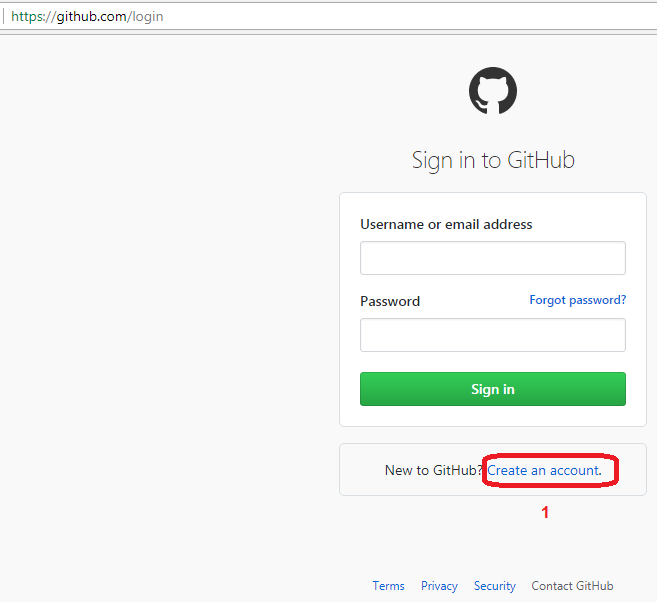
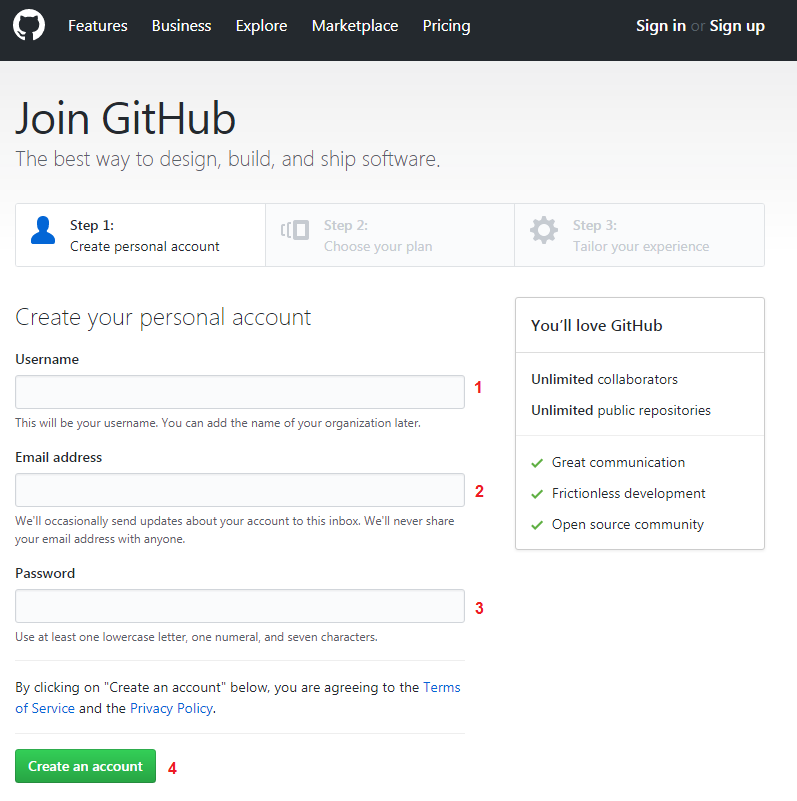
**Пример работы с Git**

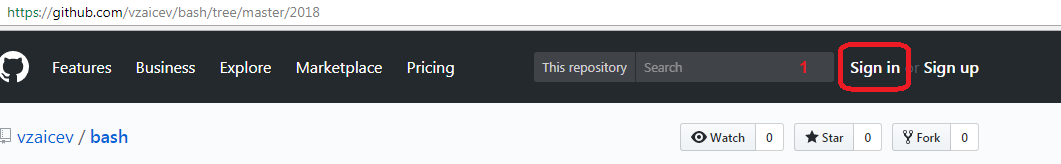
1 Регистрируемся на github.com

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Github_reg.png)[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Github_reg_acc.png)

2. Выбираем проект, над которым будем работать <https://github.com/vzaicev/bash/tree/master/2021>

В нашем случае это проект по выполнению лабораторных работ за 2021 год

3 Логинимся на GitHub, Если еще не залогинились, то

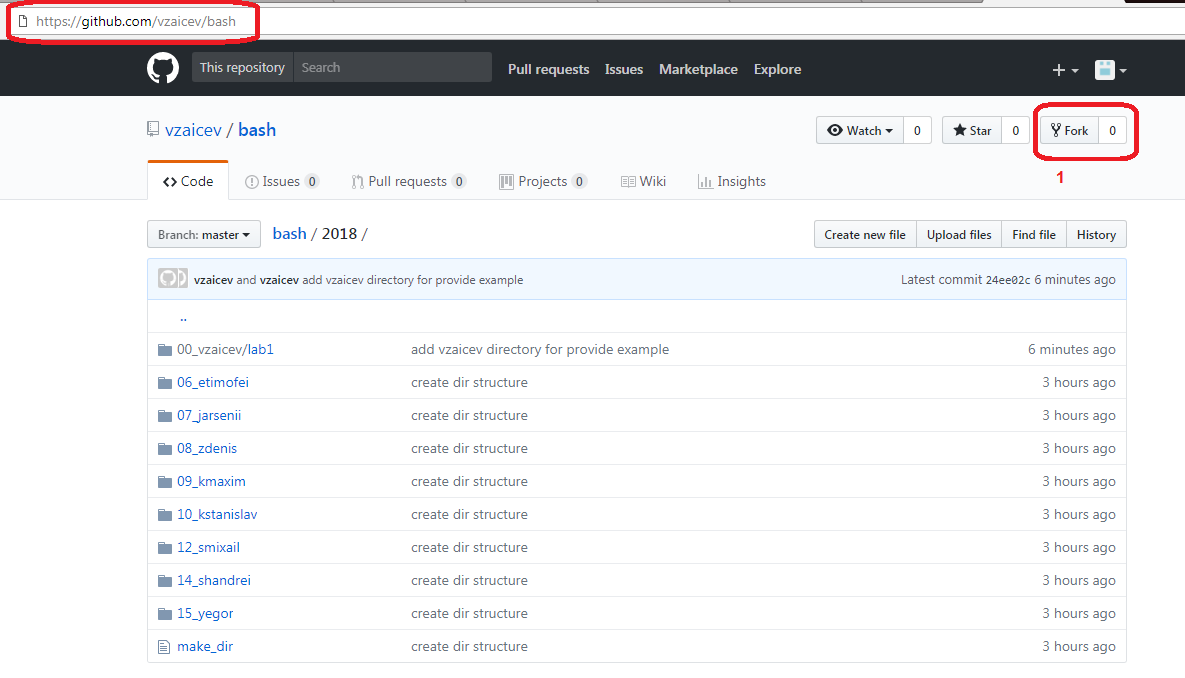
[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Github_reg_login.png)

4. Создаем fork проекта

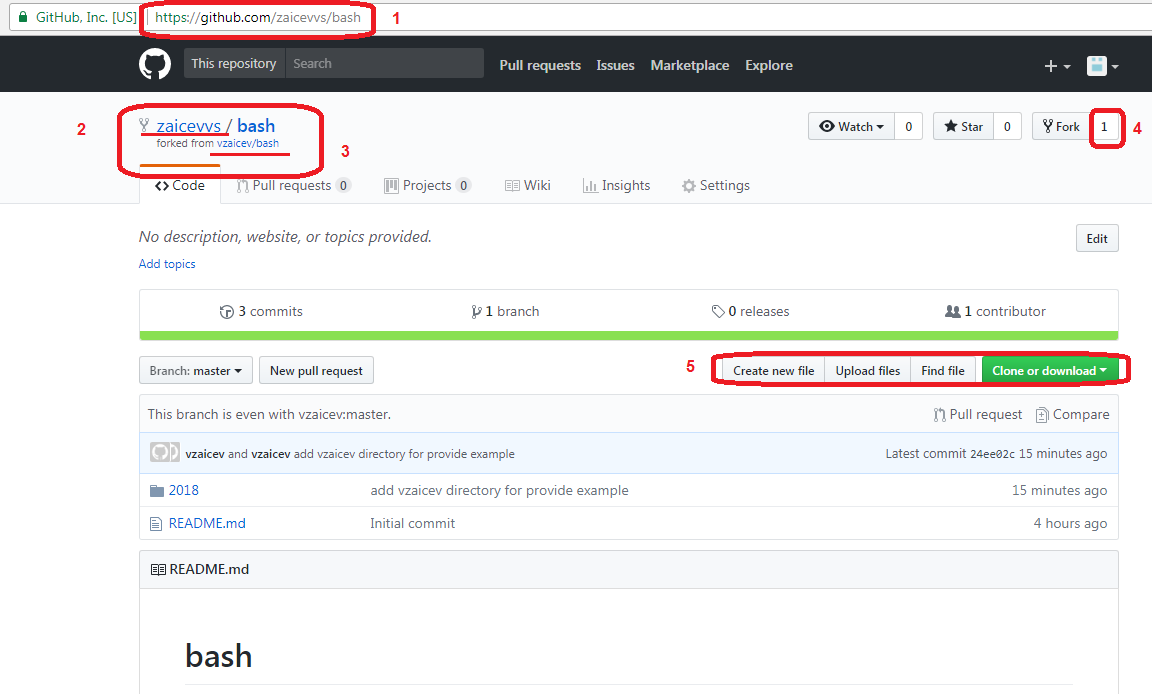
* Есть 2 способа ведения работы над проектом
  + Используя Ваш собственный репозиторий и общий пароль и права доступа к нему
  + Используя чужой репозиторий, **fork** и **push** request для добавления изменений

Мы будем использовать второй вариант (fork)

1. Заходим в проект <https://github.com/vzaicev/bash/tree/master/2021>
2. Делаем ответвление (fork) от проекта в свой репозиторий.

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Github_reg_fork.png)

* Получаем копию начального репозитория у себя
* Обратите внимание, что путь в браузере будет содержать **ваш адрес и имя** **(1)**
* Также обратите внимание на то, что будет указано из какого проекта изначально была сделана ветка **(2,3)**
* Счетчик форков будет показывать сколько ветвлений Вы сделали **(4)**
* Также вам станет доступно добавление файлов и директорий, и делать сабмиты под Вашим пользователем **(5)**

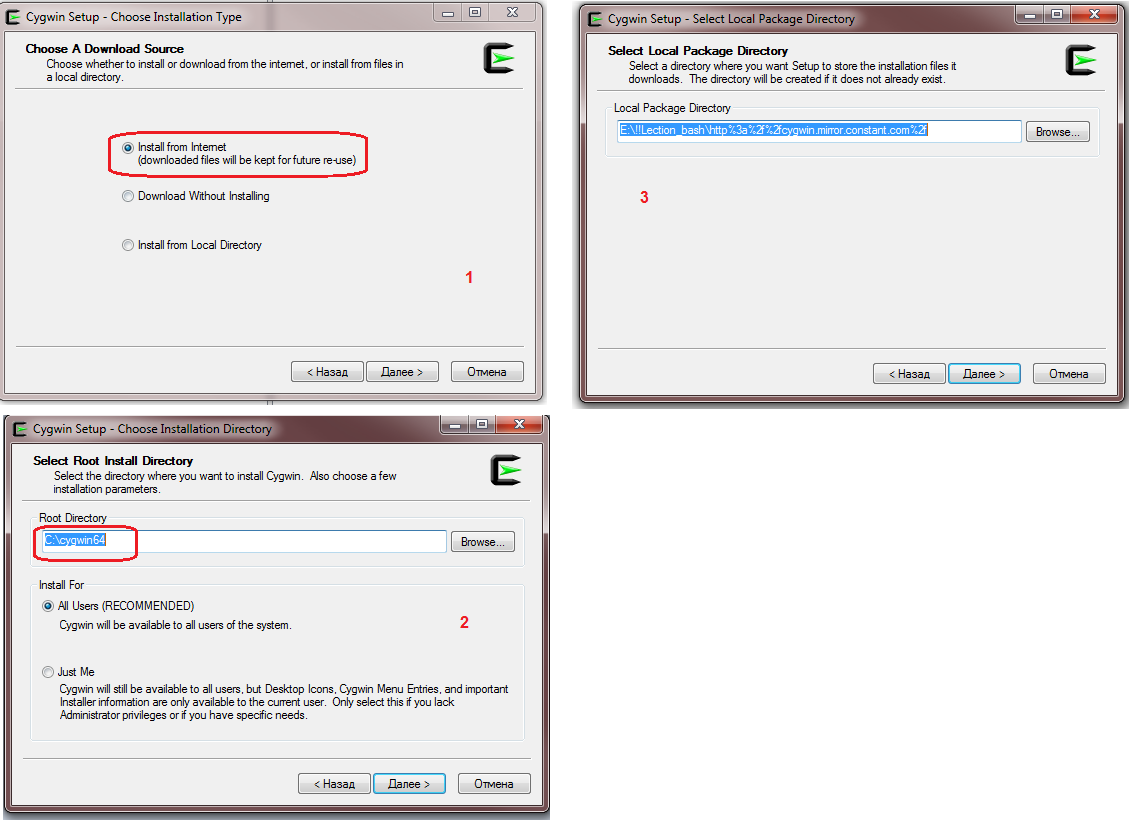
[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Github_reg_fork2.png)

5. Устанавливаем пакет для Cygwin для работы с git

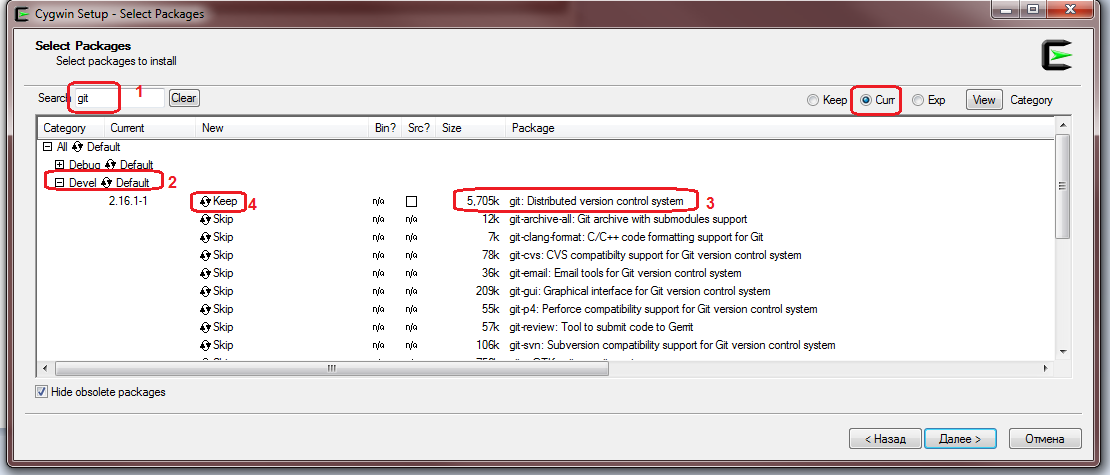
* Запускаем скачанный с сайта установщик <https://cygwin.com/install.html>

**Обязательно обращаем внимание на то, чтобы путь не содержал русских букв и пробелов**

* Задаем путь установки
* Задаем путь где локально дуте хранится информация о пакетах
* Выбираем зеркало с какого будем скачивать пакеты (можно первое в списке)

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Cygwin_install.png)

* Ищем в списке пакетов GIT и добавлем его к установке

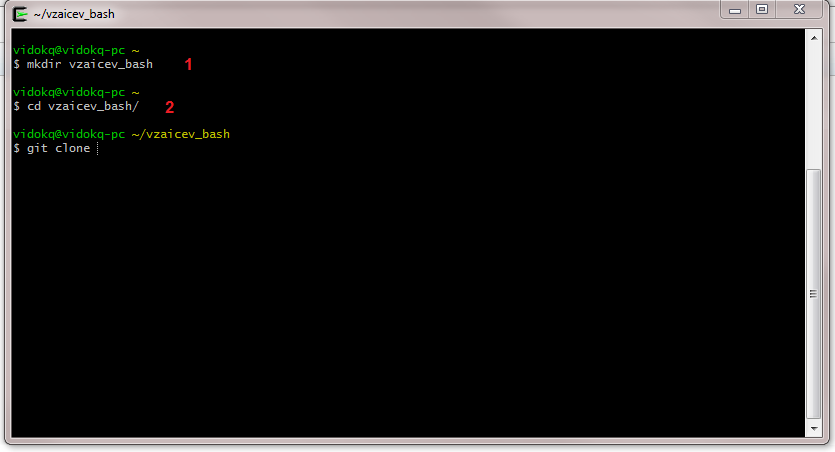
[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Cygwin_install_git.png)

6. Загружаем свой репозиторий на локальную машину

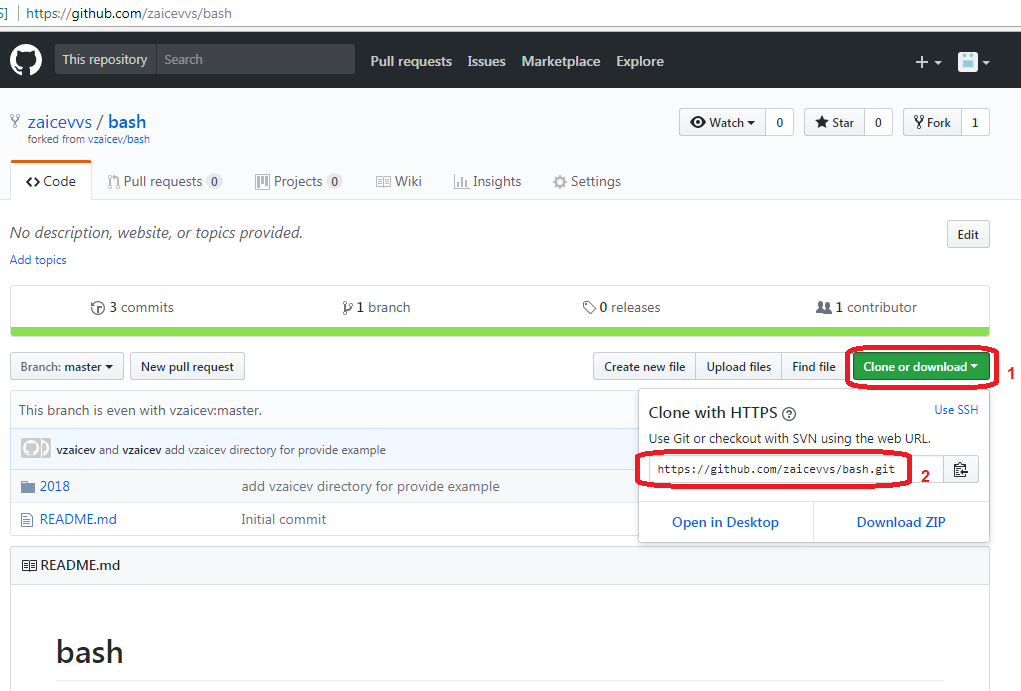
* Запускам терминал и создаем рабочую директорию

mkdir name\_of\_dir # создаем директори

cd name\_of\_dir # Переходим в директорию

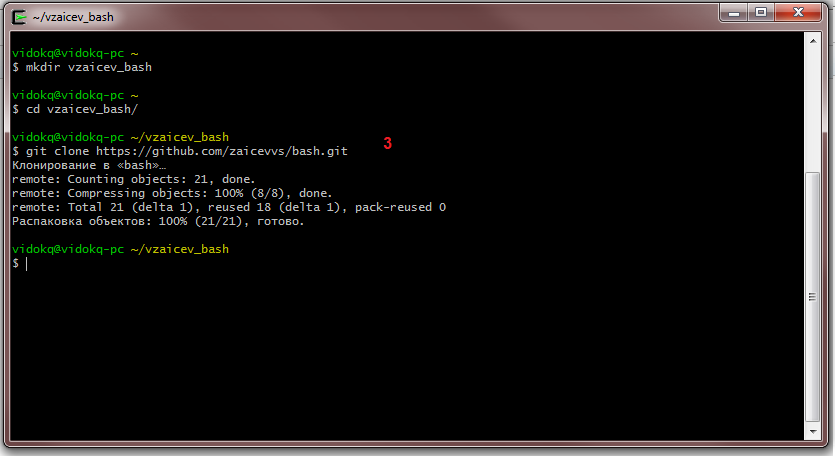
[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Mkdir_for_git.png)

* Копируем адрес нашего форкнутого репозитория

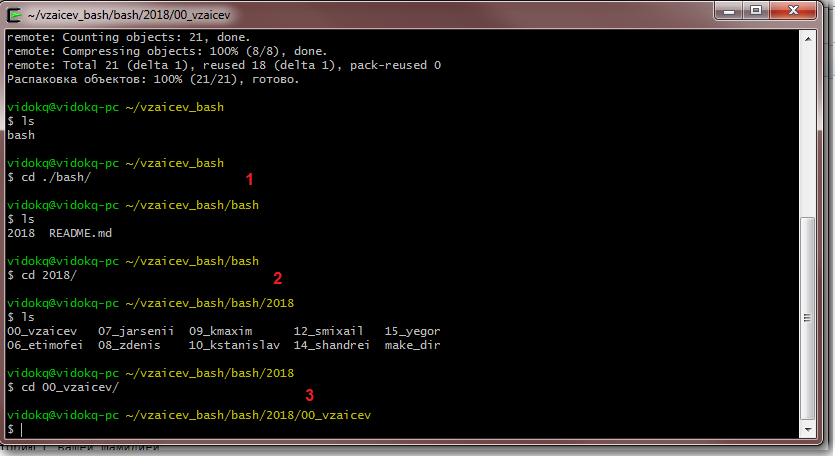
[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Clone_to_loc.png)

* Извлекаем репозиторий на локальную машину

git clone <https://github.com/zaicevvs/bash.git>

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Clone_to_loc_cmd.png)

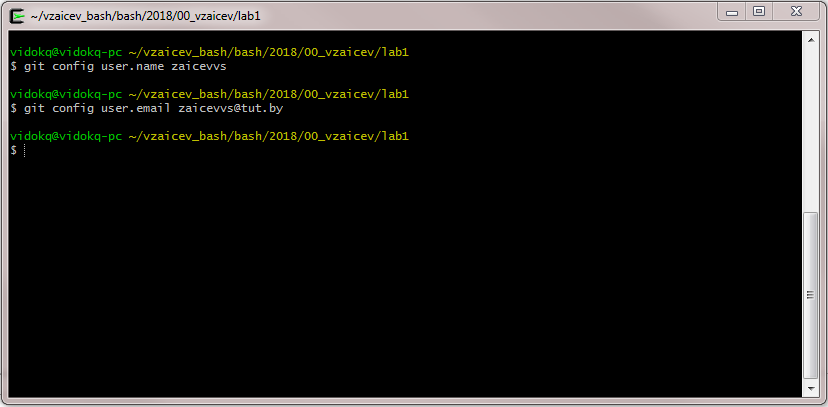
* Теперь в нашем локальной директории есть копия всего проекта.

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Cd_to_work.png)

* Переходим в свежесозданную папку репозитория и настраиваем его
  + Указываем имя пользователя (то, что использовали при регистрации на Github)
  + Указываем адрес электронной почты (тот, что использовали при регистрации на Github)

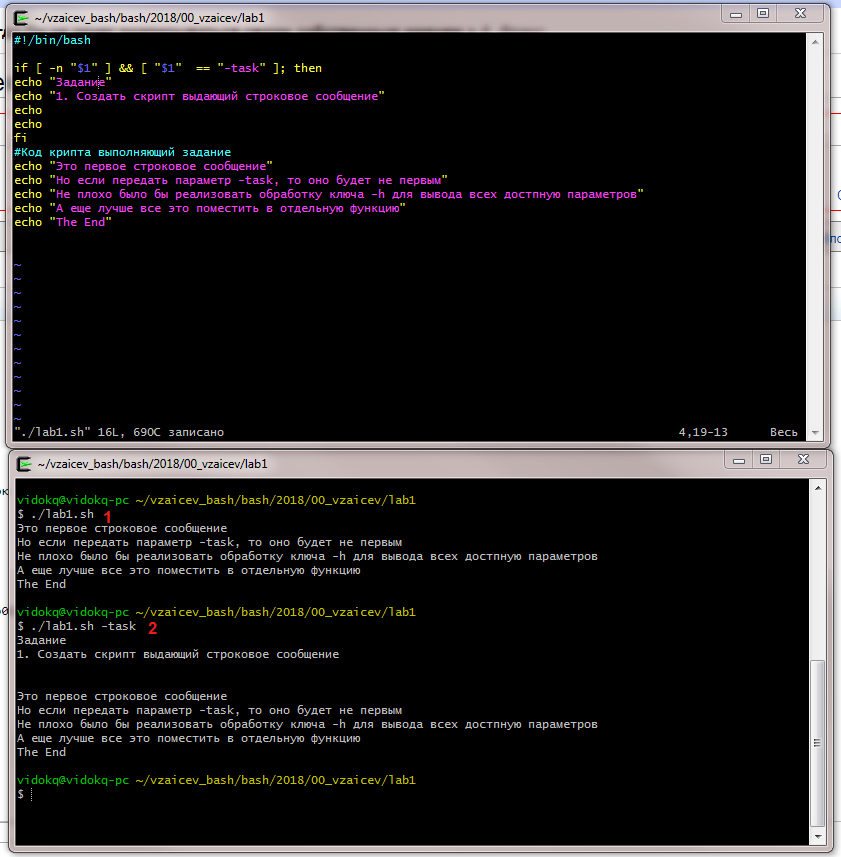
git config user.name zaicevvs

git config user.email zaicevvs@ex.com

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Config_git.png)

7. Выполняем практическое задание

* Результатом задания будет скрипт
* Скрипт должен содержать, ключ который позволит вывести в задание
  + Название ключа -task
  + Ключ может идти не обязательно первым
  + Сообщение должно идти первым и отображать задание
  + После задания должны следовать 2 пустые строки
  + Далее результат выполнения скрипта
  + В случае, если ключ не задан, то результат выполнения идет сразу без дополнительных пустых строк
* Скрипт должен иметь название следующего формата lab<XX>\_z<YY>.sh, где
  + XX – номер лабораторной
  + YY – номер задачи или имя дополнительного скрипта, необходимого для выполнения задачи
  + файл должен иметь расширение **\*.sh**
  + Файл должен иметь права доступа 755 (команда установки прав доступа: chmod 755 lab01\_z1.sh)
  + Если в лабораторной нет подзадач, то в названии допускается отсутствие \_z<YY> (например: lab01.sh)

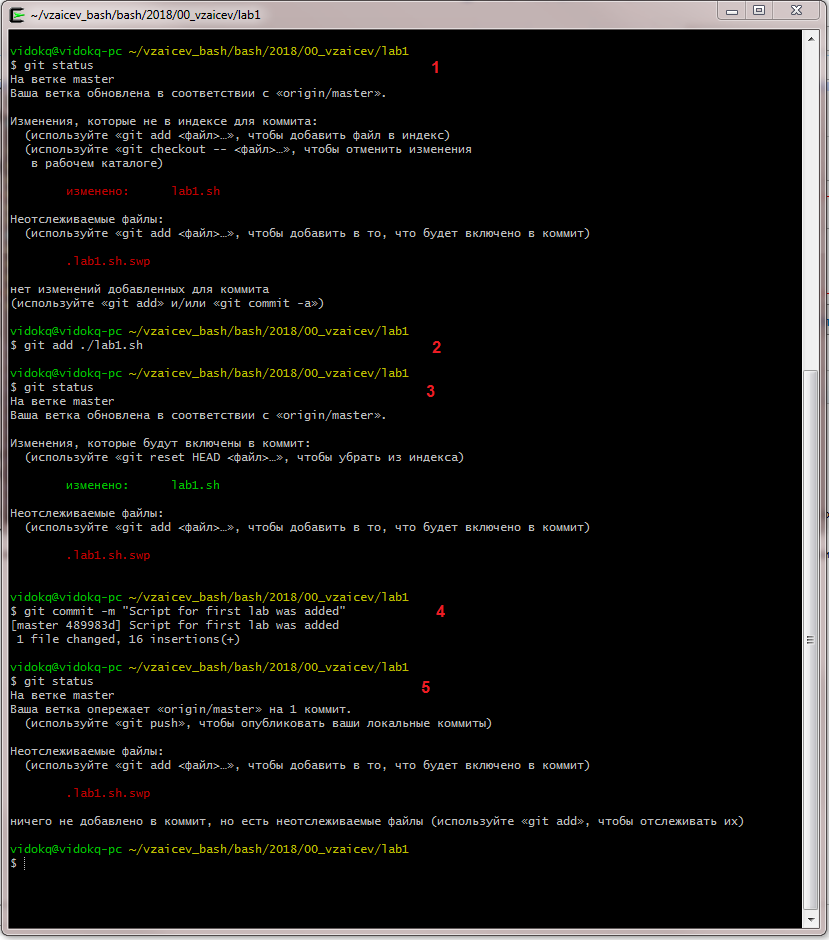
[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Bash_Lab1_z1.png)

Добавление изменений в локальный репозиторий

* После того, как скрипт написан можно его добавить в локальный репозиторий (**закомитить**)
  + **git status**. Воспользовавшись командой git status можно узнать, на какой ветке (branch) репозитория Вы сейчас находитесь, какие изменения присутствуют в Вашей рабочей копии и другую информацию.

*Рабочей копией* называется совокупность файлов в локальной папке репозитория за исключением служебных файлов.

* + **git add %file\_path%** – Подготавливаем файлы к добавлению
  + **git commit -m "%commit\_message%"** – Сохраняем файлы в репозиторий. Использование команды без аргументов откроет текстовый редактор, где надо будет написать комментарий для коммита, коммит обязательно должен иметь комментарий. Другим вариантом задания комментария к коммиту является использование команды
  + Историю изменений можно посмотреть командой **git log** или **git log --name-only**. Если вся история изменений не умещается на экране, то можно пользоваться клавишами прокрутки на клавиатуре ("стрелочки", PgUp, PgDown), выход из режима просмотра изменений осуществляется нажатием клавиши "q".

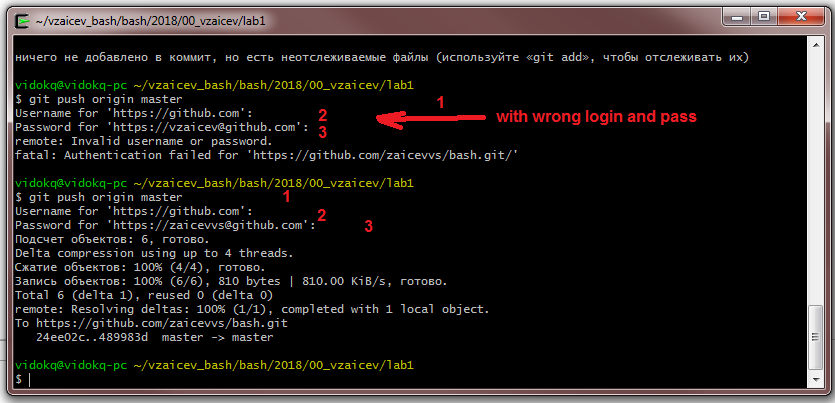
[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Commit_git.png)

Отправка изменений на GitHub репозиторий

* Далее отправим изменения на GitHub сервер

git push origin master

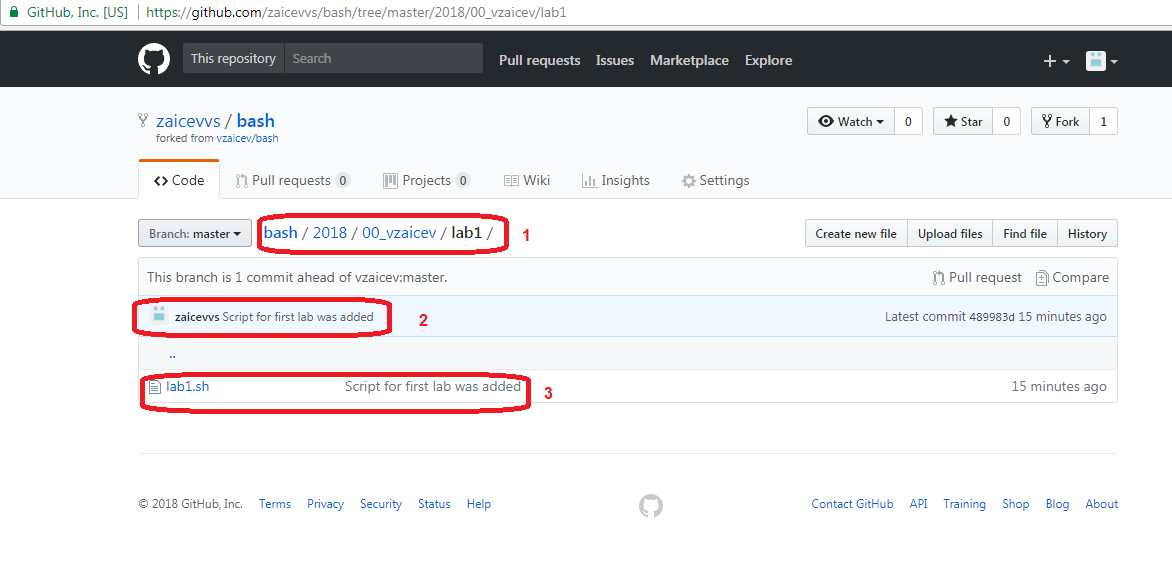
* При выполнении команда запросит логин и пароль к вашему репозиторию на GitHub.
  + При вводе из командной строки текст отображаться не будет, он скрыт, но он подставляется, так что продолжайте вводить.

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Push_git.png)

* Если выполнение команды произошло с ошибкой, значит на сервере кто-то изменил этот файл. Чтобы решить эту проблему нужно забрать последние изменения с сервера к себе и объединить их с локальными правками.

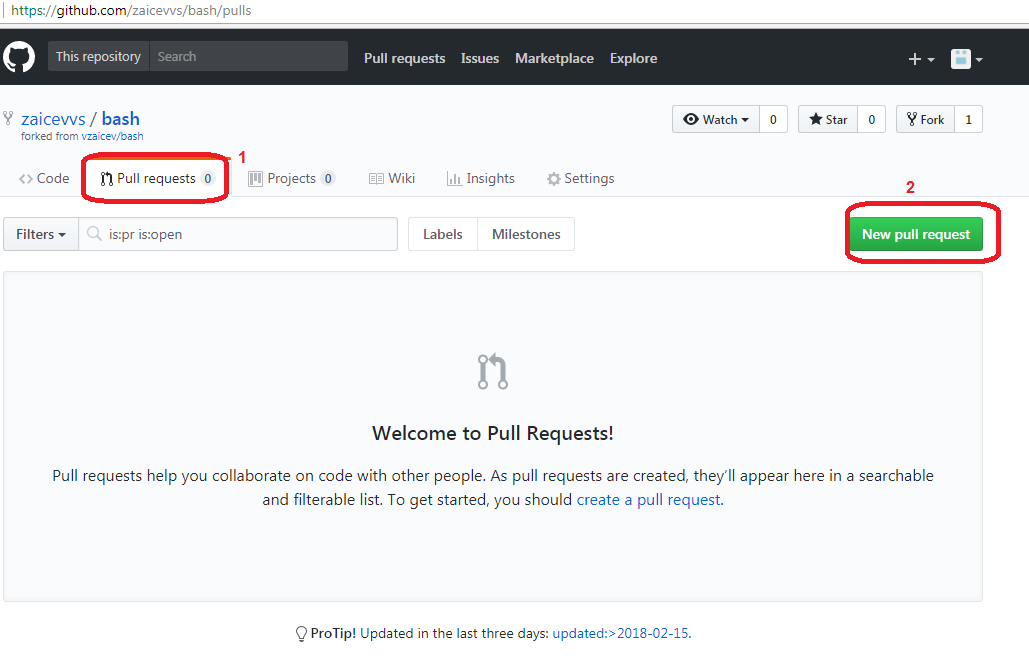
git pull

* После чего можно повторить команду отправки снова.
* Если выполнение прошло успешно, то можно посмотреть статус на GitHub.

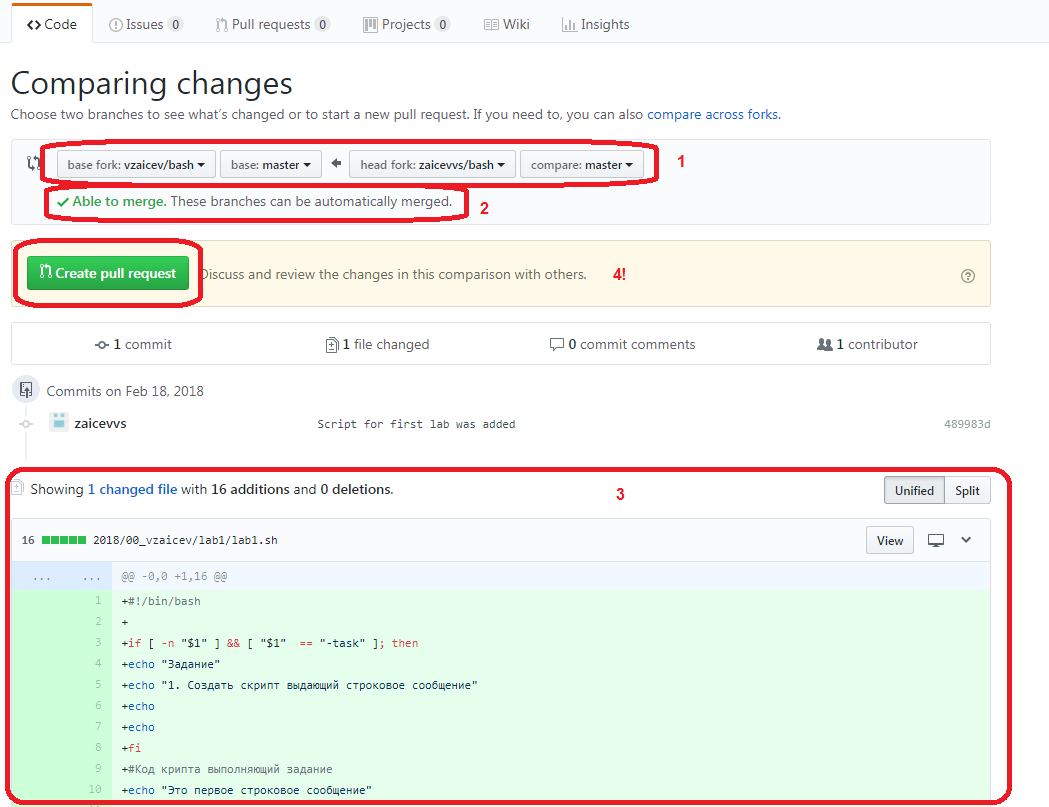
[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Status_on_web_git.png)

8. Отправка результатов в общий репозиторий для проверки преподавателем

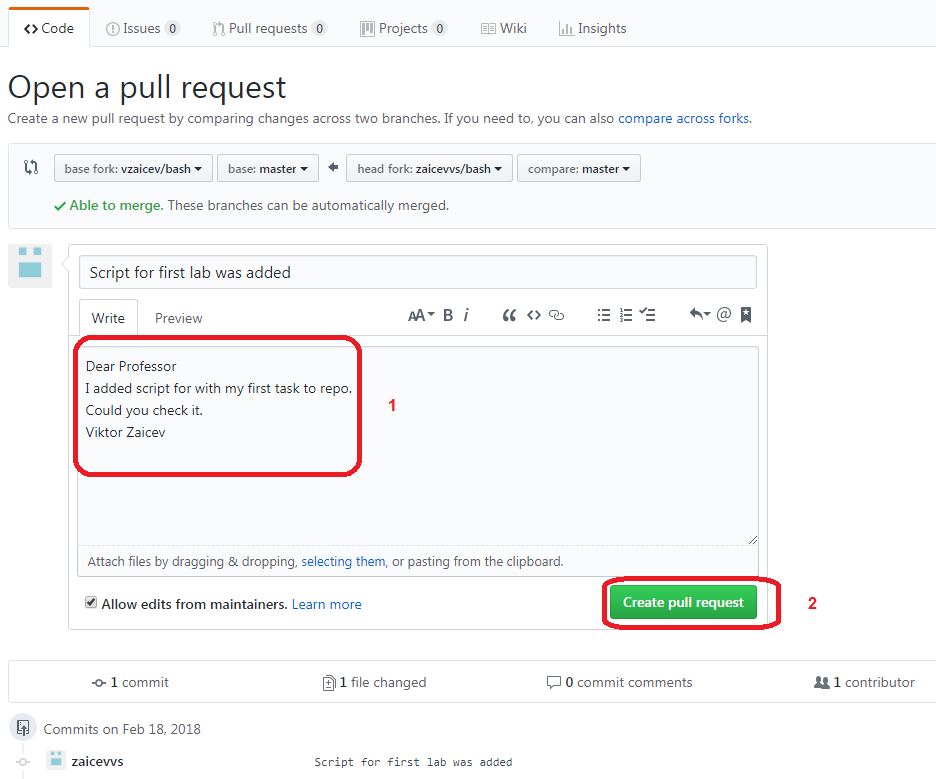
* Для отправки ваших изменений в общий репозиторий Вашей группы нужно сделать pull запрос
* Выполним такой запрос из Web на GitHub.
  + Переходим во вкладку **Pull request**
  + Нажимаем кнопку **New pull request**

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Pull_to_common_rep.png)

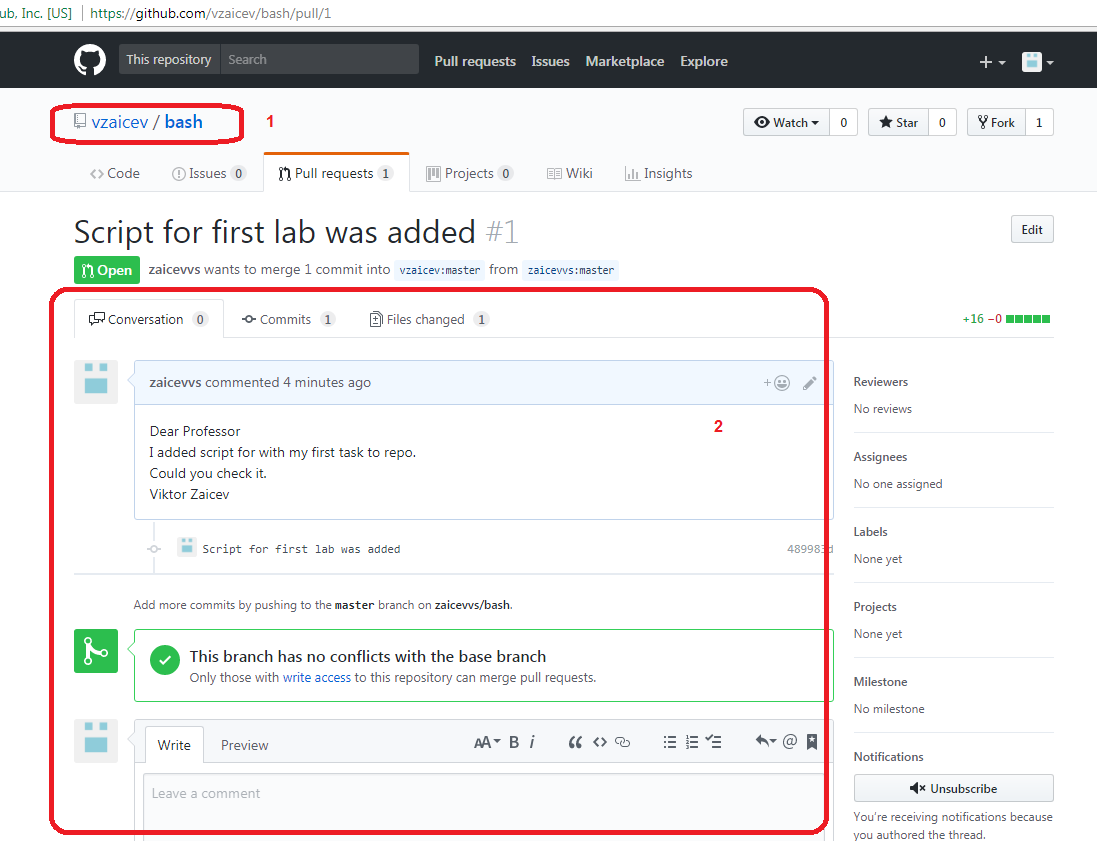
* Далее мы увидим статус нашего запроса
  + Что и куда собираемся мержить
  + Можно ли его замержить автоматически или придется вручную решать какие части кода оставить
  + Содержимое файлов которые были добавлены или нуждаются в мержинге
* Нажимаем кнопку **Create pull request** (номер 4! на рисунке ниже)

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Pull_to_common_merge_rep.png)

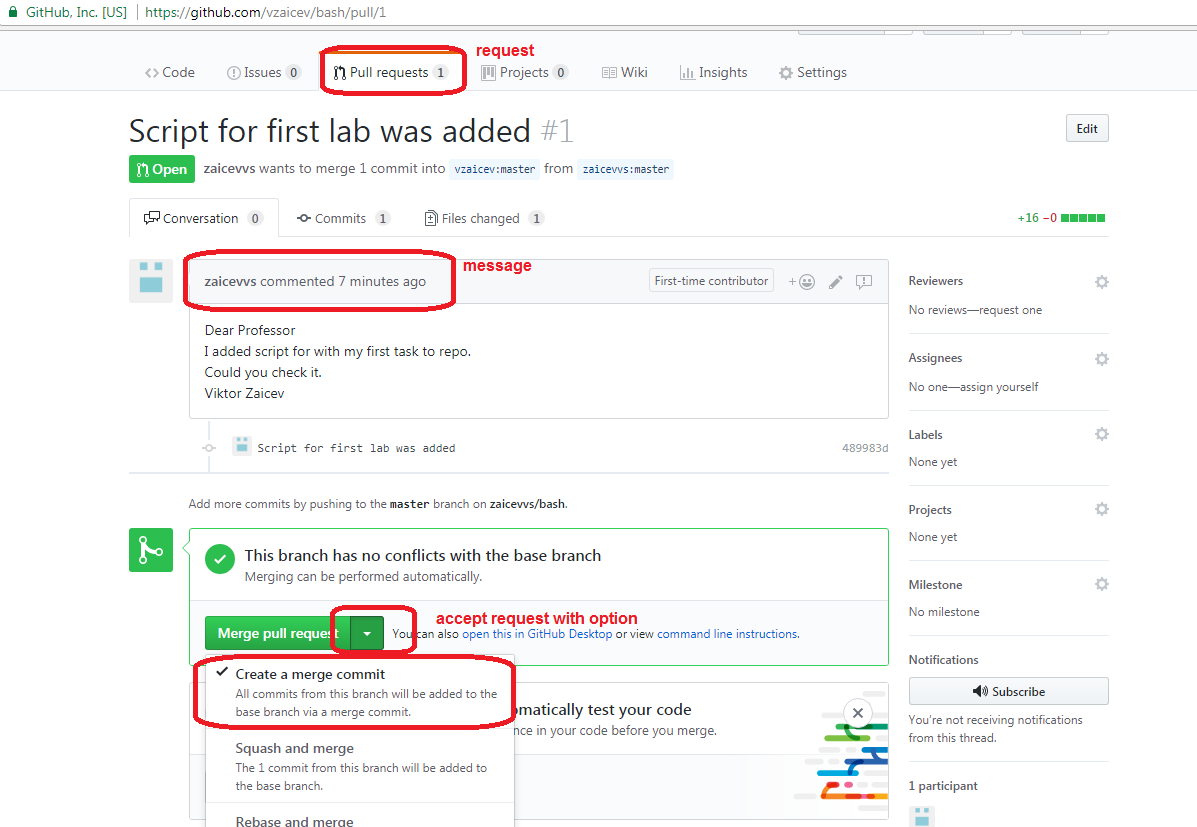
* Добавляем описание для нашего запроса.
  + Описание может содержать сообщение о выполненной работе
  + Или об исправлениях сделанных в ней

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Description_pull_to_common_merge_rep.png)

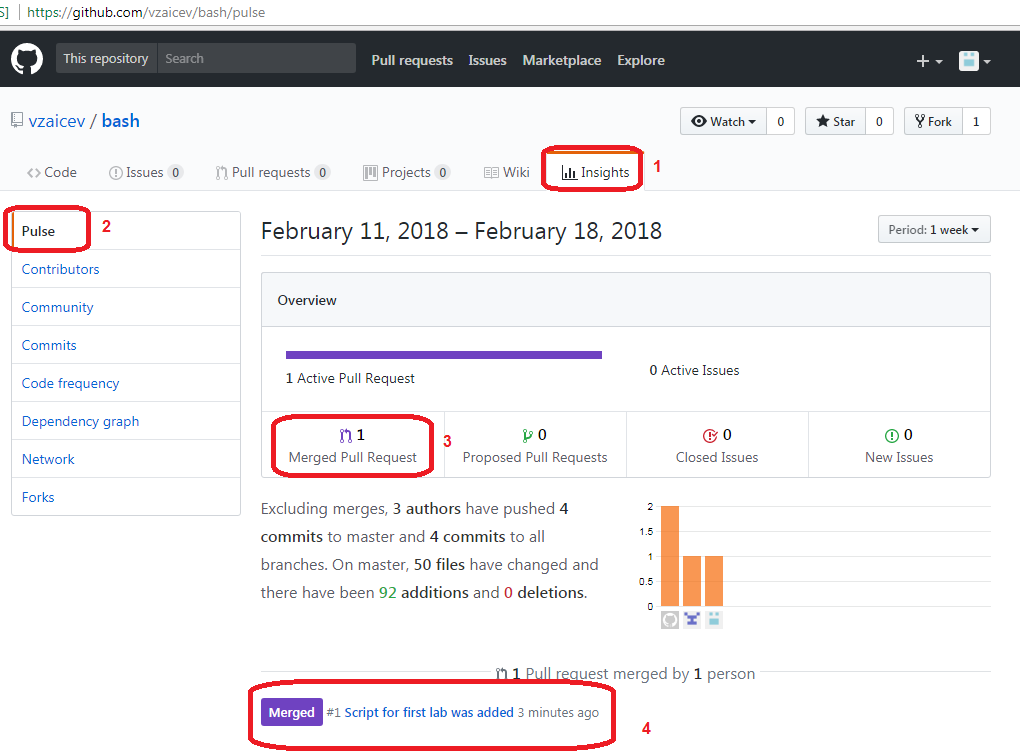
* Проверяем статус нашего запроса в общем репозитории группы

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Status_pull_to_common_merge_rep.png)

* Если кто-то запрашивает у Вас разрешение на внесение изменений в ваш репозиторий, то необходимо будет их проверить и принять или отклонить

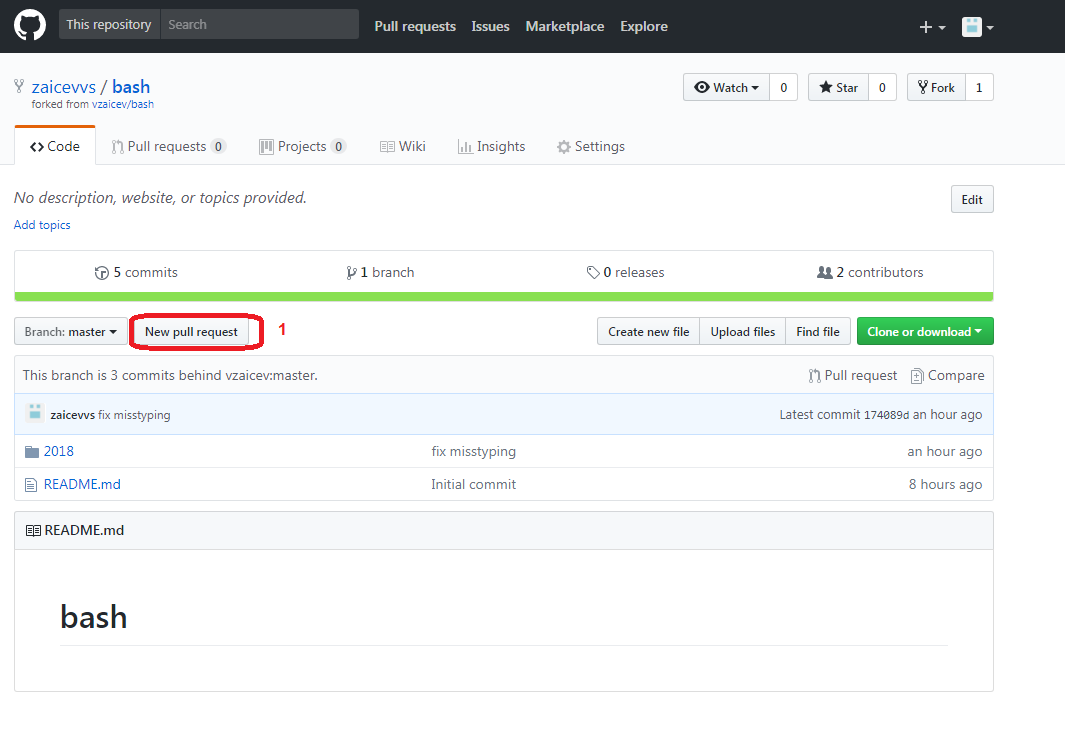
[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Accept_pull_request.png)

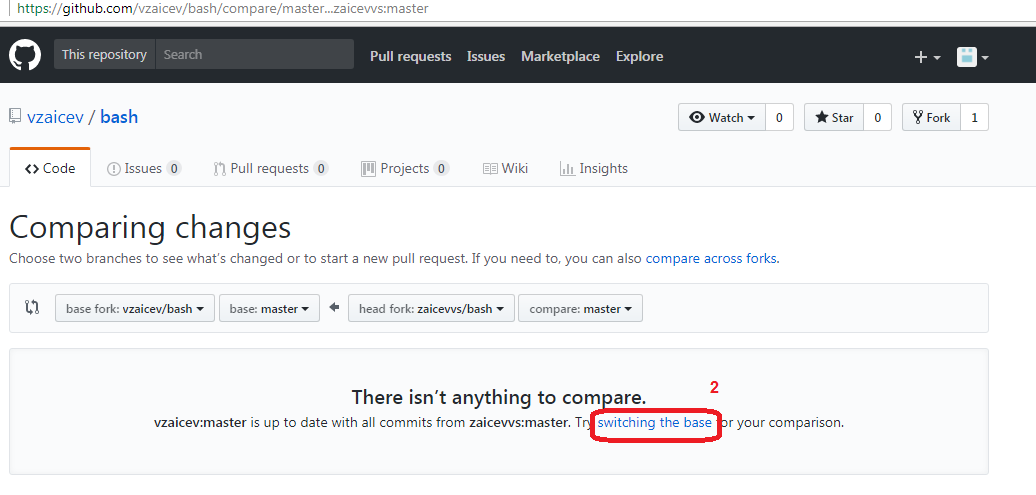
* После того как ваши изменения были приняты, Вы можете видеть сообщение об этом.

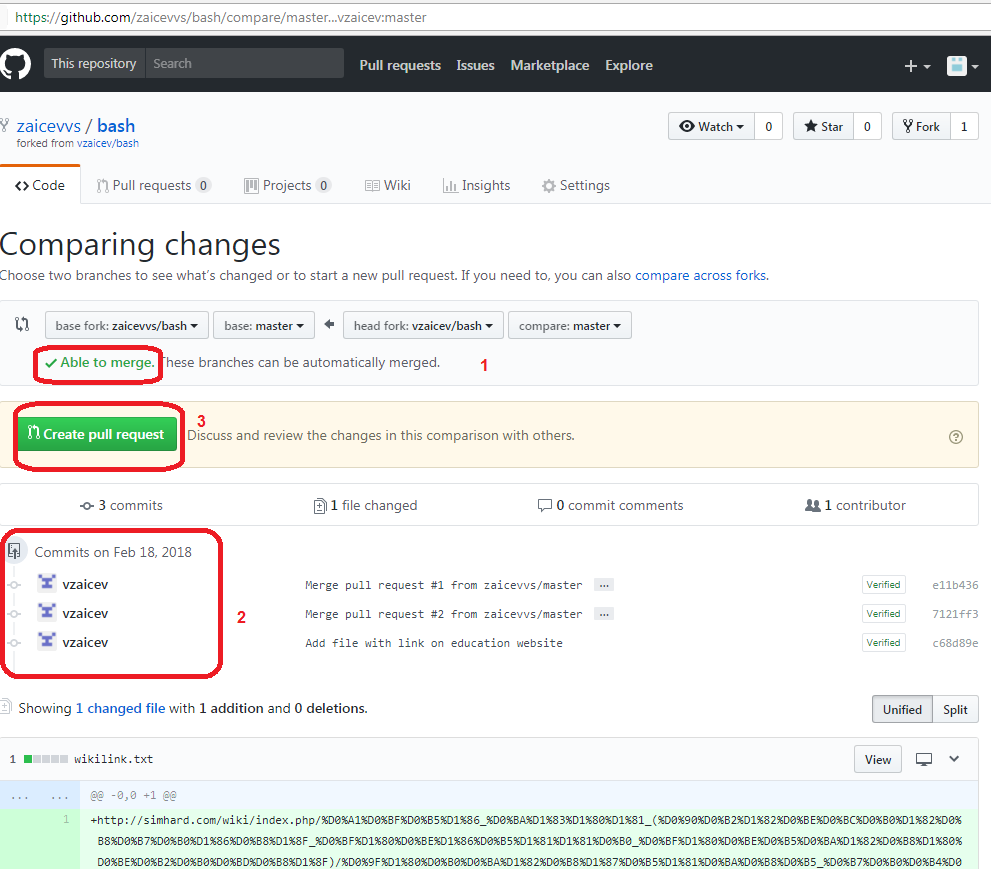
[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Check_after_accept_request.png)

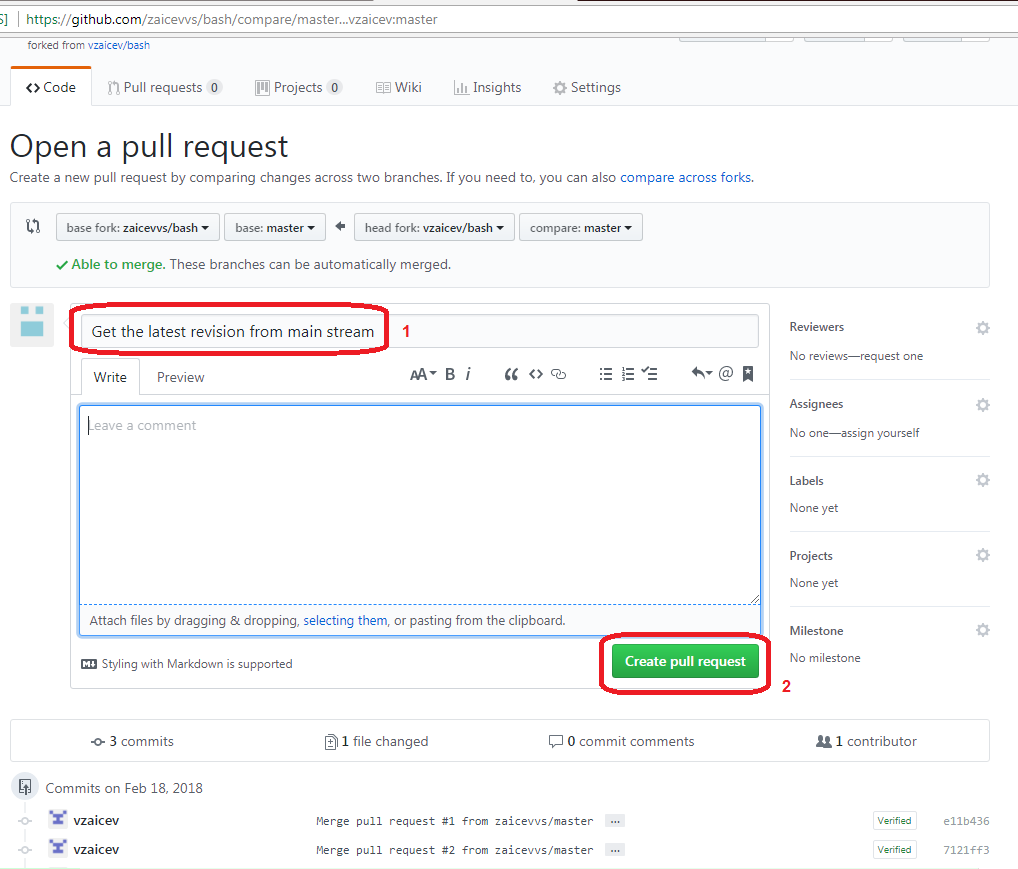
Перед выполнение следующего задания

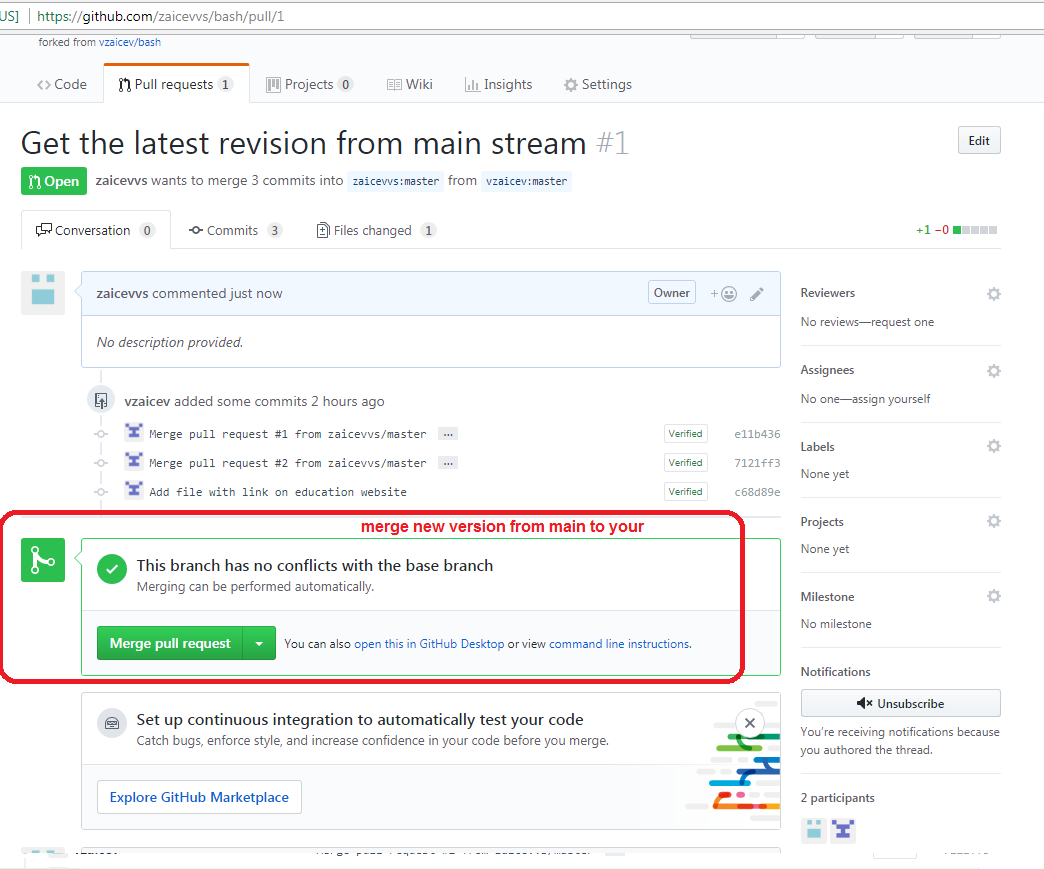
* После того как ваш Pull request был принят, можно приступать к след заданию
* Перед этим заберите последние изменения в ваш репозиторий на GitHub
  + Перейти на вкладку Pull Requests
  + Создаем новый реквест New pull request
  + Пишет, что нечего сравнивать, но не расстраиваемся и жмем switching the base
  + Появились отличия родительского и форкнутого проектов. Жмем Create pull request и даем ему название
  + Автоматически оказываемся в этом пул реквесте и жмем Merge pull request
  + И напоследок Confirm merge

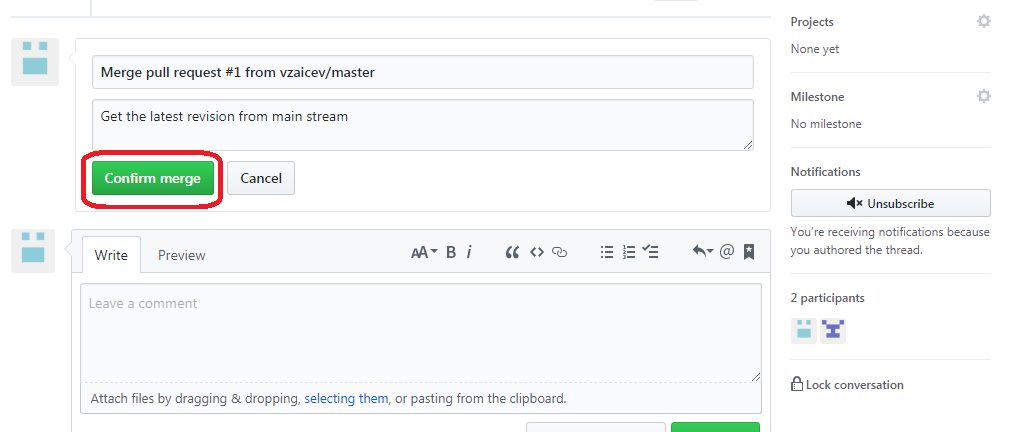
[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Rebase1_git.png)

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Rebase2_git.png)

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Rebase3_git.png)

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Rebase4_git.png)

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Rebase5_git.png)

[](http://simhard.com/wiki/index.php/%D0%A4%D0%B0%D0%B9%D0%BB:Rebase6_git.png)

* После заберите изменения в ваш локальный репозиторий

git pull