

Установка GIT и настройка GitHub

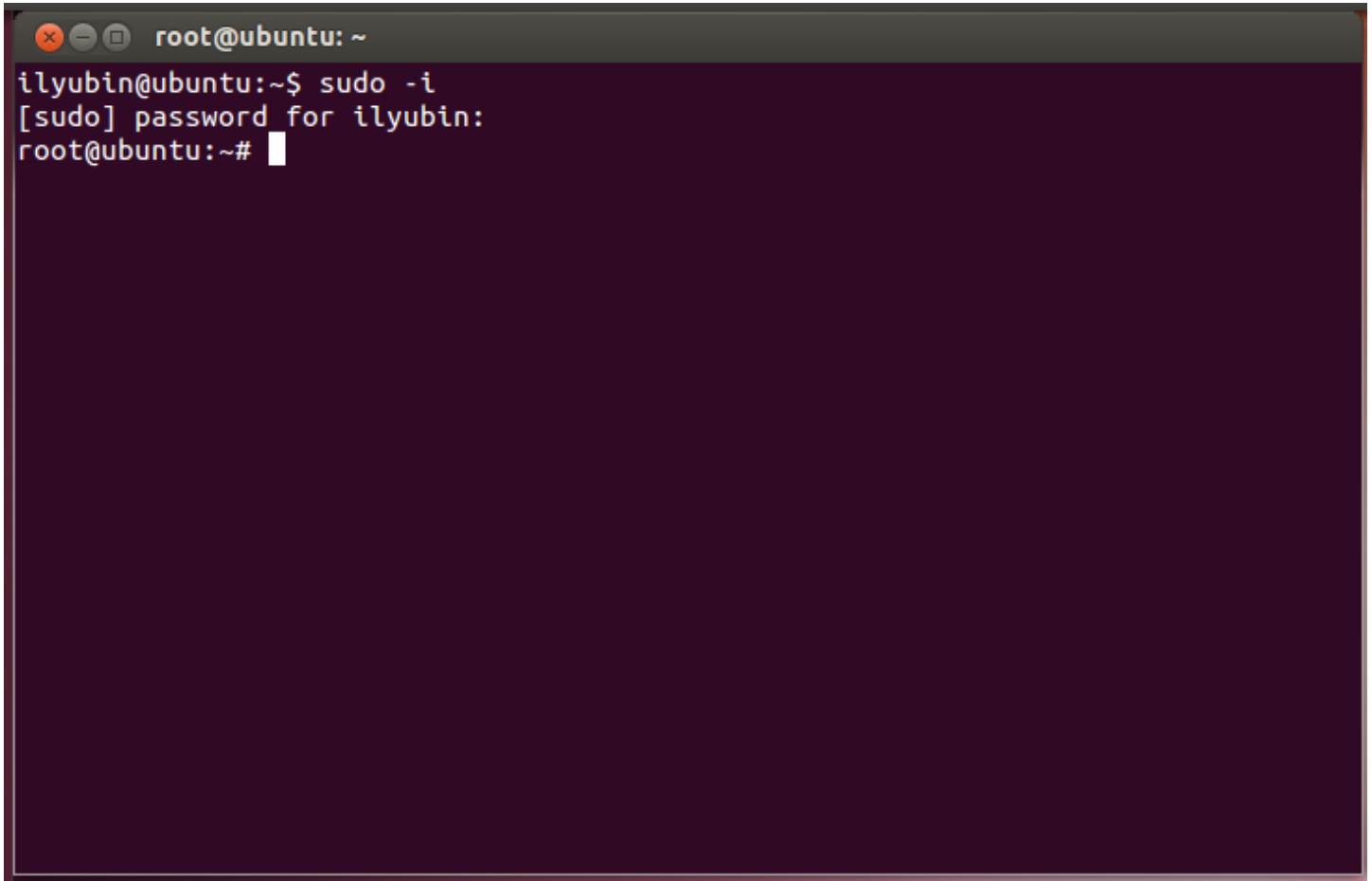
Здесь описывается практическая часть вопроса использования Git - его установка и регистрация на сервере GitHub.com.

Установка Git под Ubuntu

Пользователям Linux, думаю, нет смысла объяснять, как устанавливать Git - в каждой системе это делается по-разному.

Прежде переключаемся в режим администратора.

sudo -i

A terminal window with a dark purple background. The title bar shows 'root@ubuntu: ~'. The terminal text shows a user 'ilyubin@ubuntu:~\$' running 'sudo -i'. A prompt '[sudo] password for ilyubin:' is shown, followed by the user becoming 'root@ubuntu:~#'.

```
root@ubuntu: ~
ilyubin@ubuntu:~$ sudo -i
[sudo] password for ilyubin:
root@ubuntu:~#
```

Для установки Git, можно использовать команду:

apt-get install git

```
root@ubuntu: ~
root@ubuntu:~# apt-get install git
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  gir1.2-ubuntuoneui-3.0 libubuntuoneui-3.0-1 thunderbird-globalmenu
Use 'apt-get autoremove' to remove them.
The following extra packages will be installed:
  git-man liberror-perl
Suggested packages:
  git-daemon-run git-daemon-sysvinit git-doc git-el git-arch git-cvs git-svn
  git-email git-gui gitk gitweb
The following NEW packages will be installed:
  git git-man liberror-perl
0 upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 6,741 kB of archives.
After this operation, 15.2 MB of additional disk space will be used.
Do you want to continue [Y/n]? Y
Get:1 http://us.archive.ubuntu.com/ubuntu/ precise/main liberror-perl all 0.17-1
[23.8 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu/ precise/main git-man all 1:1.7.9.5-1
[630 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu/ precise/main git amd64 1:1.7.9.5-1 [6
,087 kB]
```

Далее (выходим из режима администратора и) проверяем, доступен ли Git для работы. В любом каталоге даем команду:

git --version

Если получаем информацию о версии, то Git установлен и работает. Если получаем информацию что программа git не найдена, разбираемся что сделали не так.

```
ilyubin@ubuntu: ~  
Get:1 http://us.archive.ubuntu.com/ubuntu/ precise/main liberror-perl all 0.17-1  
[23.8 kB]  
Get:2 http://us.archive.ubuntu.com/ubuntu/ precise/main git-man all 1:1.7.9.5-1  
[630 kB]  
Get:3 http://us.archive.ubuntu.com/ubuntu/ precise/main git amd64 1:1.7.9.5-1 [6  
,087 kB]  
Fetched 6,741 kB in 22s (298 kB/s)  
Selecting previously unselected package liberror-perl.  
(Reading database ... 171132 files and directories currently installed.)  
Unpacking liberror-perl (from .../liberror-perl_0.17-1_all.deb) ...  
Selecting previously unselected package git-man.  
Unpacking git-man (from .../git-man_1%3a1.7.9.5-1_all.deb) ...  
Selecting previously unselected package git.  
Unpacking git (from .../git_1%3a1.7.9.5-1_amd64.deb) ...  
Processing triggers for man-db ...  
Setting up liberror-perl (0.17-1) ...  
Setting up git-man (1:1.7.9.5-1) ...  
Setting up git (1:1.7.9.5-1) ...  
root@ubuntu:~#  
root@ubuntu:~# exit  
logout  
ilyubin@ubuntu:~$ git --version  
git version 1.7.9.5  
ilyubin@ubuntu:~$
```

Настройка SSH-ключей в Ubuntu

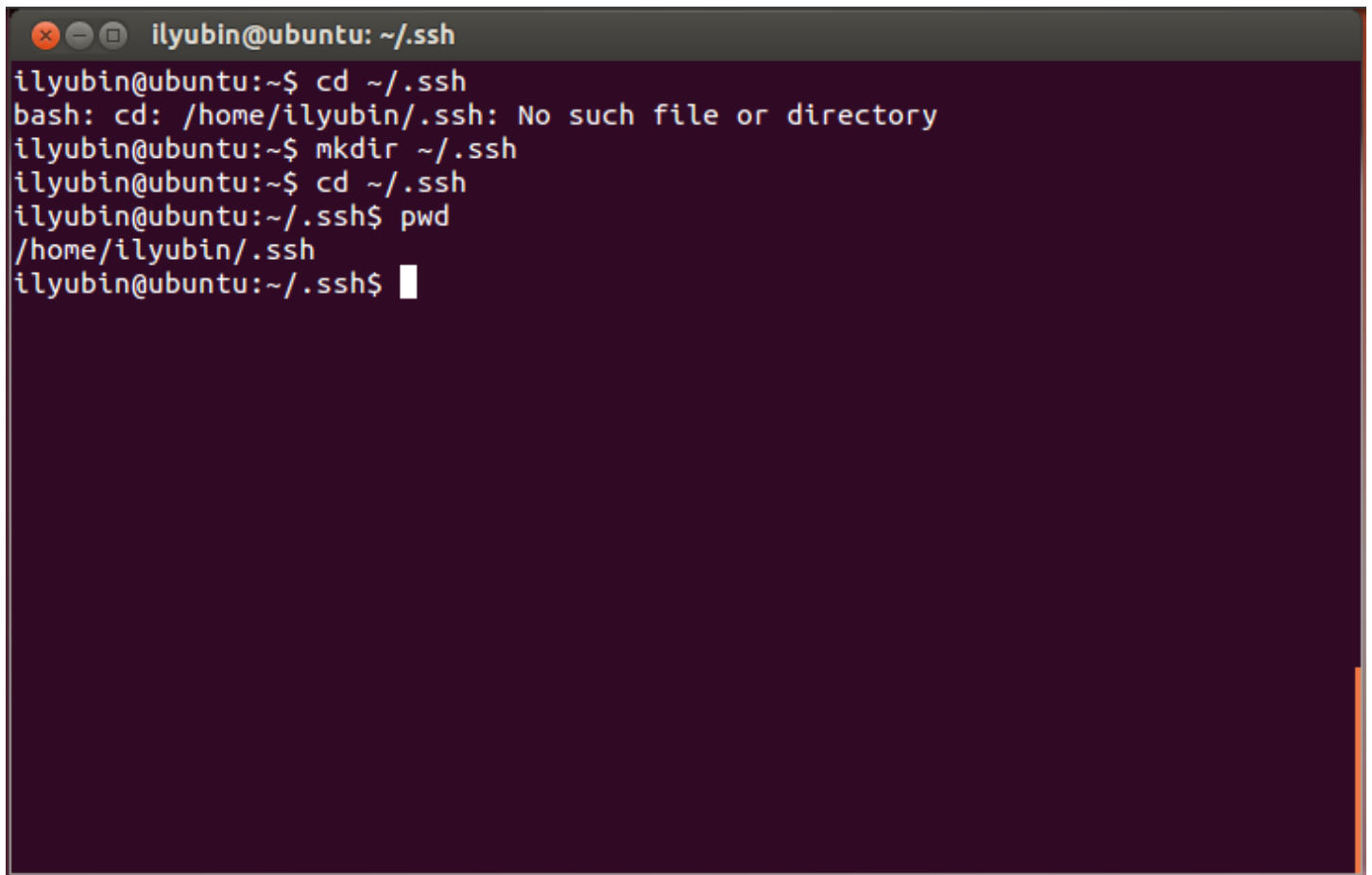
Перед тем, как регистрироваться на GitHub, следует вначале сгенерировать SSH-ключ шифрования. Этот ключ необходим, чтобы быстро устанавливать соединение с GitHub, не вводя пароля. Без такого ключа GitHub просто не будет работать.

В операционной системе вначале нужно перейти в каталог ~/.ssh.

```
cd ~/.ssh
```

Если такого каталога нет, то его нужно создать.

```
mkdir ~/.ssh
```

A terminal window with a dark purple background and a grey title bar. The title bar contains three window control icons (close, minimize, maximize) and the text 'ilyubin@ubuntu: ~/.ssh'. The terminal shows a series of commands and their outputs: 'cd ~/.ssh' results in an error 'bash: cd: /home/ilyubin/.ssh: No such file or directory'; 'mkdir ~/.ssh' is executed successfully; 'cd ~/.ssh' is executed successfully; 'pwd' returns '/home/ilyubin/.ssh'. The prompt 'ilyubin@ubuntu: ~/.ssh\$' is shown at the bottom with a white cursor.

```
ilyubin@ubuntu: ~/.ssh
ilyubin@ubuntu:~$ cd ~/.ssh
bash: cd: /home/ilyubin/.ssh: No such file or directory
ilyubin@ubuntu:~$ mkdir ~/.ssh
ilyubin@ubuntu:~$ cd ~/.ssh
ilyubin@ubuntu:~/.ssh$ pwd
/home/ilyubin/.ssh
ilyubin@ubuntu:~/.ssh$
```

Далее даем команду, чтобы сгенерировать ключи:

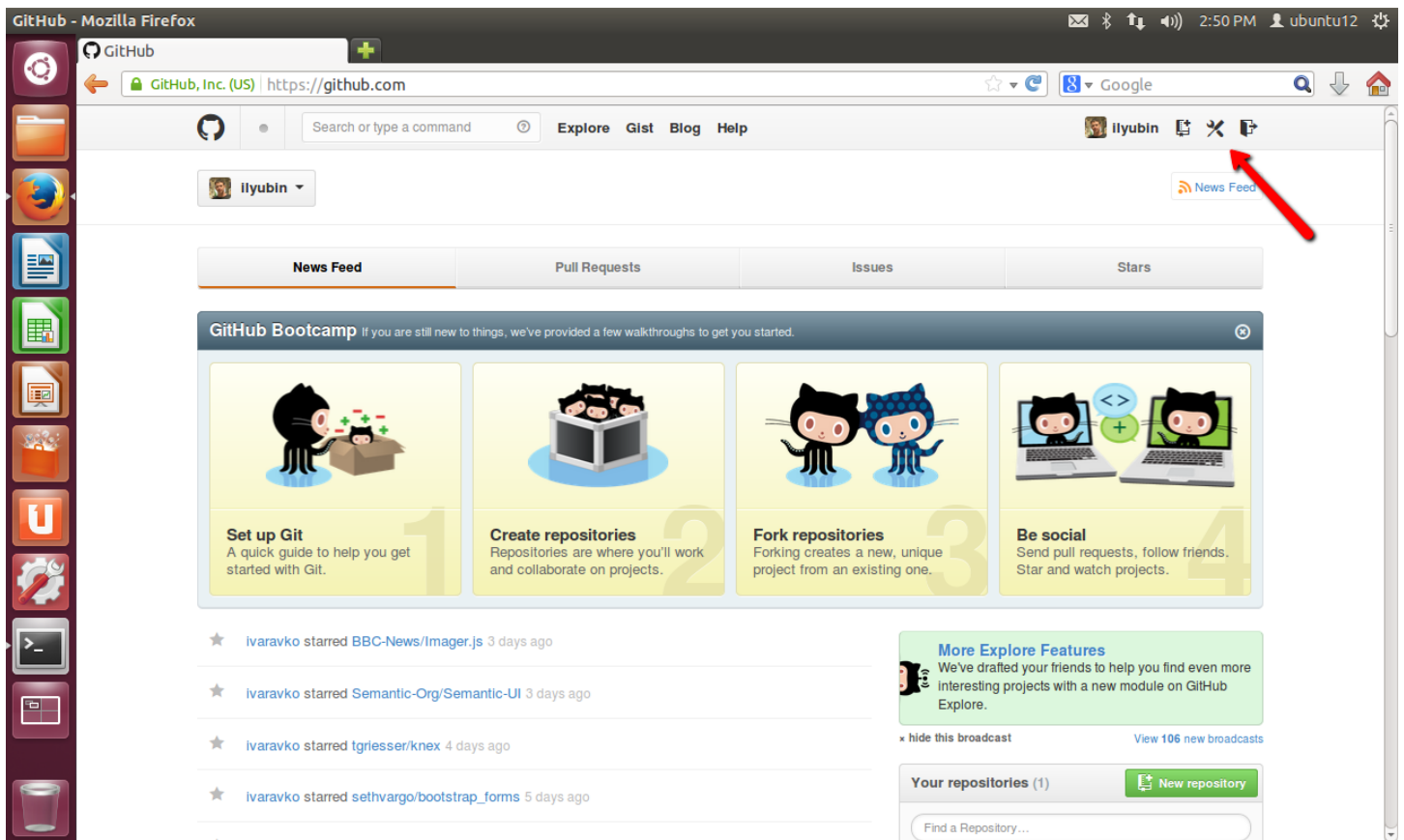
ssh-keygen -t rsa -C 'myemail@mail.ru'

Вместо myemail@mail.ru нужно указать свой email. В процессе генерации ключа у вас спросят куда положить файлы, в ответ просто нажимаем Enter. При запросе пароля просто нажимаем Enter. После генерации, в каталоге ~/.ssh должны появиться файлы id_rsa и id_rsa.pub, они нам пригодятся в дальнейшем.

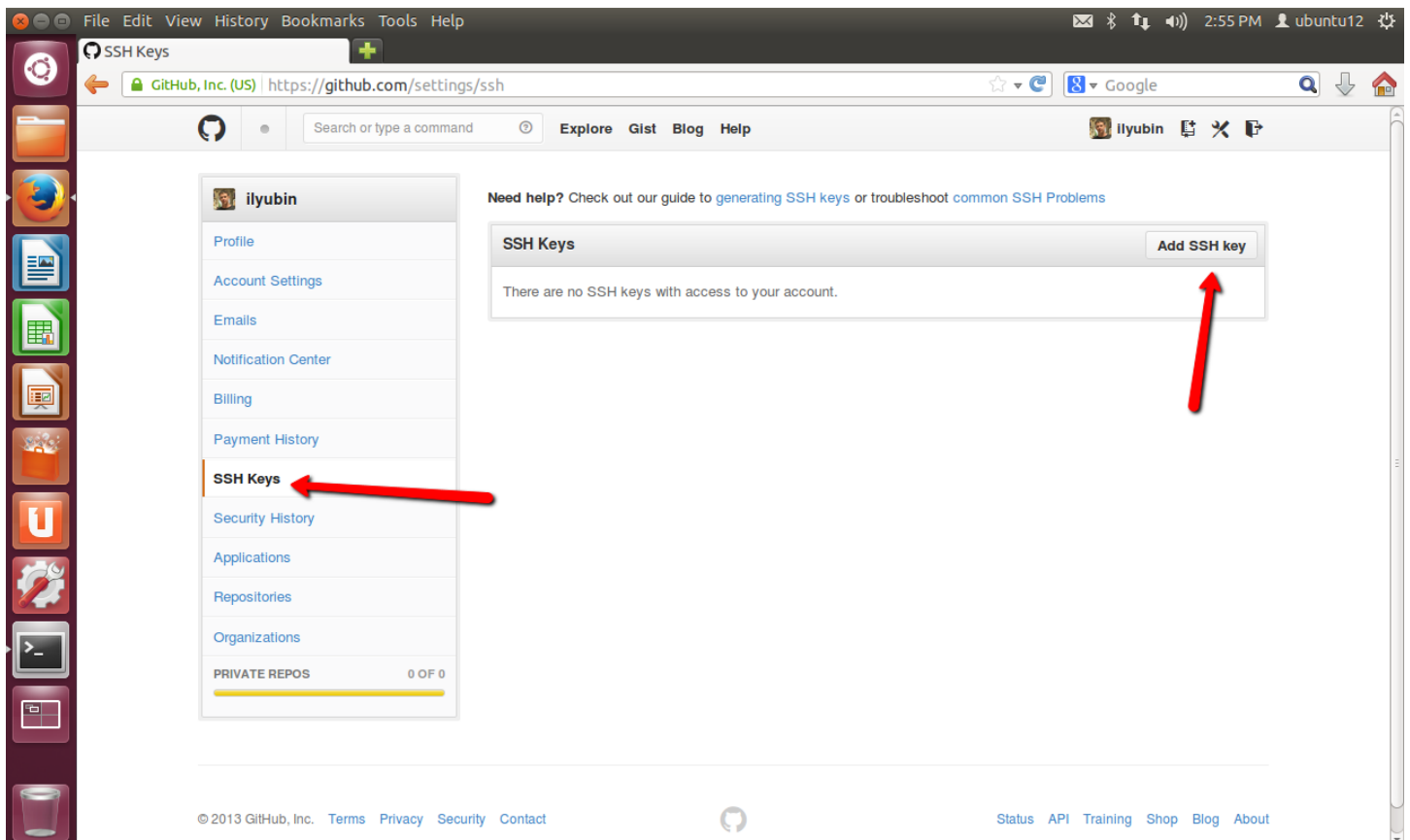
```
ilyubin@ubuntu: ~/.ssh$ ssh-keygen -t rsa -C 'ilyubin@inbox.ru'
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ilyubin/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ilyubin/.ssh/id_rsa.
Your public key has been saved in /home/ilyubin/.ssh/id_rsa.pub.
The key fingerprint is:
77:0f:92:f2:26:d5:dd:e7:0f:bc:47:fd:5d:f9:13:d4 ilyubin@inbox.ru
The key's randomart image is:
+--[ RSA 2048 ]-----+
|
|
|
|      .
|    o . oE|
|   S = + o =|
|  = o + +=|
| . o   +.B|
|  o     +*|
|       ..O|
+-----+
ilyubin@ubuntu:~/.ssh$ ls
id_rsa id_rsa.pub
ilyubin@ubuntu:~/.ssh$
```

Установка SSH-ключа в GitHub

Сразу после регистрации необходимо прописать в системе GitHub свой публичный ключ шифрования (открытый SSH-ключ). Для добавления ключа, надо в правом верхнем углу нажать "Account Settings":

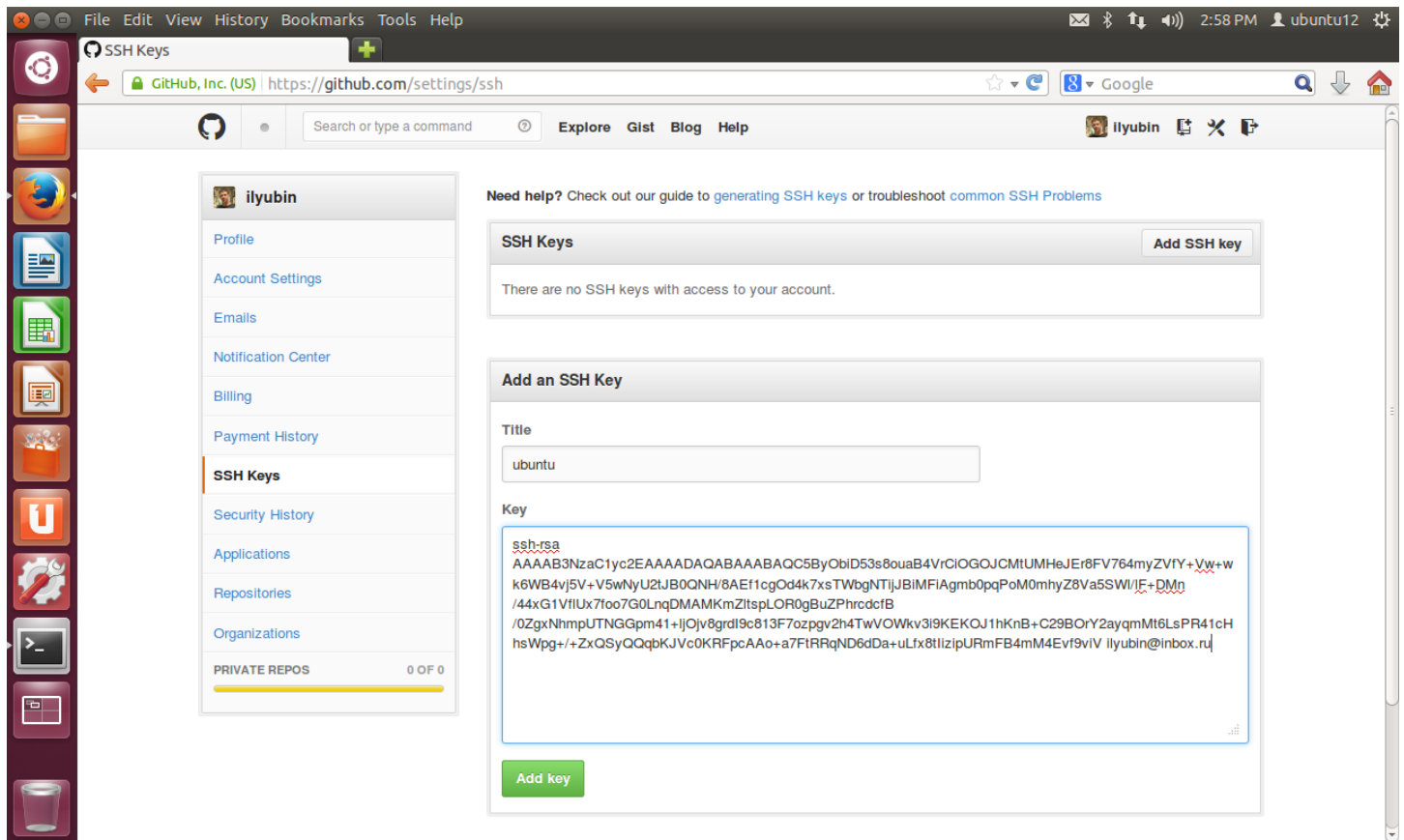


В открывшемся окне нужно кликнуть на пункт меню "SSH Keys".



Нажать "Add SSH key". Появится два поля - название ключа (Title) и содержимое ключа (Key).

В поле Title можно написать название компьютера, на котором сгенерирован публичный ключ. Можно писать по-русски.



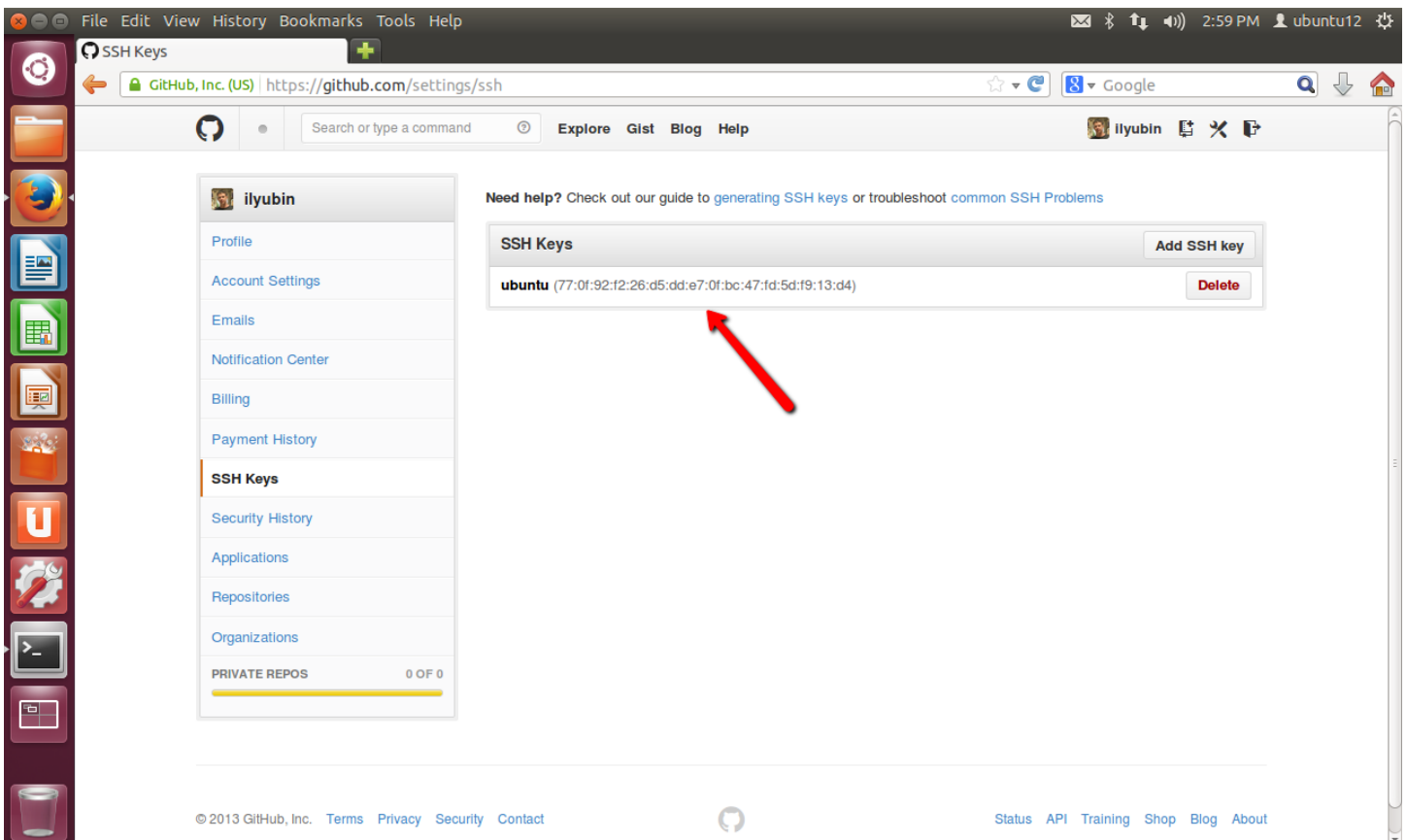
В поле Key надо вставить содержимое файла `id_rsa.pub`. Помните, в каком каталоге они находятся? Переходим в этот каталог, открываем любым текстовым редактором файл `id_rsa.pub` (именно с расширением `.pub`, не перепутайте). Можно воспользоваться командой:

```
cat id_rsa.pub
```

Выделяем весь текст, копируем, и вставляем на странице GitHub в поле Key.

```
ilyubin@ubuntu: ~/.ssh
ilyubin@ubuntu:~/.ssh$ cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQBAQC5By0biD53s8ouaB4VrCiOG0JCmtUMHeJEr8FV764m
yZVfY+Vw+wk6WB4vj5V+V5wNyU2tJB0QNH/8AEf1cg0d4k7xsTWbgNTijJBimFiAgmb0pqPoM0mhyZ8V
a5SWl/lF+DMn/44xG1VfLUX7foo7G0LnqDMAMKmZltspLOR0gBuZPhrcdcfB/0ZgxNhmpUTNGGpm41+l
j0jv8grdI9c813F7ozpgv2h4TwVOWkv3i9KEK0J1hKnB+C29B0rY2ayqmMt6LsPR41cHhsWpg+/+ZxQS
yQQqbKJVc0KRFpcAAo+a7FtRRqND6dDa+uLfx8tIizipURmFB4mM4Evf9viV ilyubin@inbox.ru
ilyubin@ubuntu:~/.ssh$
```

После добавления ключа, компьютер может соединяться с GitHub через программу git, и никаких ошибок не должно возникать.



Создание репозитория на GitHub

Теперь пришло время создать свой первый репозиторий на GitHub. Репозиторий можно рассматривать просто как директорию, в которой будут лежать синхронизируемые файлы и поддиректории. Создавать репозиторий нужно в web-интерфейсе GitHub, а наполнять его файлами и работать с ним можно будет уже с помощью программы git на своем компьютере.

Для создания репозитория, нужно в правом верхнем углу нажать "Create new repo", и далее заполняем форму согласно видео инструкции от Алексея:

Create a New Repository - Mozilla Firefox

Create a New Repository

GitHub, Inc. (US) | <https://github.com/new>

Search or type a command | Explore | Gist | Blog | Help

ilyubin

Owner: ilyubin

Repository name: pft-16-igor.lybulin

Great repository names are short and memorable. Need inspiration? How about [north-american-octo-dubstep](#).

Description (optional): Programming for Testers

Public: Anyone can see this repository. You choose who can commit.

Private: You choose who can see and commit to this repository.

☒ Initialize this repository with a README
This will allow you to `git clone` the repository immediately.

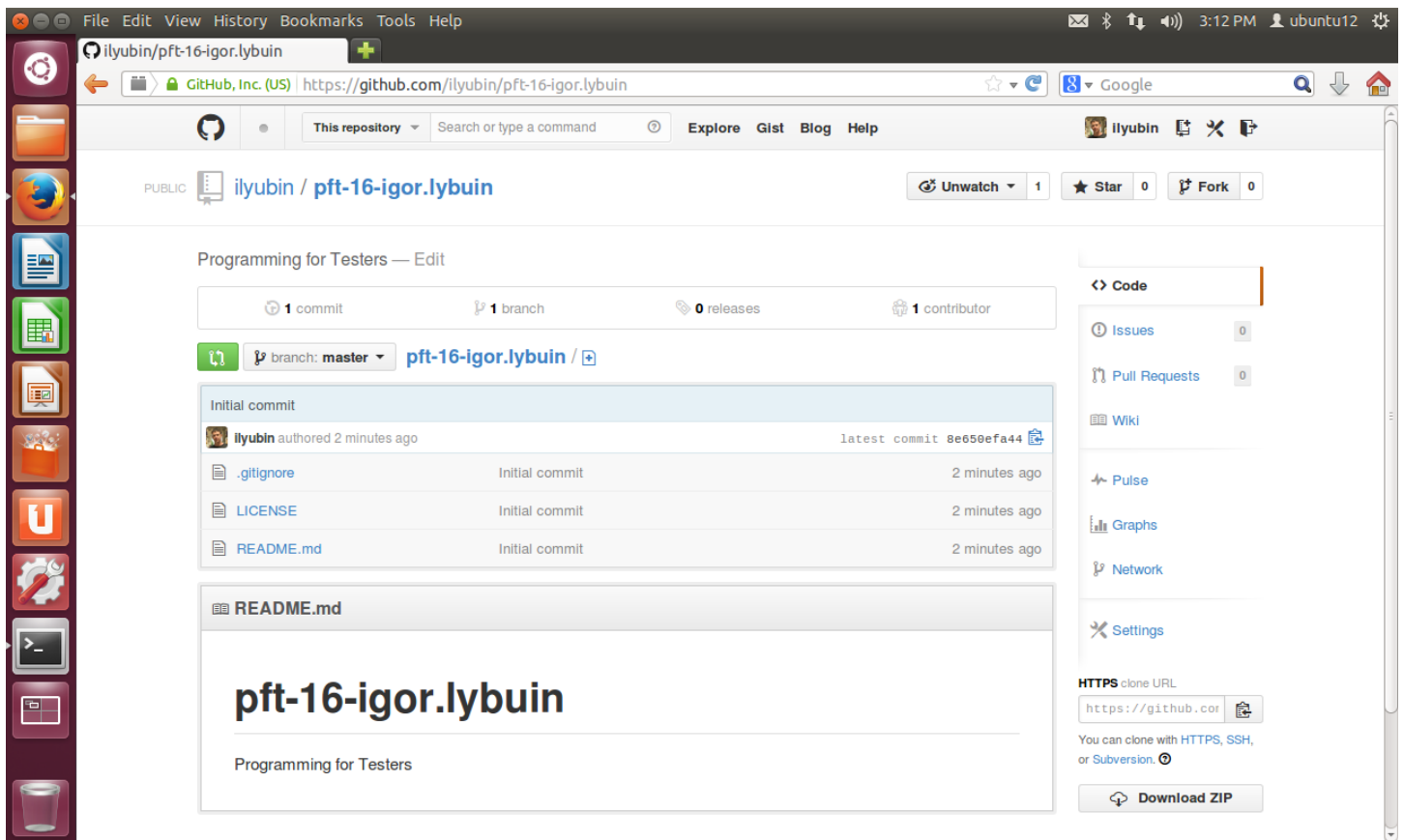
Add .gitignore: Java | Add a license: Apache v2 License

Create repository

© 2013 GitHub, Inc. | [Terms](#) | [Privacy](#) | [Security](#) | [Contact](#)

Status | API | Training | Shop | Blog | About

Создайте README файл, .gitignore файл для Java и выберите (любую) лицензию. В итоге у вас должно все получиться примерно так:



Работа с репозиторием на GitHub через программу Git

Начиная с этого момента, пляски вокруг web-интерфейса GitHub можно считать законченными. Далее можно работать только используя программу git.

Вначале нужно сделать небольшую настройку программы git: указать локальной системе git имя пользователя и email. Это делается следующими командами, которые можно выполнить, находясь в любом каталоге:

```
git config --global user.name "YourFullName"
```

```
git config --global user.email myemail@mail.ru
```

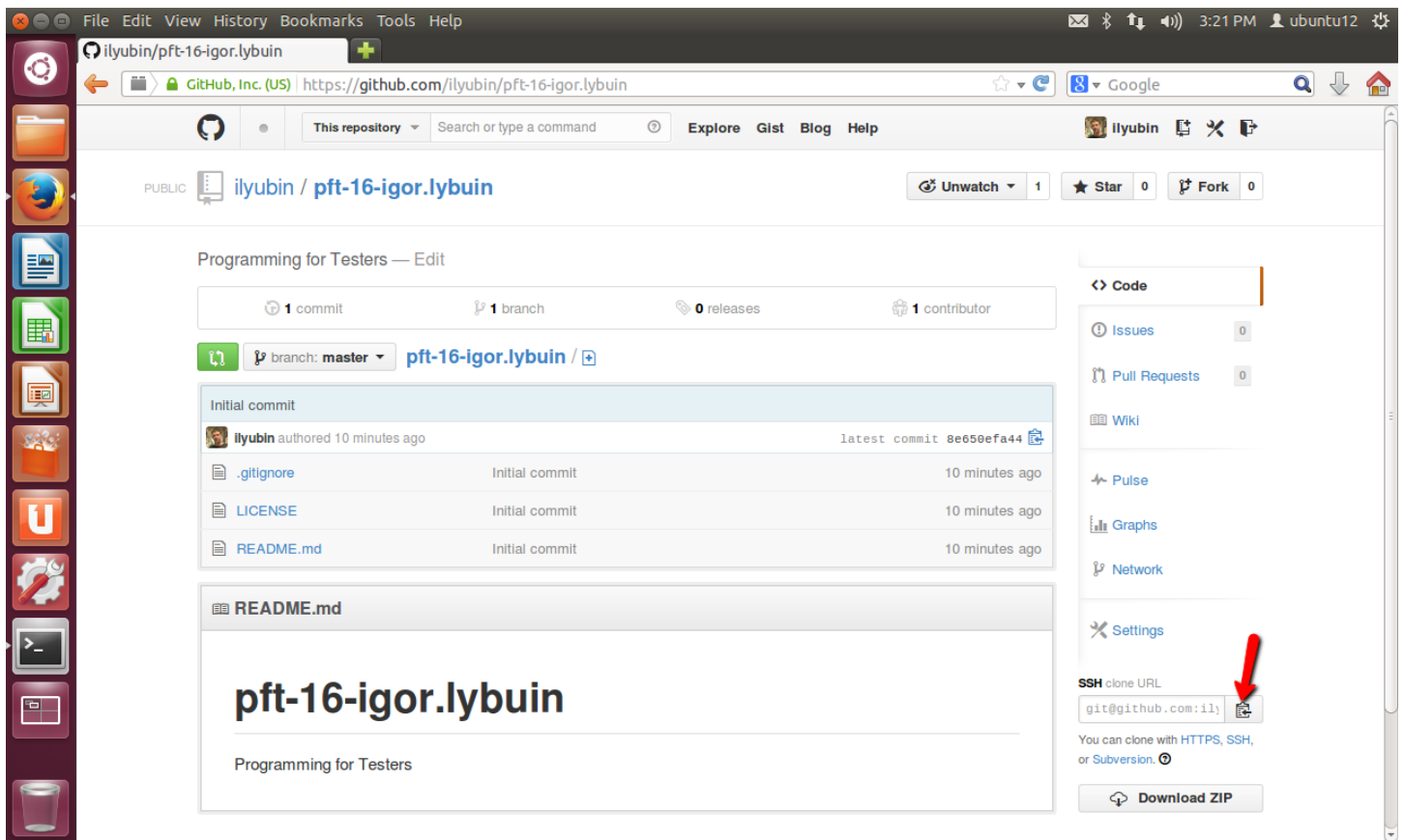
где вместо YourFullName нужно написать свое имя, а вместо myemail@mail.ru - свой email. Эти значения используются для логина на GitHub. Поэтому на месте YourFullName нужно указать ваш логин на GitHub-е, а на месте myemail@mail.ru нужно указать email, который вы вводили при генерации ключей шифрования.

Проверяем, что наши настройки применились:

```
git config --global --list
```

```
ilyubin@ubuntu: ~  
ilyubin@ubuntu: ~/.ssh$ cd ~  
ilyubin@ubuntu: ~$ git config --global user.name "igor.lyubin"  
ilyubin@ubuntu: ~$ git config --global user.email ilyubin@inbox.ru  
ilyubin@ubuntu: ~$ git config --global --list  
user.name=igor.lyubin  
user.email=ilyubin@inbox.ru  
ilyubin@ubuntu: ~$
```

После этих настроек, можно заливать свои файлы в репозиторий. Но для этого клонируем репозиторий. В веб интерфейсе копируем SSH ссылку к проекту:



В консоли переходим в каталог со своими проектами (у меня это папка dev), и даем команду:

git clone [SSH ссылка на проект, копируемая с веб]

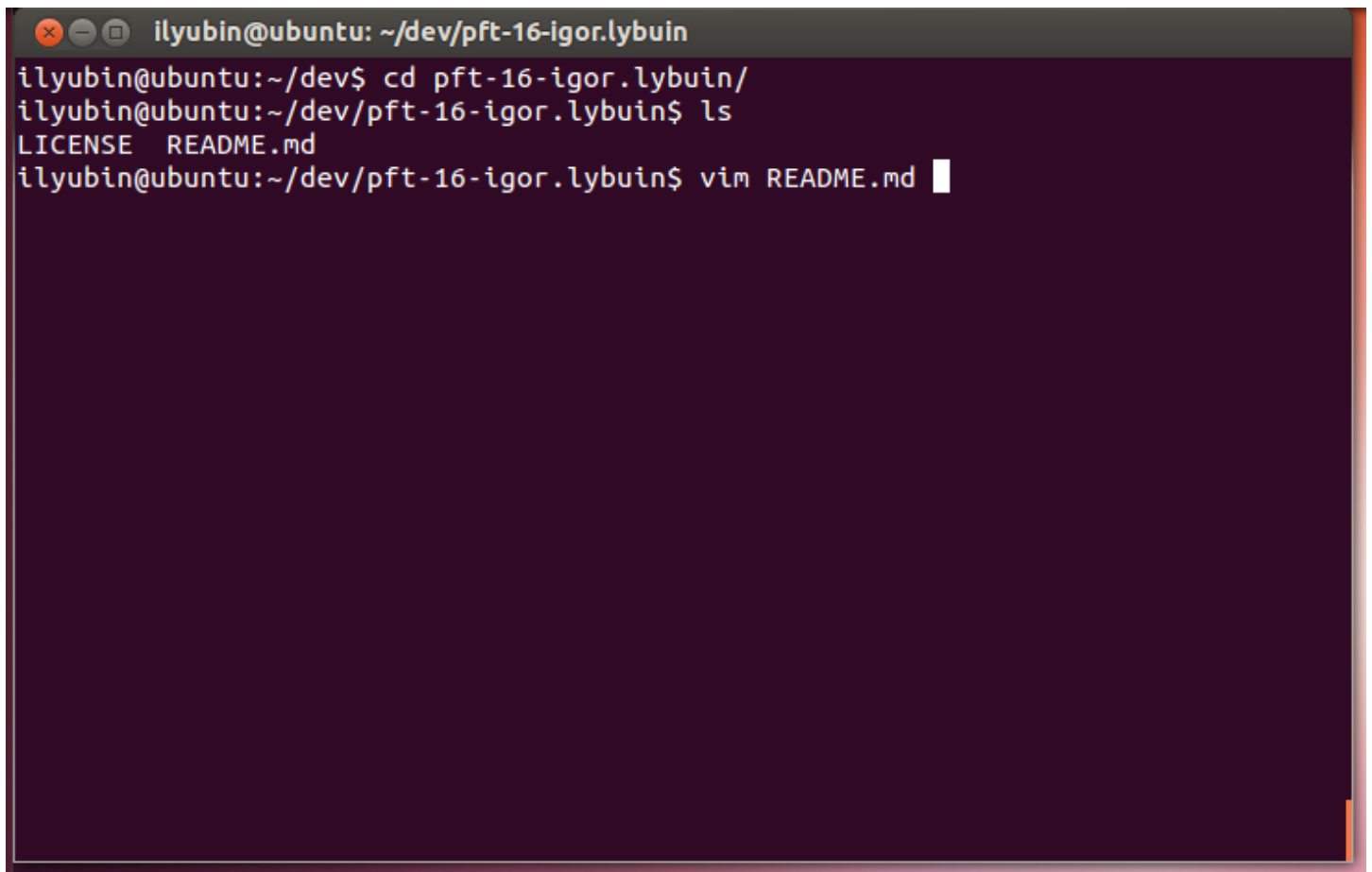
```
ilyubin@ubuntu: ~/dev
ilyubin@ubuntu:~$ mkdir dev
ilyubin@ubuntu:~$ cd dev/
ilyubin@ubuntu:~/dev$ git clone git@github.com:ilyubin/pft-16-igor.lybuin.git
Cloning into 'pft-16-igor.lybuin'...
The authenticity of host 'github.com (192.30.252.130)' can't be established.
RSA key fingerprint is 16:27:ac:a5:76:28:2d:36:63:1b:56:4d:eb:df:a6:48.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'github.com,192.30.252.130' (RSA) to the list of known hosts.
remote: Counting objects: 5, done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 0), reused 0 (delta 0)
Receiving objects: 100% (5/5), 4.22 KiB, done.
ilyubin@ubuntu:~/dev$ ls
pft-16-igor.lybuin
ilyubin@ubuntu:~/dev$
```

Отправка изменений в удаленный репозиторий

После выполнения домашнего задания необходимо отправлять изменения в удаленный репозиторий Github. Эта операция происходит в 2 этапа:

1. Данные коммитятся (git commit) в локальный репозиторий
2. Данные отправляются (git push) в удаленный репозиторий

Давайте попробуем сделать пробный коммит. Для этого перейдем в папку с проектом и поменяем README-файл:

A terminal window with a dark purple background and a grey title bar. The title bar contains window control icons and the text 'ilyubin@ubuntu: ~/dev/pft-16-igor.lybui'. The terminal shows a sequence of commands: 'cd pft-16-igor.lybui/', 'ls', and 'vim README.md'. The output of 'ls' shows 'LICENSE' and 'README.md'. The cursor is at the end of the 'vim' command.

```
ilyubin@ubuntu: ~/dev/pft-16-igor.lybui
ilyubin@ubuntu:~/dev$ cd pft-16-igor.lybui/
ilyubin@ubuntu:~/dev/pft-16-igor.lybui$ ls
LICENSE  README.md
ilyubin@ubuntu:~/dev/pft-16-igor.lybui$ vim README.md
```

Добавим строчку текста в файл «This is my first project» и сохраним файл.

[illegible]

Теперь выполняем первый шаг отправки данных в репозиторий, делаем локальный коммит. Даем команду:

```
git commit -a -m "Adding more documentation"
```

Здесь ключик `-a` говорит чтобы программа `git` добавила все файлы в коммит, ключик `-m` говорит с каким сообщением отправить коммит в репозиторий.

```
ilyubin@ubuntu: ~/dev/pft-16-igor.lybuin
ilyubin@ubuntu:~/dev/pft-16-igor.lybuin$ git status
# On branch master
# Changes not staged for commit:
#   (use "git add <file>..." to update what will be committed)
#   (use "git checkout -- <file>..." to discard changes in working directory)
#
#       modified:   README.md
#
no changes added to commit (use "git add" and/or "git commit -a")
ilyubin@ubuntu:~/dev/pft-16-igor.lybuin$ git commit -a -m "Adding more documenta
tion"
[master 6ac2aa1] Adding more documentation
 1 file changed, 2 insertions(+)
ilyubin@ubuntu:~/dev/pft-16-igor.lybuin$
```

Затем отправляем данные в удаленный репозиторий и даем команду:

git push origin HEAD


```
ilyubin@ubuntu: ~/dev/pft-16-igor.lybuin
ilyubin@ubuntu:~/dev/pft-16-igor.lybuin$ git push origin HEAD
Counting objects: 5, done.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 330 bytes, done.
Total 3 (delta 1), reused 0 (delta 0)
To git@github.com:ilyubin/pft-16-igor.lybuin.git
8e650ef..6ac2aa1 HEAD -> master
ilyubin@ubuntu:~/dev/pft-16-igor.lybuin$
```

Заходим в Github и проверяем коммиты.

The screenshot shows the GitHub web interface for the repository `ilyubin / pft-16-igor.lybuin`. The repository is public and has 1 contributor. A red arrow points to the '2 commits' link. Below this, a table lists recent commits:

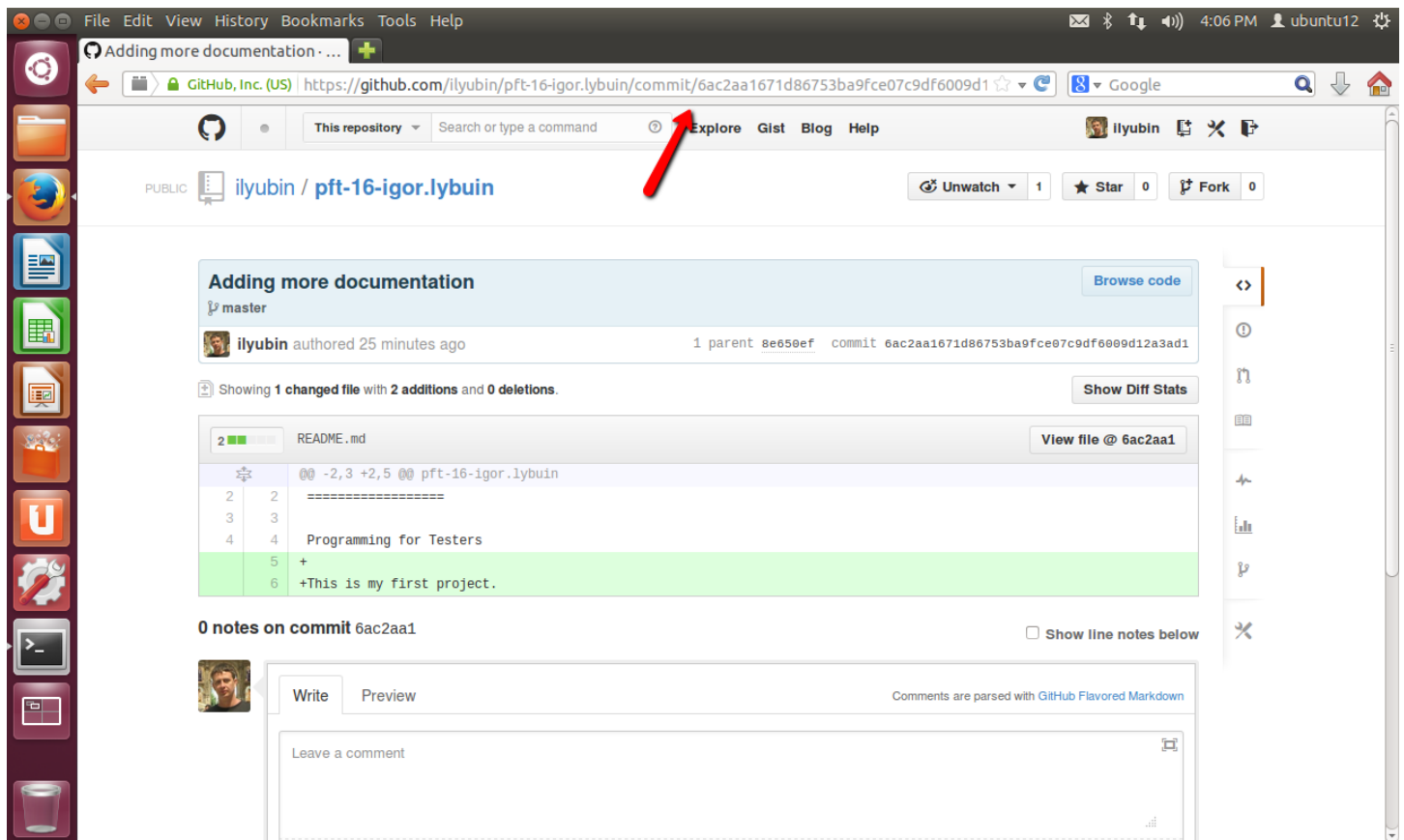
Commit	Author	Time
<code>6ac2aa1</code>	ilyubin	7 minutes ago
<code>8e650ef</code>	ilyubin	39 minutes ago

The README file is displayed below the commit list, showing the title `pft-16-igor.lybuin` and the content:

```
Programming for Testers

This is my first project.
```

Находим наш отправленный коммит и отправляем тренеру ссылку на проверку домашнего задания.



Работа на нескольких компьютерах

При работе на нескольких компьютерах. Например: из дома и с работы. Необходимо перед тем как приступить работать на новой машине производить синхронизацию данных. И забирать с удаленного репозитория последние изменения. Это делается с помощью команды:

git pull --rebase

ilyubin@ubuntu: ~/dev/pft-16-igor.lybuin

```
ilyubin@ubuntu:~/dev/pft-16-igor.lybuin$ git pull --rebase
```

Current branch master is up to date.

```
ilyubin@ubuntu:~/dev/pft-16-igor.lybuin$
```