode 100644
index 0000000..8d5dd87
--- /dev/null
+++ "b/\320\273\320\260\320\261\320\260 1.pdf"
@@ -0,0 +1,158 @@
+Федеральное государственное автономное образовательное учреждение
+высшего образования «Северо-Кавказский федеральный университет»
+
+Институт цифрового развития
+
+ОТЧЕТ ПО ДИСЦИПЛИНЕ
+ОСНОВЫ ПРОГРАММНОЙ ИНЖЕНЕРИИ
+
+Выполнила:
+Мирзаева Камила Мирзаевна
+2 курс, группа ПИЖ-б-о-20-1
+
+Принял:
+Воронкин Роман Александрович
+
+Ставрополь, 2021 г.
+...skipping...
+commit 971fff77cebfd52f743d1bae241abe794d6fb8d (HEAD -> main, tag: v1.6, tag: v1.5, tag: v1.4, origin/main, origin/HEAD)
+Author: mirzaevakamilla <mirzaeva.ncfu>
+Date: Thu Sep 9 20:07:43 2021 +0300
+
+    добавление файла
+
+diff --git "a/\320\273\320\260\320\261\320\260 1.pdf" "b/\320\273\320\260\320\261\320\260 1.pdf"
+new file mode 100644
+index 0000000..8d5dd87
+--- /dev/null
++++ "b/\320\273\320\260\320\261\320\260 1.pdf"
+@@ -0,0 +1,158 @@
+Федеральное государственное автономное образовательное учреждение
+высшего образования «Северо-Кавказский федеральный университет»
+
+Институт цифрового развития
+
+ОТЧЕТ ПО ДИСЦИПЛИНЕ
+ОСНОВЫ ПРОГРАММНОЙ ИНЖЕНЕРИИ
+
+Выполнила:
+Мирзаева Камила Мирзаевна
+2 курс, группа ПИЖ-б-о-20-1
+
+Принял:
+Воронкин Роман Александрович
+
+Ставрополь, 2021 г.
+Рисунок 1- скриншот терминала командами Git
+Рисунок 2 - скриншот репозитория после соответствующих изменений
+Рисунок 3 - скриншот терминала командами Git
+Рисунок 4 - скриншот репозитория после соответствующих изменений
+Рисунок 5 - скриншот терминала командами Git
+Рисунок 6 - скриншот репозитория после соответствующих изменений
+Рисунок 7 - скриншот терминала командами Git
+Рисунок 8 - скриншот репозитория после соответствующих изменений
+Рисунок 9 - скриншот терминала командами Git
+Рисунок 10 - скриншот репозитория после соответствующих изменений
+Рисунок 11 - скриншот терминала командами Git
+Рисунок 12 - скриншот репозитория после соответствующих изменений
+
+Ответы на вопросы:
```

Рисунок 4 -скриншот терминала командами Git

```
Git CMD

C:\Users\Камилла>cd moon

C:\Users\Камилла\moon>git show HEAD~1
commit 40ff82b07401ea5b0202dfbaf1ab38c9485842ab
Author: mirzaevakamilla <mirzaeva.ncfu>
Date: Thu Sep 9 19:23:57 2021 +0300

    change

diff --git a/README.md b/README.md
index 6765d68..f373fdf 100644
--- a/README.md
+++ b/README.md
@@ -1,2 +1,2 @@
 # moon
->>> Третий уровень цитирования
+>>> Третий уровень цитирования.

C:\Users\Камилла\moon>
```

Рисунок 5 - скриншот терминала командами Git

```
Git CMD

+++ b/README.md
@@ -1,2 +1,2 @@
 # moon
->>> Третий уровень цитирования
+>>> Третий уровень цитирования.

C:\Users\Камилла\moon>git show c8ceafa
commit c8ceafac8498f4148ea190569f505699c7a8edf2
Author: mirzaevakamilla <mirzaeva.ncfu>
Date: Tue Sep 7 22:42:26 2021 +0300

    изменение ридми

diff --git a/README.md b/README.md
index 986a57b..974941b 100644
--- a/README.md
+++ b/README.md
@@ -1,3 +1,2 @@
 # moon
-### Заголовок третьего уровня
-> Первый уровень цитирования

C:\Users\Камилла\moon>
```

Рисунок 6 - скриншот терминала командами Git

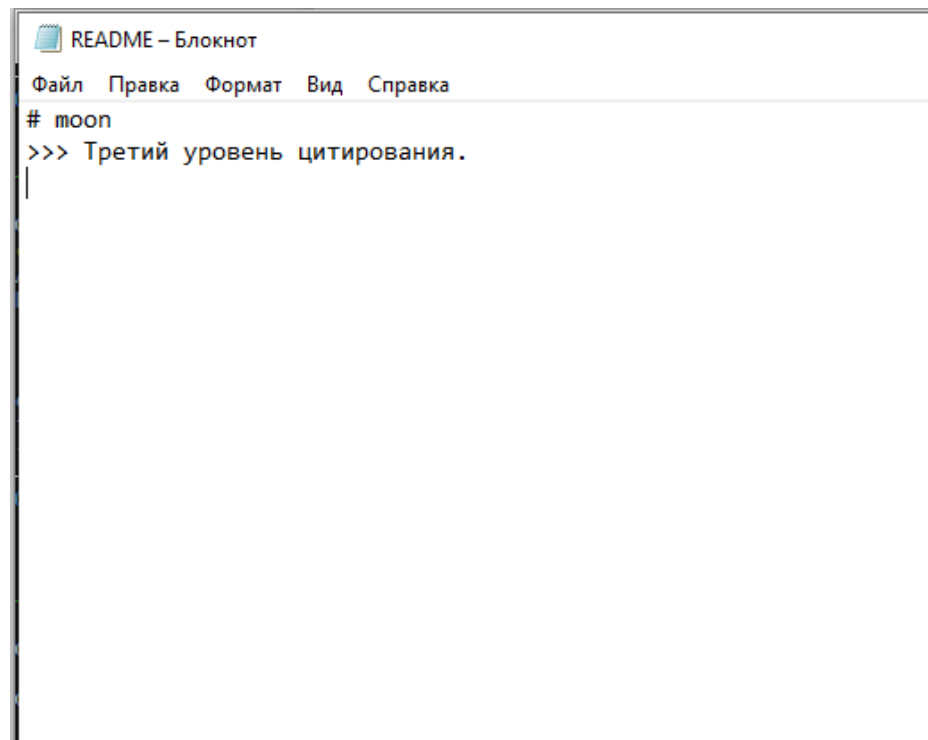


Рисунок 7 - скриншот файла

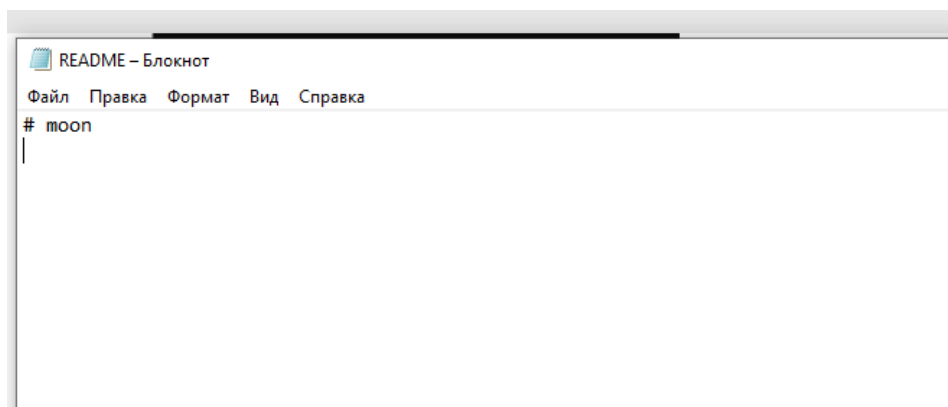


Рисунок 8 - скриншот файла

```
Git CMD

изменение ридми
diff --git a/README.md b/README.md
index 986a57b..974941b 100644
--- a/README.md
+++ b/README.md
@@ -1,3 +1,2 @@
-# moon
-### Заголовок третьего уровня
+> Первый уровень цитирования

C:\Users\Камилла\moon>explorer .
C:\Users\Камилла\moon>git checkout -- README.md
C:\Users\Камилла\moon>git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

C:\Users\Камилла\moon>explorer .
C:\Users\Камилла\moon>
```

Рисунок 9 - скриншот терминала командами Git

```
Git CMD

+++ b/README.md
@@ -1,3 +1,2 @@
-# moon
-### Заголовок третьего уровня
+> Первый уровень цитирования

C:\Users\Камилла\moon>explorer .
C:\Users\Камилла\moon>git checkout -- README.md
C:\Users\Камилла\moon>git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

C:\Users\Камилла\moon>explorer .
C:\Users\Камилла\moon>git reset --hard HEAD~1
HEAD is now at 40ff82b change
C:\Users\Камилла\moon>
```

Рисунок 10 - скриншот терминала командами Git

```
. If you believe this repository exists and you have access, make sure you're au
thenticated.
fatal: repository 'https://bitbucket.org/kamilla156/pops+refs/remotes/origin/*:r
efs/heads/*/' not found

C:\Users\student-09-525\Desktop\moon>git push --prune https://kanilla156@bitbuck
et.org/kamilla156/pops/src+refs/remotes/origin/*:refs/heads/* +refs/tags/*:refs/
tags/*
Enumerating objects: 35, done.
Counting objects: 100% (35/35), done.
Delta compression using up to 2 threads
Compressing objects: 100% (22/22), done.
Writing objects: 100% (35/35), 769.85 KiB | 109.98 MiB/s, done.
Total 35 (delta 12), reused 35 (delta 12), pack-reused 0
To https://bitbucket.org/kamilla156/pops/src+refs/remotes/origin/*:refs/heads/*
 * [new tag]      v1.4 -> v1.4
 * [new tag]      v1.5 -> v1.5
 * [new tag]      v1.6 -> v1.6

C:\Users\student-09-525\Desktop\moon>
```

Рисунок 11 - скриншот терминала командами Git

Вопросы для защиты:

## 1. Как выполнить историю коммитов в Git? Какие существуют дополнительные опции для просмотра истории коммитов?

Одним из основных и наиболее мощных инструментов для этого является команда `git log`.

Опции:

Одним из самых полезных аргументов является `-p` или `--patch`, который показывает разницу (выводит патч), внесенную в каждый коммит. Так же вы можете ограничить количество записей в выводе команды; Так же есть возможность использовать серию опций для обобщения. Например, если вы хотите увидеть сокращенную статистику для каждого коммита, вы можете использовать опцию `--stat`. Следующей действительно полезной опцией является `--pretty`. Эта опция меняет формат вывода. Существует несколько встроенных вариантов отображения. Опция `oneline` выводит каждый коммит в одну строку, что может быть очень удобным если вы просматриваете большое количество коммитов. К тому же, опции `short`, `full` и `fuller` делают вывод приблизительно в том же формате, но с меньшим или большим количеством информации соответственно. Наиболее интересной опцией является `format`, которая позволяет указать формат для вывода информации. Полезные опции для `git log --pretty=format` отображает наиболее полезные опции для изменения формата. Опции `oneline` и `format` являются особенно полезными с опцией `--graph` команды `log`. С этой опцией вы сможете увидеть небольшой граф в формате ASCII, который показывает текущую ветку и историю слияний.

## 2. Как ограничить вывод при просмотре истории коммитов?

Опция	Описание
<code>-(n)</code>	Показывает только последние n коммитов.
<code>--since, --after</code>	Показывает только те коммиты, которые были сделаны после указанной даты.
<code>--until, --before</code>	Показывает только те коммиты, которые были сделаны до указанной даты.
<code>--author</code>	Показывает только те коммиты, в которых запись author совпадает с указанной строкой.
<code>--committer</code>	Показывает только те коммиты, в которых запись committer совпадает с указанной строкой.
<code>--grep</code>	Показывает только коммиты, сообщение которых содержит указанную строку.
<code>-S</code>	Показывает только коммиты, в которых изменение в коде повлекло за собой добавление или удаление указанной строки.

### 3. Как внести изменения в уже сделанный коммит?

Если вы хотите переделать коммит - внесите необходимые изменения, добавьте их в индекс и сделайте коммит ещё раз, указав параметр `--amend` : Эта команда использует область подготовки (индекс) для внесения правок в коммит. Если вы ничего не меняли с момента последнего коммита (например, команда запущена сразу после предыдущего коммита), то снимок состояния останется в точности таким же, а всё что вы сможете изменить — это ваше сообщение к коммиту

### 4. Как отменить индексацию файла в Git?

Используйте `git reset HEAD ...` для исключения из индекса.

### 5. Как отменить изменения в файле?

`git checkout --` опасная команда. Все локальные изменения в файле пропадут — Git просто заменит его версией из последнего коммита.

### 6. Что такое удаленный репозиторий Git?

Удалённые репозитории представляют собой версии вашего проекта, сохранённые в интернете или ещё где-то в сети.

### 7. Как выполнить просмотр удаленных репозиториях данного локального репозитория?



Для того, чтобы просмотреть список настроенных удалённых репозиториях, вы можете запустить команду `git remote`. Она выведет названия доступных удалённых репозиториях.

8. Как добавить удаленный репозиторий для данного локального репозитория?

Для того, чтобы добавить удалённый репозиторий и присвоить ему имя (shortname), просто выполните команду `git remote add <shortname> <url>`

9. Как выполнить отправку/получение изменений с удаленного репозитория?

Когда вы хотите поделиться своими наработками, вам необходимо отправить их в удалённый репозиторий. Команда для этого действия простая: `git push <remote-name> <branch-name>`.

10. Как выполнить просмотр удаленного репозитория?

Если хотите получить побольше информации об одном из удалённых репозиториях, вы можете использовать команду `git remote show <remote>`.

11. Каково назначение тэгов Git?

Как и большинство СКВ, Git имеет возможность пометить определённые моменты в истории как важные. Такие пометки в Git называются тегами.

12. Как осуществляется работа с тэгами Git?

Просмотреть список имеющихся тегов в Git можно очень просто. Достаточно набрать команду `git tag` (параметры `-l` и `--list` опциональны): Данная команда перечисляет теги в алфавитном порядке; порядок их отображения не имеет существенного значения. Так же можно выполнить поиск тега по шаблону. Создание тегов Git использует два основных типа тегов: легковесные и аннотированные. Легковесный тег — это что-то очень похожее на ветку, которая не изменяется — просто указатель на определённый коммит. А вот аннотированные теги хранятся в базе данных Git как полноценные объекты. Обычно рекомендуется создавать аннотированные теги, чтобы иметь всю перечисленную информацию; но если вы хотите сделать временную метку или по какой-то причине не хотите сохранять остальную информацию, то для этого годятся и легковесные.

Создание аннотированного тега в Git. Самый простой способ — это указать - а при выполнении команды `tag`: Опция `-m` задаёт сообщение, которое будет храниться вместе с тегом. Если не указать сообщение, то Git запустит редактор, чтобы вы смогли его ввести. С помощью команды `git show` вы можете посмотреть данные тега вместе с коммитом.

Обмен тегами. Процесс аналогичен отправке веток — достаточно выполнить команду `git push origin <tagname>`. Если у вас много тегов, и вам хотелось бы

отправить все за один раз, то можно использовать опцию `--tags` для команды `git push`. В таком случае все ваши теги отправятся на удалённый сервер (если только их уже там нет).

Удаление тегов. Для удаления тега в локальном репозитории достаточно выполнить команду `git tag -d`.

Переход на тег. Если вы хотите получить версии файлов, на которые указывает тег, то вы можете сделать `git checkout` для тега.

13. Самостоятельно изучите назначение флага `--prune` в командах `git fetch` и `git push`. Каково назначение этого флага?

Для создания зеркала необходимо вначале выполнить команду получения всех изменений с репозитория GitHub