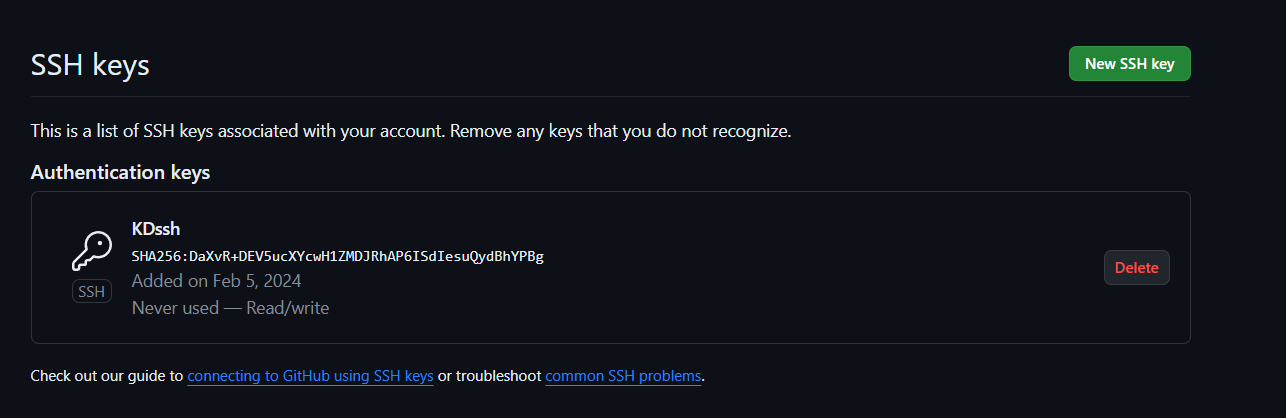
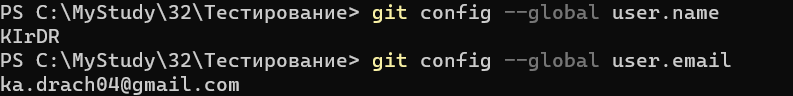
I Can Win

1. Установите git и сгенерируйте пару ssh ключей. Авторизуйте публичный ключ на github.com.  
  


|  |
| --- |
| PS C:\MyStudy\32\Тестирование> ssh-keygen  Generating public/private rsa key pair.  Enter file in which to save the key (C:\Users\kadra/.ssh/id\_rsa): klrdrssh  Enter passphrase (empty for no passphrase):  Enter same passphrase again:  Your identification has been saved in klrdrssh  Your public key has been saved in klrdrssh.pub  The key fingerprint is:  SHA256:7sCtyd1yBP/dE5FyaWWUbV6PKcIPuu76HrYKmbFbtqY kadra@LAPTOP-220I1R3V  The key's randomart image is:  +---[RSA 3072]----+  | .+|  | .\*|  | . \*\*|  | . + ..o\*o|  | . S+ + .+ .|  | \* o. o . . |  | = = =o . . ..|  | \*.Oo+. . ...|  | EoBBOo. .|  +----[SHA256]-----+  PS C:\MyStudy\32\Тестирование> Generating public/private rsa key pair.  Enter file in which to save the key (C:\Users\kadra/.ssh/id\_rsa): KDssh  Enter passphrase (empty for no passphrase):  Enter same passphrase again:  Your identification has been saved in KDssh  Your public key has been saved in KDssh.pub  The key fingerprint is:  SHA256:DaXvR+DEV5ucXYcwH1ZMDJRhAP6ISdIesuQydBhYPBg kadra@LAPTOP-220I1R3V  The key's randomart image is:  +---[RSA 3072]----+  | E\*o o.++OX=|  | o oo . = Bo\*=|  | o.= \* = . \* .|  | . + \* X = |  | o o S = o |  | o . . |  | . . |  | . |  | |  +----[SHA256]-----+  PS C:\MyStudy\32\Тестирование> |

2. Укажите свой user.name и user.email в git.



3. Используя веб-интерфейс github’а создайте новый public репозиторий git-demo содержащий README файл (Initialize with a README) на github.com.

4. Склонируйте созданный репозиторий git-demo на свой компьютер.

5. Создайте текстовый файл названием song.txt и поместите туда фрагмент текста песни:

sword and spear

wall of shields

6. Сделайте коммит с названием "add first two lines in the song".

7. Отправьте локальные изменения на сервер.

8. Убедитесь, что на github’е есть файл song.txt с текстом песни.

9. Используя веб-интерфейс github’а добавьте вторую половину текста песни в файл song.txt:

standing strong

on this their chosen battlefield

10. Сделайте коммит с названием "add two more lines to the song".

11. Выполните синхронизацию локального репозитория с github’ом (“накатите” актуальную версию мастера с github’а выполнив команду pull в локальном репозитории).

12. Убедитесь, что коммит, который вы создали на github’е, подтянулся и у вас полный текст песни в файле song.txt.

Bring It On

Данное задание выполняется сразу после предыдущего (I Can Win).

1. Добавьте в проект файл .gitignore и настройте так чтобы скрыть файлы с расширением .db, .log и директории с названиями target или bin.

2. Создайте ветку feature и добавьте в неё два коммита

3. Смержите ветку feature в master

4. Вернитесь в feature и создайте файл arrows.txt cледующего содержания:

The ship glides gently on the waves

As day turns into night

Выполните коммит.

5. Перейдите в master. Создайте там файл arrows.txt и добавьте следующий текст:

One thousand burning arrows

Fill the starlit sky

Выполните коммит.

6. Смержите feature в master решив конфликт: сохраните все 4 строки в файле arrows.txt в порядке их добавления в пунктах 4 и 5.

Hurt Me Plenty

Данное задание выполняется после предыдущего (Bring It On).

1. Создайте ветку storm и добавьте коммит в файл storm.txt:

Twenty ships with Norsemen braves

Riding the northern wind

2. Добавьте еще 2 строки в storm.txt и сделайте еще один коммит:

They left their shores at early dawn

As a red sun was rising in the east

3. Вернитесь в master и создайте файл pursuit.txt с текстом ниже:

The warming sun returns again

And melts away the snow

The sea is freed from icy chains

Winter is letting go

Выполните коммит.

4. Отметьте коммит тегом session1 и перейдите в ветку storm

5. Сделайте rebase ветки storm так чтобы она содержала последний коммит из мастера.

Hardcore

Данное задание выполняется сразу после предыдущего (Hurt Me Plenty).

1. Сделайте push вашего репозитория и убедитесь, что все коммиты есть на github.

2. Сделайте новый репозиторий на github.

3. Смените remote в локальном репозитории так, чтобы fetch и push шел на новый репозиторий который был создан в предыдущем шаге.

4. Сделайте push и убедитесь, что второй репозиторий на гитхабе выглядит так же, как и первый.

5. Верните настройки remote в исходное состояние: пул и пуш первого локального репозитория ведет в один удаленный репозиторий на гитхабе.

Nightmare!

Чтобы выполнить это задание, вам нужно будет повторить все 4 предыдущих используя только командную строку MINGW64 (в которой работает гит). Вы не можете использовать файловый менеджеры, текстовые редакторы с графическим интерфейсом, gitk / git gui. Вам разрешается использовать браузер только для операций с github. Все остальное только через консоль. Это самая сложная задача в этом курсе.

Команды которые могу вам пригодиться:

echo

cd

cat

vi