

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Кафедра «Информационная безопасность систем и технологий»

Отчет

о практической работе №1

на тему «Системы контроля версий и совместной разработки» и «Управление
сборкой программных проектов»

Дисциплина: ТиМП

Группа: 22ПТ2

Выполнил студент: _____ Яблоновский И.М.

Количество баллов:

Дата сдачи:

Принял: к.т.н., доцент Лупанов М.Ю.

2023

- 1 Цель работы: изучить основы git и создание Makefile
- 2 Задания к лабораторной работе:
 - 2.1 Создать свой аккаунт на github.com.
 - 2.2 Клонировать учебный репозиторий в свой аккаунт на github.com.
 - 2.3 Сделать локальную копию репозитория на своем компьютере.
 - 2.4 Внести изменения в Readme.md на локальном компьютере, сделать коммит, и синхронизировать локальный репозиторий с сетевым
 - 2.5 Ознакомиться с примером проекта example-1 и исправить цель clean в make-файле, чтобы она корректно работала, т. е. удаляла все файлы, созданные в процессе сборки. Проверьте корректность работы цели clean.
 - 2.6 Создать в каталоге файл .gitignore для игнорирования в коммитах исполняемых и прочих файлов, создаваемых процессе сборки.
 - 2.7 Сделать все необходимые для выполнения п.п. 5 и 6 коммиты, и синхронизировать локальный репозиторий с сетевым.
 - 2.8 Ознакомиться с примером example-2 и добавить в make-файле переменную для имени файла с исходным кодом. Проверьте корректность его работы для всех целей.
 - 2.9 Создать в каталоге файл .gitignore для игнорирования в коммитах исполняемых и прочих файлов, создаваемых процессе сборки.
 - 2.10 Сделать все необходимые для выполнения п.п. 8 и 9 коммиты, и синхронизировать локальный репозиторий с сетевым.
 - 2.11 Ознакомиться с примером example-3 и создать для него make-файл. За основу можно взять make-файл из примера example-2. Проверьте корректность его работы для всех целей.
 - 2.12 Создать в каталоге файл .gitignore для игнорирования в коммитах исполняемых и прочих файлов, создаваемых процессе сборки.
 - 2.13 Сделать все необходимые для выполнения п.п. 11 и 12 коммиты, и синхронизировать локальный репозиторий с сетевым.
 - 2.13 Сделать отчет о выполненной работе, включив в него ссылку на свой репозиторий на сайте github.com

3 Результат работы:

3.1 Был создан свой аккаунт на github.com.

3.2 Учебный репозиторий <https://github.com/MikeLupanov/StudyMake>.
git клонирован в свой аккаунт на github.com. Результат представлен на рисунке 1.

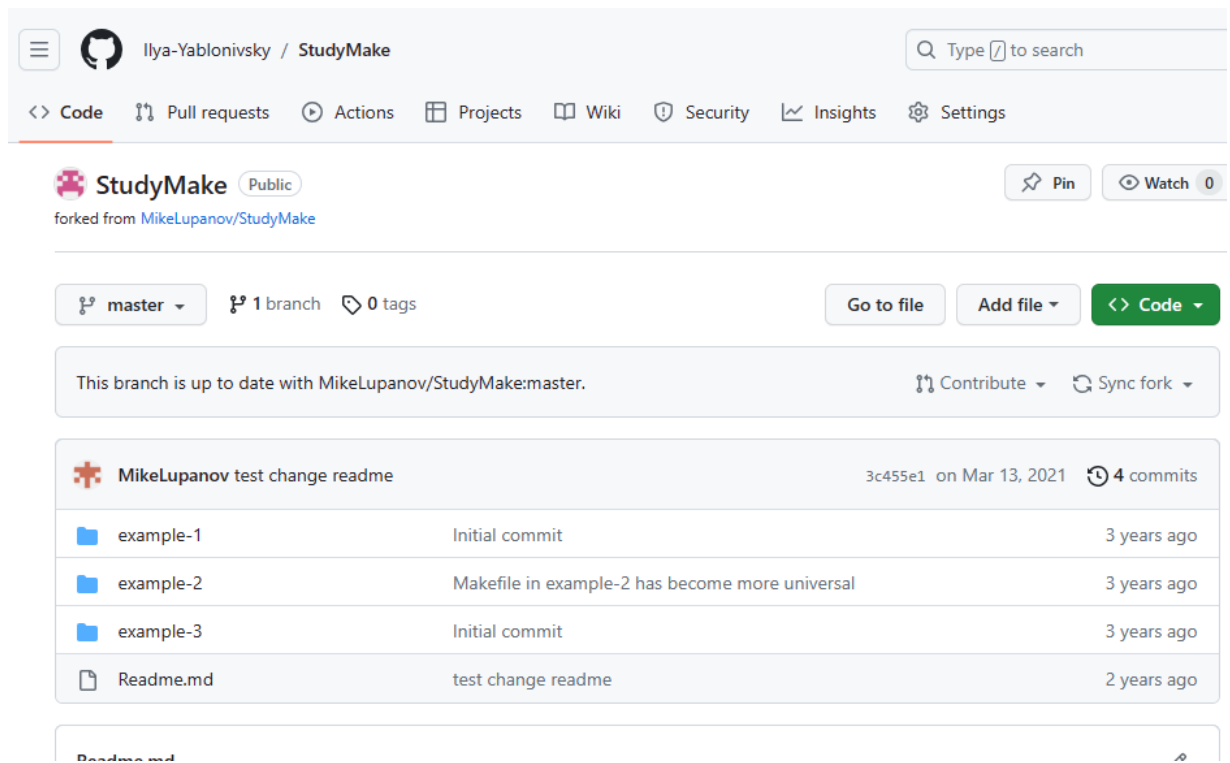


Рисунок 1 - Клонированный репозиторий

3.3 Сделана локальная копия репозитория на компьютер. Результат представлен на рисунке 2.

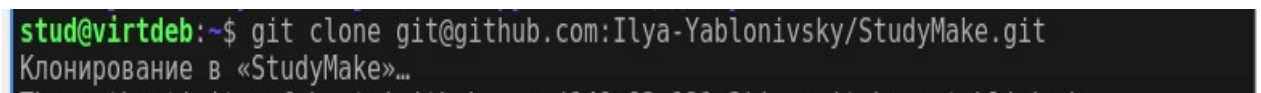
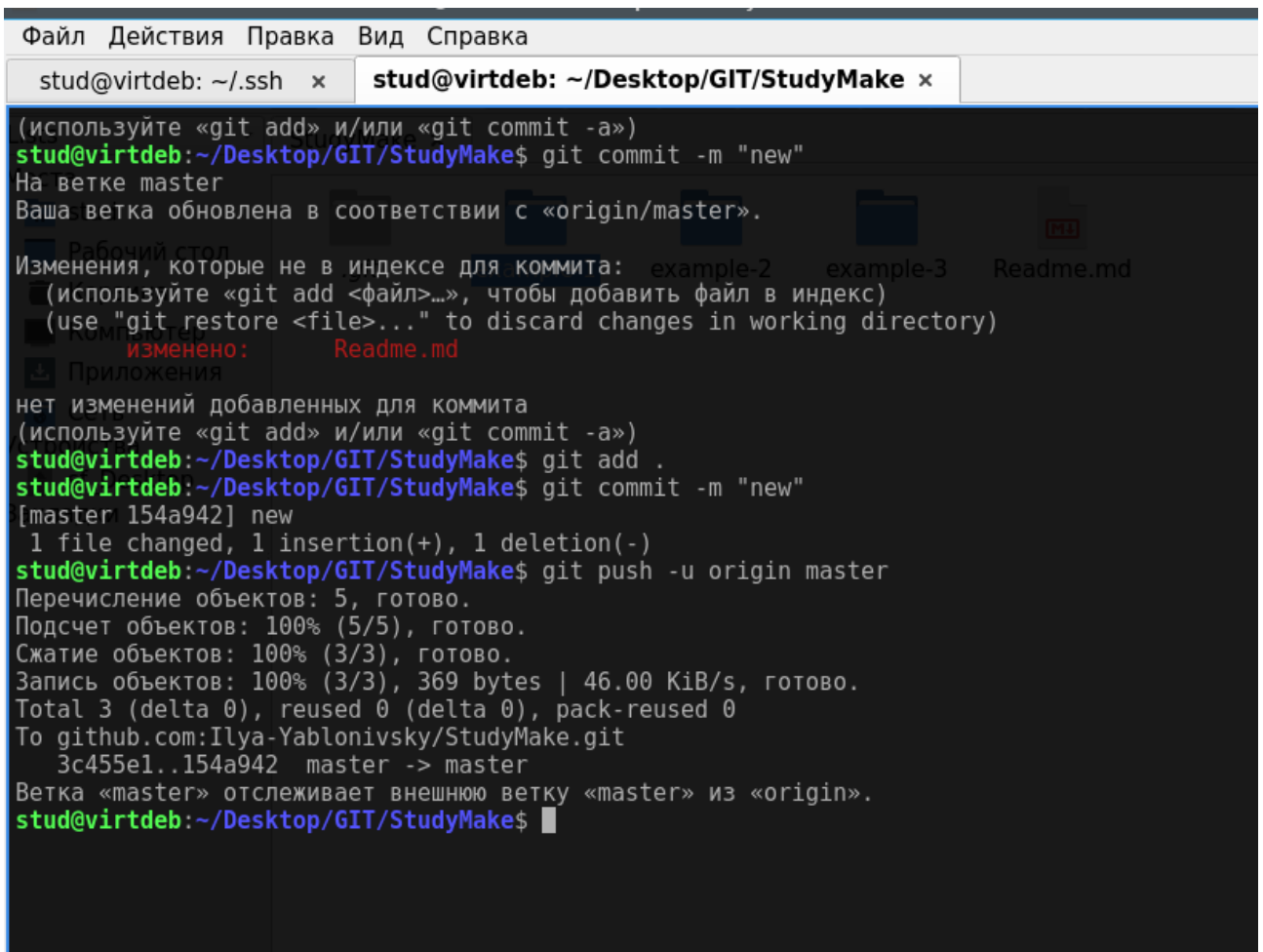


Рисунок 2 - Локальная копия репозитория

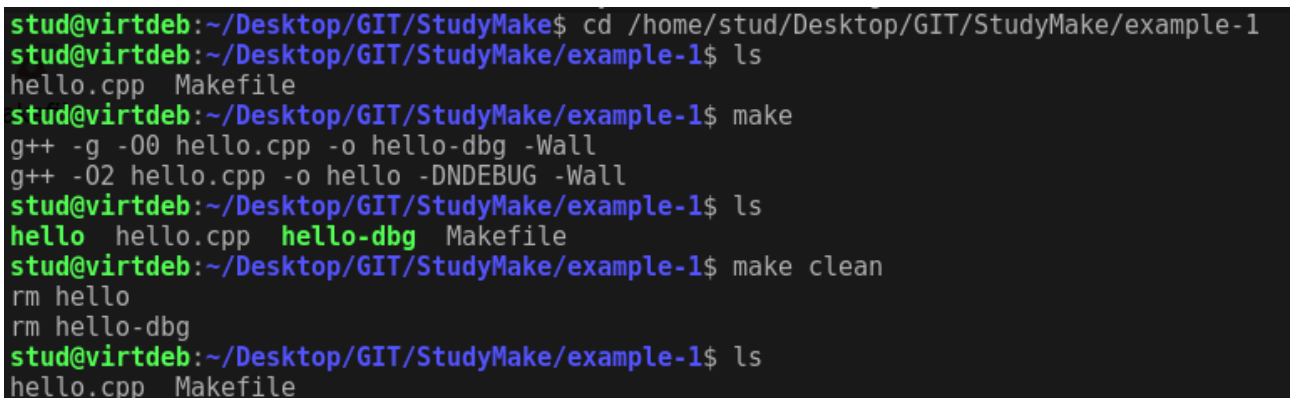
3.4 Внесены изменения в Readme.md на локальном компьютере, сделан коммит, и синхронизирован локальный репозиторий с сетевым. Результат представлен на рисунках 3,4.



```
Файл Действия Правка Вид Справка
stud@virtdeb: ~/.ssh x stud@virtdeb: ~/Desktop/GIT/StudyMake x
(используйте «git add» и/или «git commit -a»)
stud@virtdeb:~/Desktop/GIT/StudyMake$ git commit -m "new"
На ветке master
Ваша ветка обновлена в соответствии с «origin/master».
Изменения, которые не в индексе для коммита: example-2 example-3 Readme.md
(используйте «git add <файл>...», чтобы добавить файл в индекс)
(use "git restore <file>..." to discard changes in working directory)
изменено: Readme.md
нет изменений добавленных для коммита
(используйте «git add» и/или «git commit -a»)
stud@virtdeb:~/Desktop/GIT/StudyMake$ git add .
stud@virtdeb:~/Desktop/GIT/StudyMake$ git commit -m "new"
[master 154a942] new
1 file changed, 1 insertion(+), 1 deletion(-)
stud@virtdeb:~/Desktop/GIT/StudyMake$ git push -u origin master
Перечисление объектов: 5, готово.
Подсчет объектов: 100% (5/5), готово.
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 369 bytes | 46.00 KiB/s, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:Ilya-Yablonivsky/StudyMake.git
3c455e1..154a942 master -> master
Ветка «master» отслеживает внешнюю ветку «master» из «origin».
stud@virtdeb:~/Desktop/GIT/StudyMake$
```

Рисунок 3 - Исправление файла Readme

3.5 Был исправлен example-1. Цель clean в make-файле, корректно работает. Проверена работа цели clean. Результат представлен на рисунках 4,5.



```
stud@virtdeb:~/Desktop/GIT/StudyMake$ cd /home/stud/Desktop/GIT/StudyMake/example-1
stud@virtdeb:~/Desktop/GIT/StudyMake/example-1$ ls
hello.cpp Makefile
stud@virtdeb:~/Desktop/GIT/StudyMake/example-1$ make
g++ -g -O0 hello.cpp -o hello-dbg -Wall
g++ -O2 hello.cpp -o hello -DNDEBUG -Wall
stud@virtdeb:~/Desktop/GIT/StudyMake/example-1$ ls
hello hello.cpp hello-dbg Makefile
stud@virtdeb:~/Desktop/GIT/StudyMake/example-1$ make clean
rm hello
rm hello-dbg
stud@virtdeb:~/Desktop/GIT/StudyMake/example-1$ ls
hello.cpp Makefile
```

Рисунок 4 - Проверка clean

```
clean:
    rm hello
    rm hello-dbg
```

Рисунок 5 - Внесенные исправления

3.6 В example-2 изменен make-файл, добавлена переменная для имени файла с исходным кодом. Проверена работа. Результат представлен на рисунках 6,7

```
stud@virtdeb:~/Desktop/GIT/StudyMake$ cd example-2
stud@virtdeb:~/Desktop/GIT/StudyMake/example-2$ ls
hello.cpp  Makefile
stud@virtdeb:~/Desktop/GIT/StudyMake/example-2$ make
g++ -g -O0 -Wall hello.cpp -o hello-dbg
g++ -O2 -Wall -DNDEBUG hello.cpp -o hello
stud@virtdeb:~/Desktop/GIT/StudyMake/example-2$ ls
hello  hello.cpp  hello-dbg  Makefile
stud@virtdeb:~/Desktop/GIT/StudyMake/example-2$ make clean
rm -f hello hello-dbg
stud@virtdeb:~/Desktop/GIT/StudyMake/example-2$ ls
hello.cpp  Makefile
```

Рисунок 6 - Проверка работы

```
RELEASE_FLAGS = -O2 -Wall -DNDEBUG
DEBUG_FLAGS   = -g -O0 -Wall
RELEASE_EXEC   = hello
DEBUG_EXEC     = hello-dbg
SOURCE        = hello.cpp

all: debug release

debug: $(DEBUG_EXEC)

$(DEBUG_EXEC): $(SOURCE)
    g++ $(DEBUG_FLAGS) $(SOURCE) -o $(DEBUG_EXEC)

release: $(RELEASE_EXEC)

$(RELEASE_EXEC): $(SOURCE)
    g++ $(RELEASE_FLAGS) $(SOURCE) -o $(RELEASE_EXEC)
```

Рисунок 7 - Внесенные изменения

3.7 В example-3 создан make-файл. За основу взят make-файл из примера example-2. Проверена его работа. Результат представлен на рисунках 8, 9.

```
stud@virtdeb:~/Desktop/GIT/StudyMake$ cd example-3
stud@virtdeb:~/Desktop/GIT/StudyMake/example-3$ ls
Kazino-Royal.cpp  Makefile
stud@virtdeb:~/Desktop/GIT/StudyMake/example-3$ make
g++ -g -O0 -Wall Kazino-Royal.cpp -o hello-dbg
g++ -O2 -Wall -DNDEBUG Kazino-Royal.cpp -o hello
stud@virtdeb:~/Desktop/GIT/StudyMake/example-3$ ./hello
Добро пожаловать в Казино-Рояль
Сыграем в кости?
Ты загадываешь число - я бросаю кость.
Если ты угадал с трех попыток - ты выиграл
Нет - проиграл, и настало время отчислений
введите число от 1 до 6 : 3
Выпало число 4
введите число от 1 до 6 : 5
Выпало число 2
введите число от 1 до 6 : 1
Выпало число 1
Ты читеп!!!
stud@virtdeb:~/Desktop/GIT/StudyMake/example-3$ ls
hello  hello-dbg  Kazino-Royal.cpp  Makefile
stud@virtdeb:~/Desktop/GIT/StudyMake/example-3$ make clean
rm -f hello hello-dbg
stud@virtdeb:~/Desktop/GIT/StudyMake/example-3$ ls
Kazino-Royal.cpp  Makefile
```

Рисунок 8 - Проверка работы

```
.PHONY: all debug release clean

RELEASE_FLAGS = -O2 -Wall -DNDEBUG
DEBUG_FLAGS   = -g -O0 -Wall
RELEASE_EXEC  = hello
DEBUG_EXEC    = hello-dbg
SOURCE        = Kazino-Royal.cpp

all: debug release

debug: $(DEBUG_EXEC)

$(DEBUG_EXEC): $(SOURCE)
    g++ $(DEBUG_FLAGS) $(SOURCE) -o $(DEBUG_EXEC)
```

Рисунок 9 - Внесенные изменения

3.8 Создан в каталоге файл .gitignore для игнорирования в коммитах файлов, создаваемых процессе сборки. Результат представлен на рисунке 10.

A screenshot of a text editor window showing the content of a .gitignore file. The window title is ".gitignore x". The content is as follows:

```
*  
!hello  
!hello-dbg  
!.gitignore
```

Рисунок 10 - Содержание файла gitignore

3.9 [Ilya-Yablonivsky/StudyMake \(github.com\)](https://github.com/Ilya-Yablonivsky/StudyMake)

4 Вывод: изучены принципы работы с Git.