Вводим команду

u211-05@PROG-02 MINGW32 ~/Desktop/lab02

$ ls

В каталоге lab02 пусто, поэтому ls ничего не выведет.

Создаем папки Алисы и Боба командой Mkdir

u211-05@PROG-02 MINGW32 ~/Desktop/lab02

$ mkdir Alice

u211-05@PROG-02 MINGW32 ~/Desktop/lab02

$ mkdir Bob

Переходим в каталог Алисы

u211-05@PROG-02 MINGW32 ~/Desktop/lab02

$ cd Alice/

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice

Создаем каталог Project и переходим в него.

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice

$ mkdir Project

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice

$ cd Project/

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project

Инициализируем репозитарий в данном каталоге

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project

$ git init

Initialized empty Git repository in C:/Users/u211-05/Desktop/lab02/Alice/Project

/.git/

Настроим репозитарий Алисы, чтобы коммиты были от ее имени:

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git config user.name 'Alice (KhlopovIK)'

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git config user.name 'example@example.uk'

Запустив CB, создаем проект в репозитарии Алисы и собираем его.

Вернувшись в Git Bash проверяем состояние рабочей копии командой git status

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git status

On branch master

//в данный момент мы находимся в ветке master

No commits yet

// пока не было коммитов

Untracked files:

(use "git add ..." to include in what will be committed)

bin/

main.cpp

obj/

project.cbp

project.depend

//Список неотслеженных файлов и папок

nothing added to commit but untracked files present (use "git add" to track)

//чтобы начать их отслеживать воспользуйтесь командой add

Добавляем файл main.cpp командой:

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git add main.cpp

Опять проверяем статус

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached ..." to unstage)

new file: main.cpp

Untracked files:

(use "git add ..." to include in what will be committed)

bin/

obj/

project.cbp

project.depend

Как видим файл добавился и программа об этом сообщила

Выполним коммит нового файла

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git commit -m "code: заготовка программы"

[master (root-commit) d0bd57a] code: заготовка программы

1 file changed, 9 insertions(+)

create mode 100644 main.cpp

Также добавим в индекс файл проекта и коммитнем его

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git add project.cbp

warning: LF will be replaced by CRLF in project.cbp.

The file will have its original line endings in your working directory

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git commit -m "build: Добавлен файл проекта"

[master de484e5] build: Добавлен файл проекта

1 file changed, 40 insertions(+)

create mode 100644 project.cbp

### Создание коммитов с изменениями

Заменим тело функции main() на ввод двух чисел:

cout << "Enter A and B: ";

int a, b;

cin >> a >> b;

После этого шага опять проверяем статус

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git status

On branch master

Changes not staged for commit:

(use "git add ..." to update what will be committed)

(use "git checkout -- ..." to discard changes in working directory)

modified: main.cpp

Untracked files:

(use "git add ..." to include in what will be committed)

bin/

obj/

project.depend

no changes added to commit (use "git add" and/or "git commit -a")

В отличие от первого случая вывод сообщает нам о том что у нас есть не отслеживаемые файлы, а также файл main.cpp а так же подсказывает команду чтобы откатить изменения в этом файле.

Рассмотрим 3 способа чтобы закоммитить изменение в файле

Способ 1. Сначала выбрать файлы, изменения которых должны войти в коммит, затем сделать коммит:

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git add main.cpp

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git commit -m "code: Hello world заменен на ввод чисел"

[master 8e8d479] code: Hello world заменен на ввод чисел

1 file changed, 3 insertions(+), 2 deletions(-)

Этот способ полезен, если изменения произошли в одном файле.

Добавим в программу вывод разности a и b.

**Способ 2.** Добавить в индекс все изменения, затем сделать коммит:

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git add -u

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git commit -m "code: Добавлен вывод разности"

[master a3427ea] code: Добавлен вывод разности

1 file changed, 1 insertion(+)

Этот способ добавляет в индекс все измененные файлы. Полезен если произошли изменения в большом количестве файлов

Добавим вывод суммы в программу.

**Способ 3.** Добавить все изменения в индекс и сделать коммит в один шаг:

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git commit -a -m "code: Добавлен вывод суммы"

[master 2de899b] code: Добавлен вывод суммы

1 file changed, 1 insertion(+), 1 deletion(-)

Способ полностью эквивалентен предыдущему и удобен, если коммит меняет только существующие файлы.

**Игнорирование файлов**

Через CB создадим файл .gitignore и запишем в первую строку /bin, а во вторую /obj чтобы в дальнейшем Git игнорировал эти директории.

Выполнив git status, можно видеть, что каталог bin не отображается.

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git status

On branch master

Untracked files:

(use "git add ..." to include in what will be committed)

.gitignore

project.depend

nothing added to commit but untracked files present (use "git add" to track)

Закоммитим файл игнорирования

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git add .gitignore

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git commit -m "git: дабавлен игнор bin и obj"

[master 4899e9b] git: дабавлен игнор bin и obj

1 file changed, 1 insertion(+)

create mode 100644 .gitignore

### Просмотр коммитов

Посмотрим коммит с добавлением суммы тремя разными способами:

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git show HEAD~2

commit 2de899b4e595bf0f0e6d064b218625b78df2ec0c

Author: example@example.uk

Date: Mon Mar 1 15:02:54 2021 +0300

code: Добавлен вывод суммы

diff --git a/main.cpp b/main.cpp

index b8b140d..6f9254c 100644

--- a/main.cpp

+++ b/main.cpp

@@ -7,5 +7,5 @@ int main()

cout << "Enter A and B: ";

int a, b;

cin >> a >> b;

- cout << "A - B = " << a - b << '\n';

+ cout << "A + B = " << a + b << '\n' << "A - B = " << a - b << '\n';

}

### Просмотр изменений

Чтобы посмотреть все изменения от первого коммита до HEAD~3 коммита воспользуемся командой git diff HEAD~3

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git diff HEAD~3

diff --git a/.gitignore b/.gitignore //Первое изменение произошло с файлом .gitignore

new file mode 100644

index 0000000..4c7473d

--- /dev/null //тут его не было

+++ b/.gitignore

@@ -0,0 +1,2 @@

+/bin //тут в него добавили 2 строки

+/obj

diff --git a/main.cpp b/main.cpp // Второе изменение произошло в файле main.cpp

index b8b140d..d737a5a 100644

--- a/main.cpp

+++ b/main.cpp

@@ -7,5 +7,5 @@ int main()

cout << "Enter A and B: ";

int a, b;

cin >> a >> b;

- cout << "A - B = " << a - b << '\n'; //вот что было

+ cout << "A + B = " << a + b << '\n' << "A - B = " << a - b << '\n' << "A \* B = " << a \* b << '\n'; //вот что стало

}

## Откат изменений

Закоммитим изменения в рабочей копии (вывод произведения).

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git commit -a -m "code: добавлен вывод произведения"

[master acef315] code: добавлен вывод произведения

1 file changed, 1 insertion(+), 1 deletion(-)

Предположим, необходимо отменить (откатить) этот коммит, то есть вернуться к предыдущему. Для этого воспользуемся командной git reset:

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git reset --hard HEAD~1

HEAD is now at f5ac6fb git: добавлен игнор obj

Также можно откатить один файл, если нужно восстановить его состояние таким, какое оно было в известный момент.

Напишем над main() любой комментарий а затем откатимся до момента где, его не было.

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git checkout HEAD -- main.cpp

## Обмен кодом через удаленное хранилище

### Отправка проекта на GitHub

Зарегистрировавшись на Github, отправим наш репозитарий туда.

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git remote add origin https://github.com/ikhlopov/cs-lab02.git

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (master)

$ git branch -M main

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (main)

$ git push -u origin main

### Получение проекта с GitHub

Предположим, к разработке проекта присоединяется Боб. Откроем новый терминал Bash в каталоге Bob.

Далее клонируем репозитарий в «компютер» Боба и называем рабочую папку Project

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Bob

$ git clone https://github.com/ikhlopov/cs-lab02.git Project

Cloning into 'Project'...

remote: Enumerating objects: 21, done.

remote: Counting objects: 100% (21/21), done.

remote: Compressing objects: 100% (16/16), done.

remote: Total 21 (delta 2), reused 21 (delta 2), pack-reused 0

Unpacking objects: 100% (21/21), done.

На машине Боба настроим Git config аналогично тому как это делалось для Алисы

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Bob/Project (main)

$ git config user.name 'Bob(KhlopovIK)'

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Bob/Project (main)

$ git config user.email 'Bob@mail.ru'

## Совместная работа над проектом без конфликтов правок

Добавим на машине Боба печать произведения, и сделаем коммит.

Отправим коммит на Github командой git push

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Bob/Project (main)

$ git push

Обновив страницу репозитария в браузере и убедившись, что коммит попал на Github, выполним подгрузку изменений на машине Алисы.

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (main)

$ git fetch

remote: Enumerating objects: 5, done.

remote: Counting objects: 100% (5/5), done.

remote: Compressing objects: 100% (2/2), done.

remote: Total 3 (delta 1), reused 3 (delta 1), pack-reused 0

Unpacking objects: 100% (3/3), done.

From https://github.com/ikhlopov/cs-lab02

f5ac6fb..a135c59 main -> origin/main

Посмотрим историю всех веток

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (main)

$ git log --oneline --decorate --all --graph

\* a135c59 (origin/main) Code: Добавлен вывод произведения

\* f5ac6fb (HEAD -> main) git: добавлен игнор obj

\* 4899e9b git: дабавлен игнор bin и obj

\* 2de899b code: Добавлен вывод суммы

\* a3427ea code: Добавлен вывод разности

\* 8e8d479 code: Hello world заменен на ввод чисел

\* de484e5 build: Добавлен файл проекта

\* d0bd57a code: заготовка программы

Как можно видеть, ветка master отстает на один коммит от ветки origin/master (версии ветки master из удаленного репозитария под названием origin, то есть на GitHub).

Продвинем ветку master к скачанной версии.

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (main)

$ git pull --ff-only

Updating f5ac6fb..a135c59

Fast-forward

main.cpp | 2 +-

1 file changed, 1 insertion(+), 1 deletion(-)

От имени Алисы добавим в программу вывод деления, сделаем коммит и отправим на github

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (main)

$ git commit -a -m "code: Добавлен вывод деления"

[main 4a1d6d0] code: Добавлен вывод деления

1 file changed, 1 insertion(+), 1 deletion(-)

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Alice/Project (main)

$ git push

Logon failed, use ctrl+c to cancel basic credential prompt.

Username for 'https://github.com': ikhlopov

Enumerating objects: 5, done.

Counting objects: 100% (5/5), done.

Compressing objects: 100% (3/3), done.

Writing objects: 100% (3/3), 405 bytes | 202.00 KiB/s, done.

Total 3 (delta 1), reused 0 (delta 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To https://github.com/ikhlopov/cs-lab02.git

a135c59..4a1d6d0 main -> main

Теперь синхронизируеся с проектом от лица Боба.

u211-05@PROG-02 MINGW32 ~/Desktop/lab02/Bob/Project (main)

$ git pull --ff-only

remote: Enumerating objects: 5, done.

remote: Counting objects: 100% (5/5), done.

remote: Compressing objects: 100% (2/2), done.

remote: Total 3 (delta 1), reused 3 (delta 1), pack-reused 0

Unpacking objects: 100% (3/3), done.

From https://github.com/ikhlopov/cs-lab02

a135c59..4a1d6d0 main -> origin/main

Updating a135c59..4a1d6d0

Fast-forward

main.cpp | 2 +-

1 file changed, 1 insertion(+), 1 deletion(-)