

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №2
дисциплины «Основы программной инженерии»

Выполнил:
Мелтонян Одиссей Гарикович
2 курс, группа ПИЖ-б-о-22-1,
09.03.04 «Программная инженерия»,
очная форма обучения

(подпись)

Проверил: Воронкин Р.А.

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Тема: «Исследование возможностей Git для работы с локальными репозиториями»

Цель работы: исследовать базовые возможности системы контроля версий Git для работы с локальными репозиториями.

Ход работы:

1. Изучил теоретический материал.
2. Создал общедоступный репозиторий на GitHub, в котором была использована лицензия MIT и выбранный язык программирования.

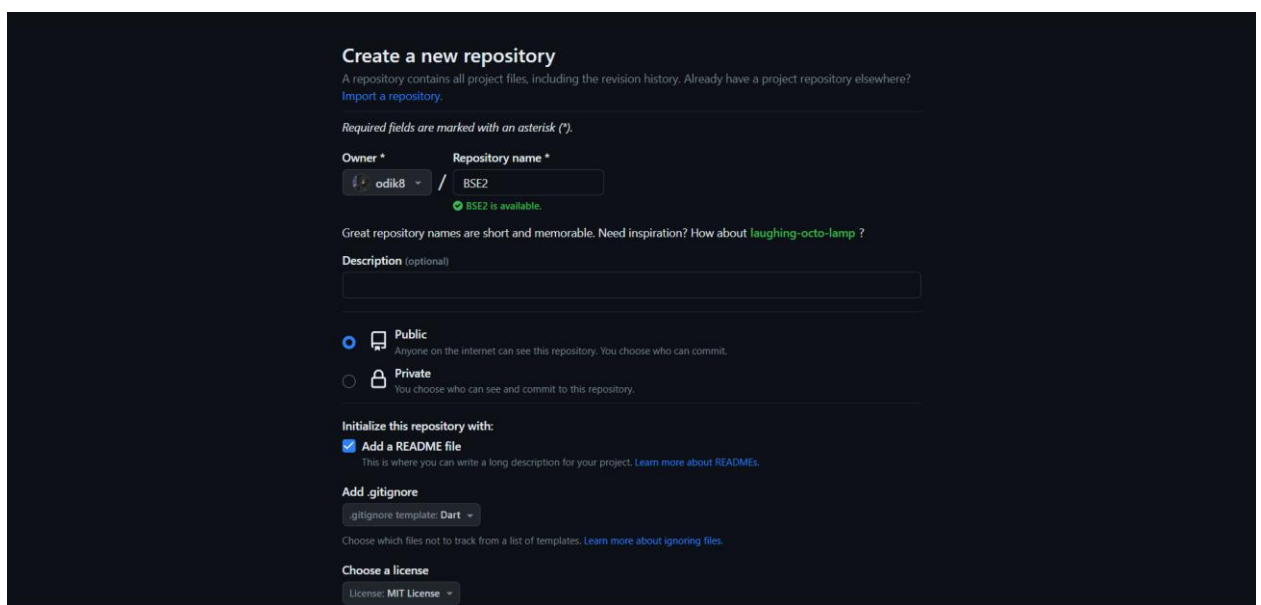


Рисунок 1. – Страница создания репозитория

3. Выполнил клонирование созданного репозитория на рабочий компьютер

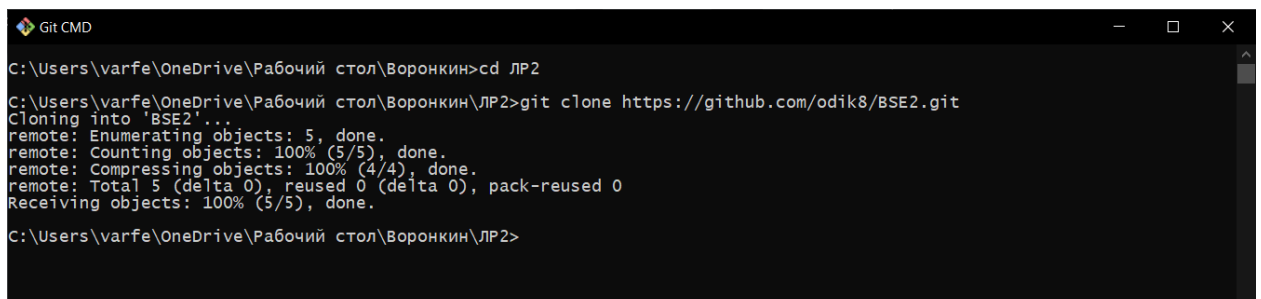
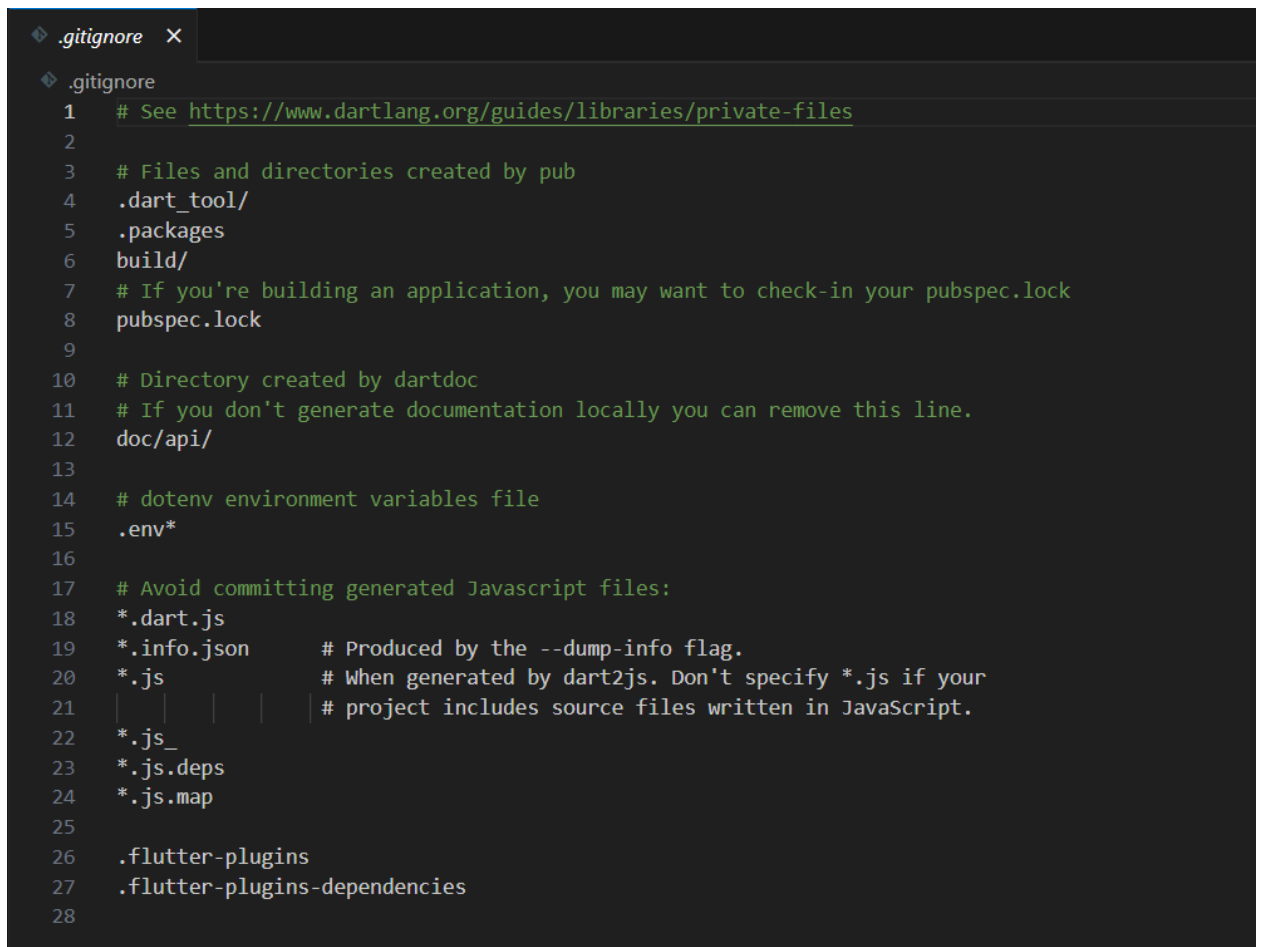


Рисунок 2. – Окно клонирования репозитория

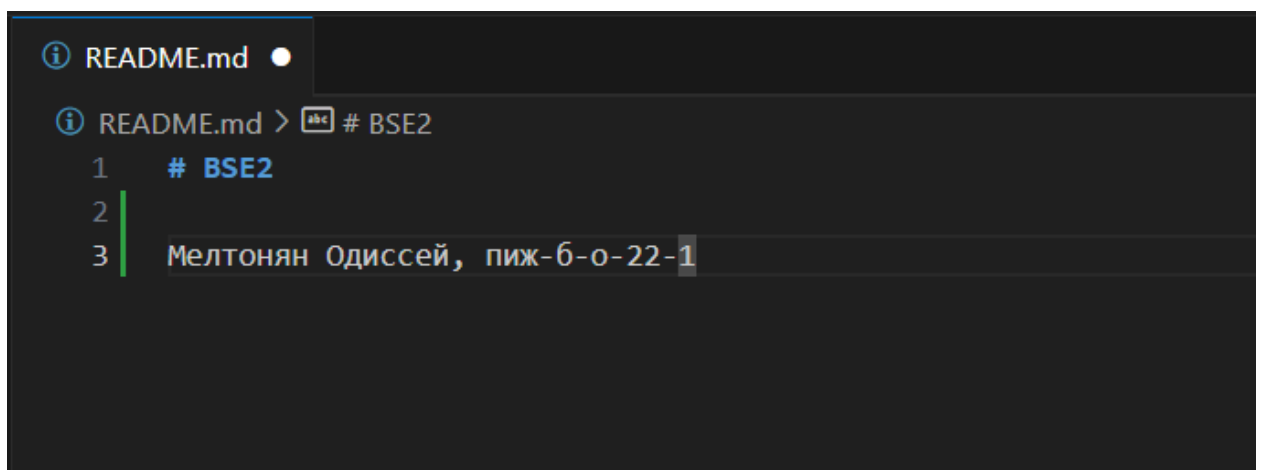
4. Дополнил файл .gitignore необходимыми правилами для выбранного языка программирования и интегрированной среды разработки.

A screenshot of a code editor showing the contents of a .gitignore file. The file is titled ".gitignore" with a close button. The content includes comments and file/directory patterns to be ignored. Line numbers 1 through 28 are visible on the left. The patterns include directories like .dart_tool/, .packages, build/, doc/api/, .env*, and Flutter-related files. It also includes wildcard patterns for generated JavaScript files like *.dart.js, *.info.json, *.js, *.js_, *.js.deps, and *.js.map.

```
1 # See https://www.dartlang.org/guides/libraries/private-files
2
3 # Files and directories created by pub
4 .dart_tool/
5 .packages
6 build/
7 # If you're building an application, you may want to check-in your pubspec.lock
8 pubspec.lock
9
10 # Directory created by dartdoc
11 # If you don't generate documentation locally you can remove this line.
12 doc/api/
13
14 # dotenv environment variables file
15 .env*
16
17 # Avoid committing generated Javascript files:
18 *.dart.js
19 *.info.json      # Produced by the --dump-info flag.
20 *.js             # When generated by dart2js. Don't specify *.js if your
21 |               | # project includes source files written in JavaScript.
22 *.js_
23 *.js.deps
24 *.js.map
25
26 .flutter-plugins
27 .flutter-plugins-dependencies
28
```

Рисунок 3. – Файл .gitignore

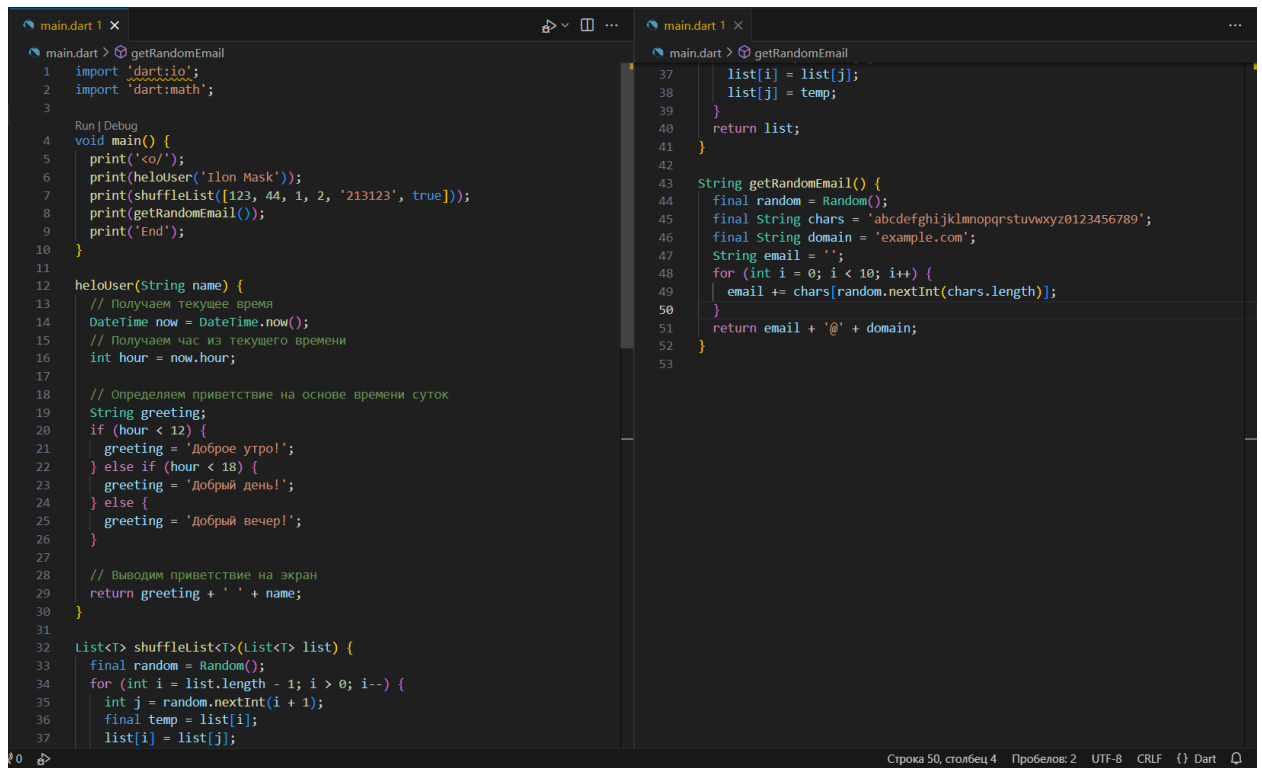
5. Добавил в файл README.md информацию о группе и ФИО студента, выполняющего лабораторную работу.

A screenshot of a code editor showing the contents of a README.md file. The file is titled "README.md" with an information icon and a close button. The content starts with a heading "# BSE2" followed by a line with the student's name and ID. Line numbers 1, 2, and 3 are visible on the left. The text is in a light blue color on a dark background.

```
1 # BSE2
2
3 Мелтонян Одиссей, пж-б-о-22-1
```

Рисунок 4. – Файл README.md

6. Написал небольшую программу на выбранном языке программирования. Зафиксировал изменения при написании программы в локальном репозитории. Было сделано 7 коммитов и использовано 3 тэга.



```
main.dart 1 x
main.dart > getRandomEmail
1 import 'dart:io';
2 import 'dart:math';
3
4 Run | Debug
5 void main() {
6   print('co/');
7   print(heloUser('Ilon Mask'));
8   print(shuffleList([123, 44, 1, 2, '213123', true]));
9   print(getRandomEmail());
10  print('End');
11 }
12
13 heloUser(String name) {
14   // Получаем текущее время
15   DateTime now = DateTime.now();
16   // Получаем час из текущего времени
17   int hour = now.hour;
18
19   // Определяем приветствие на основе времени суток
20   String greeting;
21   if (hour < 12) {
22     greeting = 'Доброе утро!';
23   } else if (hour < 18) {
24     greeting = 'Добрый день!';
25   } else {
26     greeting = 'Добрый вечер!';
27   }
28
29   // Выводим приветствие на экран
30   return greeting + ' ' + name;
31 }
32
33 List<T> shuffleList<T>(List<T> list) {
34   final random = Random();
35   for (int i = list.length - 1; i > 0; i--) {
36     int j = random.nextInt(i + 1);
37     final temp = list[i];
38     list[i] = list[j];
39     list[j] = temp;
40   }
41   return list;
42 }
43
44 String getRandomEmail() {
45   final random = Random();
46   final String chars = 'abcdefghijklmnopqrstuvwxyz0123456789';
47   final String domain = 'example.com';
48   String email = '';
49   for (int i = 0; i < 10; i++) {
50     email += chars[random.nextInt(chars.length)];
51   }
52   return email + '@' + domain;
53 }
```

Строка 50, столбец 4 Пробелов: 2 UTF-8 CRLF {} Dart

Рисунок 5. Файл main.dart

7. Просмотрел историю (журнал) хранилища командой git log.

```

Git CMD
C:\Users\varfe\OneDrive\Рабочий стол\Воронкин\ЛР2\BSE2>git log
commit 7c9b734a605b87469f7418427566cd14d9b86fee (HEAD -> main, tag: v3, tag: v2, tag: v1.0, tag: v1)
Author: Odyssey Meltonyan <odissey.meltonyan@gmail.com>
Date: Sun Sep 24 12:24:56 2023 +0300

    Commit 7

commit 830303b0a8890de6058a1a90cafa44260f3fe6ed
Author: Odyssey Meltonyan <odissey.meltonyan@gmail.com>
Date: Sun Sep 24 12:24:04 2023 +0300

    Commit 6

commit 55d560bee9190bfdcc50f9b4608a2f5bd869056
Author: Odyssey Meltonyan <odissey.meltonyan@gmail.com>
Date: Sun Sep 24 12:23:02 2023 +0300

    Commit 5

commit 405d8cf6d40d94ca74a17c2cbc171493694d8c05
Author: Odyssey Meltonyan <odissey.meltonyan@gmail.com>
Date: Sun Sep 24 12:18:57 2023 +0300

    Commit 4

commit d5390b80d4cd3ea9b3a1e197a566a97e433b7eb2 (tag: v1.4)
Author: Odyssey Meltonyan <odissey.meltonyan@gmail.com>
Date: Sun Sep 24 12:08:21 2023 +0300

    Commit 3

commit 29cb78ac612ad5304ab7f3787aa8581717f3cdfd (tag: v1.3)
Author: Odyssey Meltonyan <odissey.meltonyan@gmail.com>
Date: Sun Sep 24 11:39:32 2023 +0300

    Commit 2

commit cc04a6fbd01e8b4eb890716d32ac60b820658cc5 (tag: v1.2)
Author: Odyssey Meltonyan <odissey.meltonyan@gmail.com>
Date: Sun Sep 24 11:35:36 2023 +0300

    Commit 1

commit 19fc501fd2a7d113597b6774dfb9b1d54c40d868 (origin/main, origin/HEAD)
Author: odik8 <107572344+odik8@users.noreply.github.com>
Date: Sun Sep 24 11:22:47 2023 +0300

    Initial commit
C:\Users\varfe\OneDrive\Рабочий стол\Воронкин\ЛР2\BSE2>

```

Рисунок 6. – Результат команды git log

8. Просмотреть содержимое коммитов командой git show, где:
HEAD : последний коммит;
HEAD~1 : предпоследний коммит (и т. д.);
b34a0e : коммит с указанным хэшем.

```
Git CMD
C:\Users\varfe\OneDrive\Рабочий стол\Воронкин\ЛР2\BSE2>git show HEAD
commit 7c9b734a605b87469f7418427566cd14d9b86fee (HEAD -> main, tag: v3, tag: v2, tag: v1.0, tag: v1)
Author: Odyssey Meltonyan <odissey.meltonyan@gmail.com>
Date: Sun Sep 24 12:24:56 2023 +0300

    Commit 7

diff --git a/main.dart b/main.dart
index 76f5e96..d0094e2 100644
--- a/main.dart
+++ b/main.dart
@@ -6,6 +6,7 @@ void main() {
    print(heloUser('Ilon Mask'));
    print(shuffleList([123, 44, 1, 2, '213123', true]));
    print(getRandomEmail());
+   print('End');
}

heloUser(String name) {
C:\Users\varfe\OneDrive\Рабочий стол\Воронкин\ЛР2\BSE2>
```

Рисунок 7. – Результат команды git show HEAD

```
Git CMD
C:\Users\varfe\OneDrive\Рабочий стол\Воронкин\ЛР2\BSE2>git show HEAD~1
commit 830303b0a8890de6058a1a90cafa44260f3fe6ed
Author: Odyssey Meltonyan <odissey.meltonyan@gmail.com>
Date: Sun Sep 24 12:24:04 2023 +0300

    Commit 6

diff --git a/main.dart b/main.dart
index 326e6a5..76f5e96 100644
--- a/main.dart
+++ b/main.dart
@@ -5,6 +5,7 @@ void main() {
    print('<o/>');
    print(heloUser('Ilon Mask'));
    print(shuffleList([123, 44, 1, 2, '213123', true]));
+   print(getRandomEmail());
}

heloUser(String name) {
@@ -37,3 +38,14 @@ List<T> shuffleList<T>(List<T> list) {
    }
    return list;
}

+String getRandomEmail() {
+   final random = Random();
+   final String chars = 'abcdefghijklmnopqrstuvwxyz0123456789';
+   final String domain = 'example.com';
+   String email = '';
+   for (int i = 0; i < 10; i++) {
+       email += chars[random.nextInt(chars.length)];
+   }
+   return email + '@' + domain;
+}
+}
```

Рисунок 8. – Результат команды git show HEAD~1

```
Git CMD
C:\Users\varfe\OneDrive\Рабочий стол\Воронкин\ЛР2\BSE2>git show 19fc501fd2a7d113597b6774dfb9b1d54c40d868
commit 19fc501fd2a7d113597b6774dfb9b1d54c40d868 (origin/main, origin/HEAD)
Author: odik8 <107572344+odik8@users.noreply.github.com>
Date: Sun Sep 24 11:22:47 2023 +0300

    Initial commit

diff --git a/.gitignore b/.gitignore
new file mode 100644
index 0000000..3a83c2f
--- /dev/null
+++ b/.gitignore
@@ -0,0 +1,27 @@
+# See https://www.dartlang.org/guides/libraries/private-files
+
+# Files and directories created by pub
+.dart_tool/
+.packages
+build/
+# If you're building an application, you may want to check-in your pubspec.lock
+pubspec.lock
+
+# Directory created by dartdoc
+# If you don't generate documentation locally you can remove this line.
+doc/api/
+
+# dotenv environment variables file
+.env*
```

Рисунок 9. – Результат команды git show <хэш>

9. Освоил возможность отката к заданной версии.

9.1. Удалил весь код из файла main.dart, и сохранил этот файл.

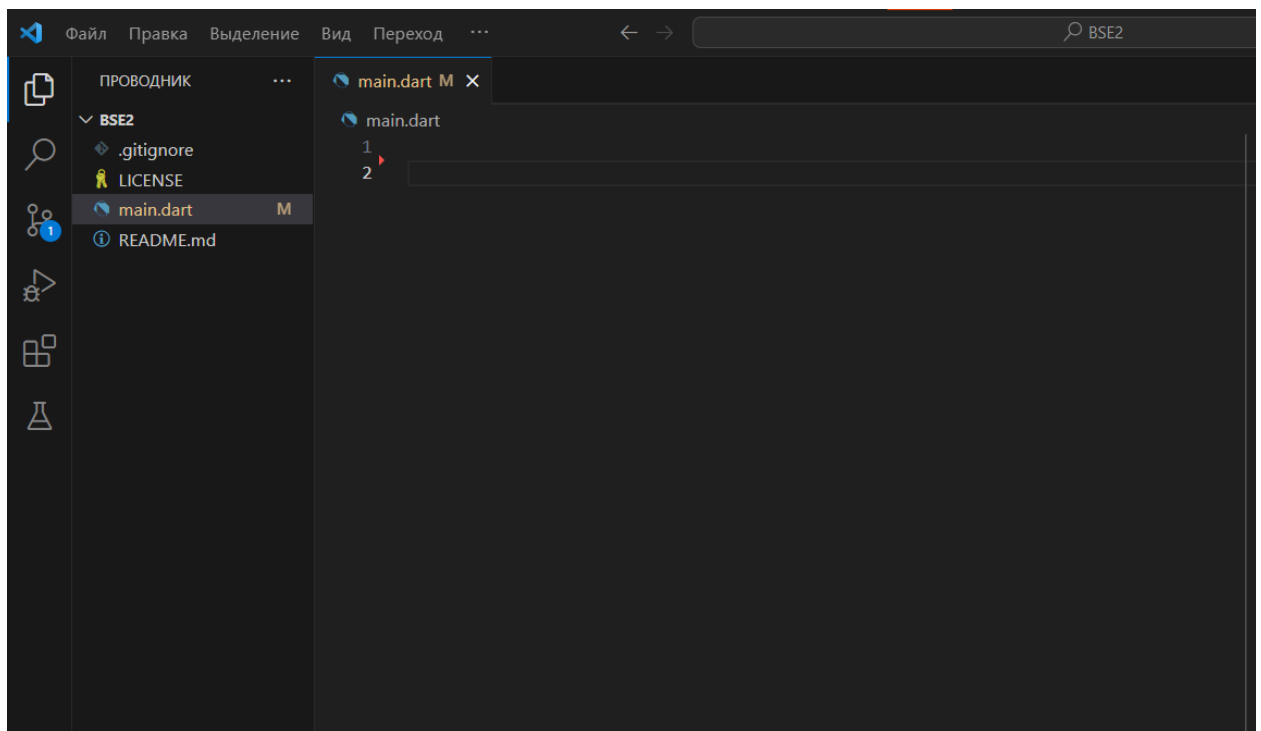
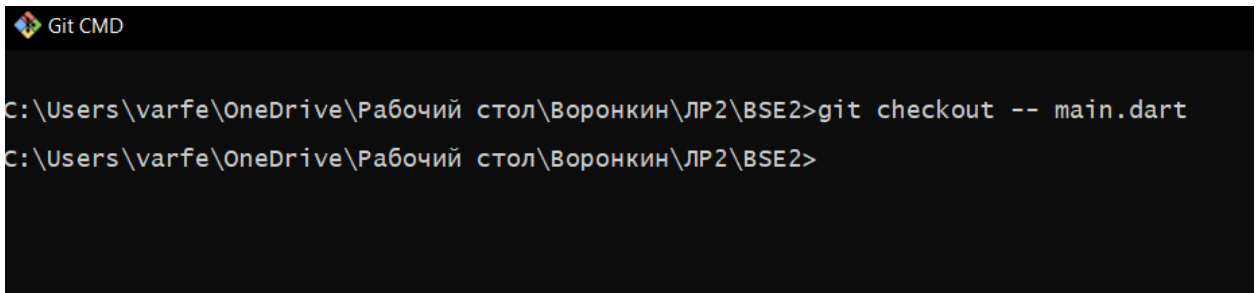


Рисунок 10. – Пустой файл main.dart

9.2. Удалил все несохраненные изменения в файле командой: git checkout -- main.dart.

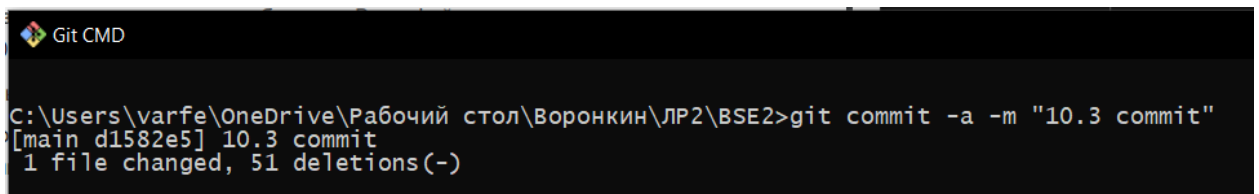


```
Git CMD

C:\Users\varfe\OneDrive\Рабочий стол\Воронкин\ЛР2\BSE2>git checkout -- main.dart
C:\Users\varfe\OneDrive\Рабочий стол\Воронкин\ЛР2\BSE2>
```

Рисунок 11. – Результат выполнения команды `git checkout -- main.dart`

9.3 Повторил пункт 10.1 и сделал коммит



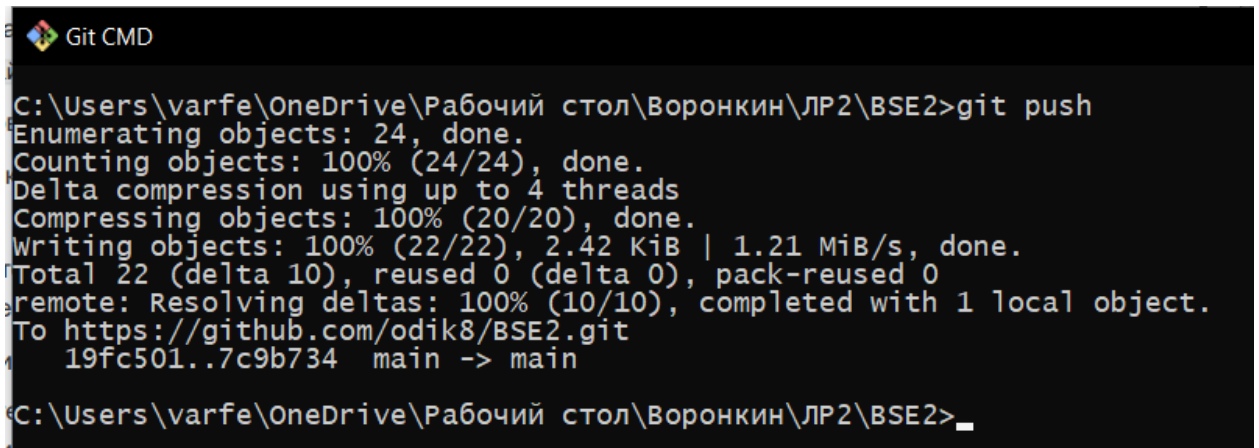
```
Git CMD

C:\Users\varfe\OneDrive\Рабочий стол\Воронкин\ЛР2\BSE2>git commit -a -m "10.3 commit"
[main d1582e5] 10.3 commit
1 file changed, 51 deletions(-)
```

Рисунок 12. – Результат выполнения команды `git commit -a -m "10.3 commit"`

9.4 Откатить состояние хранилища к предыдущей версии командой: `git reset --hard HEAD~1`.

10. Зафиксировал, оправил и проконтролировал изменения.



```
Git CMD

C:\Users\varfe\OneDrive\Рабочий стол\Воронкин\ЛР2\BSE2>git push
Enumerating objects: 24, done.
Counting objects: 100% (24/24), done.
Delta compression using up to 4 threads
Compressing objects: 100% (20/20), done.
Writing objects: 100% (22/22), 2.42 KiB | 1.21 MiB/s, done.
Total 22 (delta 10), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (10/10), completed with 1 local object.
To https://github.com/odik8/BSE2.git
   19fc501..7c9b734  main -> main
C:\Users\varfe\OneDrive\Рабочий стол\Воронкин\ЛР2\BSE2>
```

Рисунок 13. – Результат выполнения команды `git push`

Вывод: исследованы базовые возможности системы контроля версий Git для работы с локальными репозиториями.

Вопросы для защиты:

1. Как выполнить историю коммитов в Git? Какие существуют дополнительные опции для просмотра истории коммитов? – Для просмотра истории коммитов в Git используйте команду "git log". Дополнительные опции позволяют настроить вывод истории, такие как "--oneline" для краткого формата, "--author" для фильтрации по автору и другие.

2. Как ограничить вывод при просмотре истории коммитов? – Вы можете ограничить вывод истории коммитов с помощью опции "--since" для отображения коммитов, сделанных с определенной даты, или "--until" для отображения коммитов до определенной даты. Также можно указать определенный файл для просмотра истории только этого файла.

3. Как внести изменения в уже сделанный коммит? – Для внесения изменений в уже сделанный коммит, используйте команду "git commit --amend". Это позволит внести изменения и переписать последний коммит.

4. Как отменить индексацию файла в Git? – Чтобы отменить индексацию файла в Git, используйте команду "git reset <файл>". Это уберет файл из индекса без удаления его изменений.

5. Как отменить изменения в файле? – Для отмены изменений в файле, которые еще не были закоммичены, используйте "git checkout --<файл>". Если изменения уже закоммичены, можно вернуться к предыдущему коммиту с помощью "git reset HEAD <файл>".

6. Что такое удаленный репозиторий Git? – Удаленный репозиторий Git - это репозиторий, который располагается на удаленном сервере и используется для совместной работы над проектом. Он позволяет разработчикам обмениваться изменениями и синхронизировать свои локальные репозитории с общим кодом.

7. Как выполнить просмотр удаленных репозиториях данного локального репозитория? – Вы можете просмотреть удаленные репозитории,

связанные с вашим локальным репозиторием, с помощью команды "git remote -v".

8. Как добавить удаленный репозиторий для данного локального репозитория? – Для добавления удаленного репозитория используйте команду "git remote add <имя> <URL>", где <имя> - это имя удаленного репозитория, а <URL> - его адрес.

9. Как выполнить отправку/получение изменений с удаленного репозитория? – Для отправки изменений на удаленный репозиторий используйте "git push". Для получения изменений с удаленного репозитория - "git pull".

10. Как выполнить просмотр удаленного репозитория? – Для просмотра удаленного репозитория, вы можете перейти к его URL-адресу в веб-браузере и посмотреть содержимое.

11. Каково назначение тэгов Git? – Тэги Git используются для маркировки определенных коммитов как важных точек в истории репозитория, таких как релизы или версии. Они облегчают управление версиями кода.

12. Как осуществляется работа с тэгами Git? – Для создания тэга используйте "git tag <имя_тэга>". Для просмотра существующих тэгов - "git tag". Чтобы поделиться тэгами с удаленным репозиторием, используйте "git push --tags".

13. Самостоятельно изучите назначение флага --prune в командах git fetch и git push. Каково назначение этого флага? – Флаг "--prune" в командах "git fetch" и "git push" используется для удаления удаленных веток или ссылок, которые больше не существуют на удаленном репозитории. Это помогает поддерживать локальное окружение в актуальном состоянии и избегать мусора в ветках.