МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Кафедра «Вычислительной техники»

ОТЧЕТ

по лабораторной работе №9

«GitHub»

Выполнили: ст.гр 23ВА1

Бедин З.Р.

Буйдин К.Е.

Якунин В.А.

Приняли: Юрова О.В.

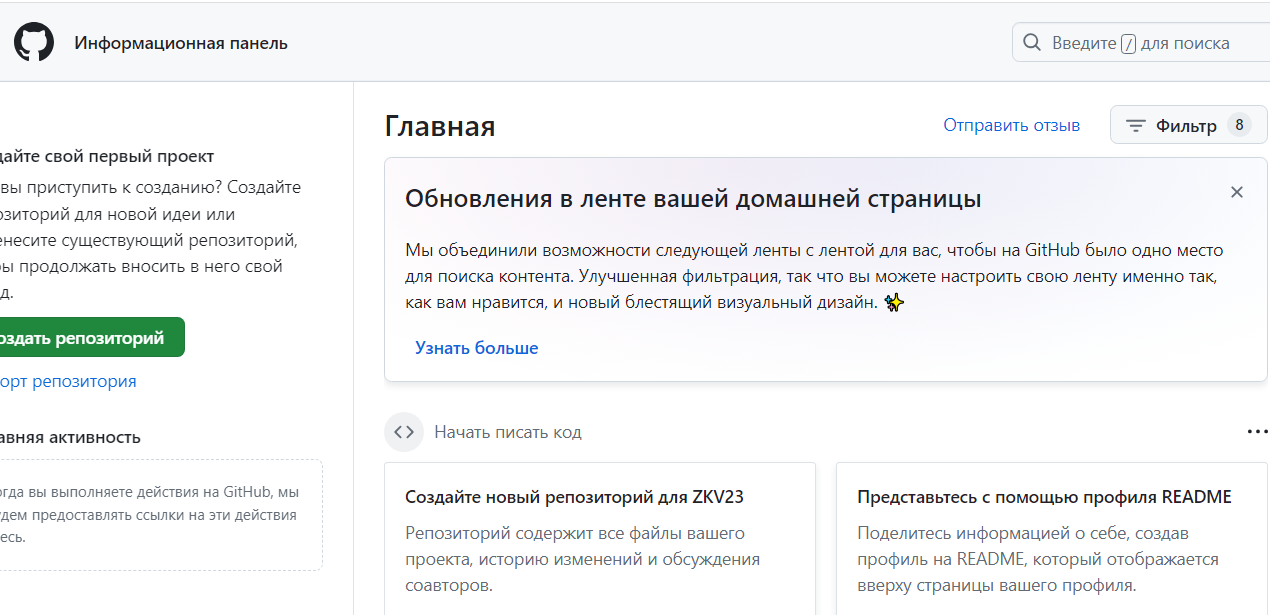
Митрохина Н.Ю.

2024

Цель: научиться работать с веб-сервисом для хостинга проектов и их совместной разработки GitHub. Создать свой публичный репозиторий.

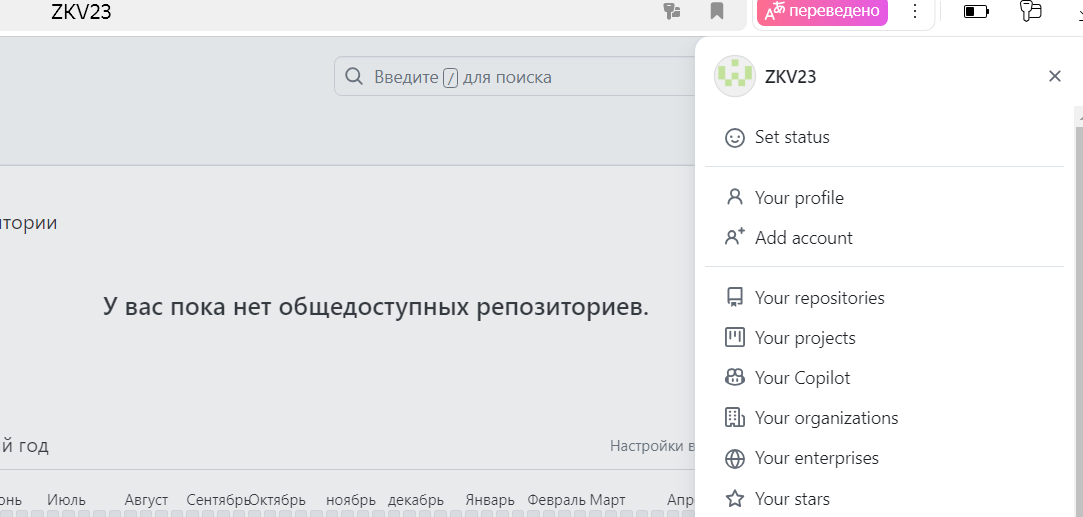
#### 1) Регистрация

Зарегистрировались



#### 2) Основные настройки профиля

Посмотрели основные настройки



#### 3) Создание нового репозитория

Для создания нового репозитория перейдите в пункт «Repositories» и нажмите кнопку «New».

Или нажмите в правом верхнем углу на кнопку «+» и выберите пункт меню «New repository».

Следующим шагом укажите основные параметры репозитория:

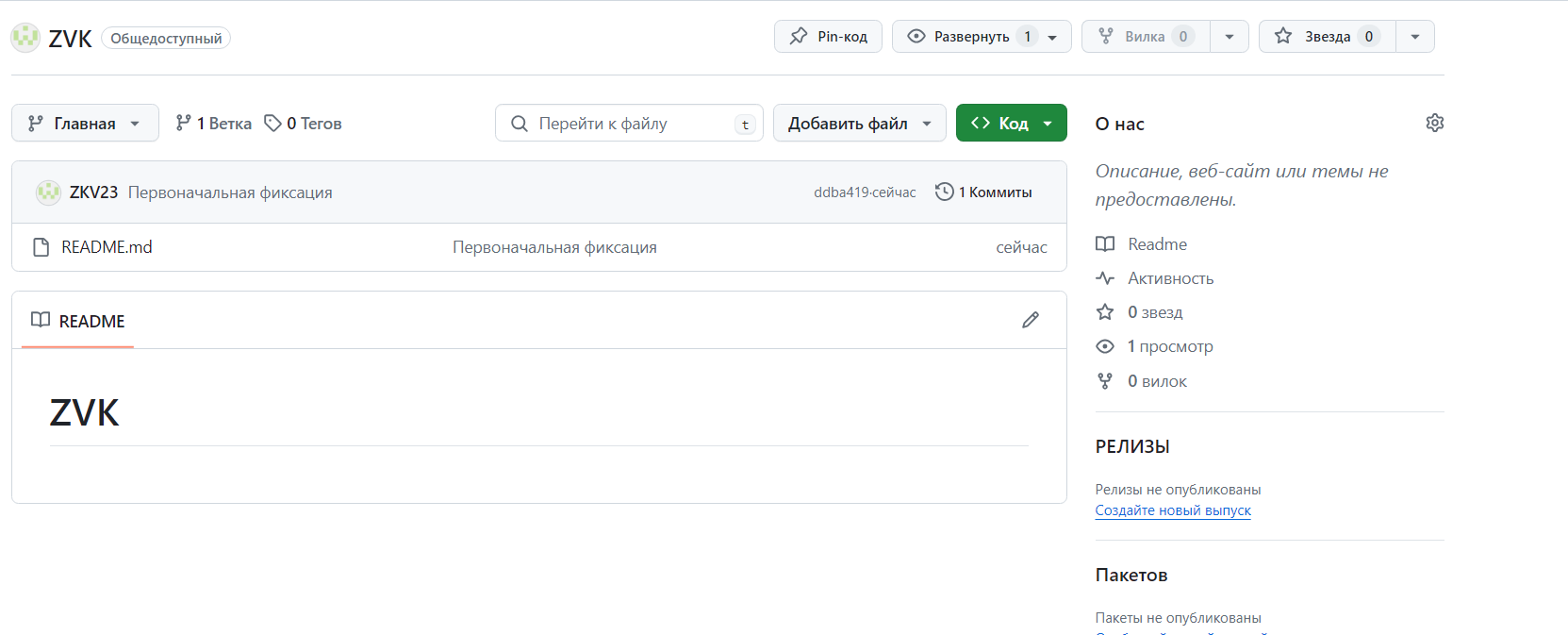
Name – название репозитория (укажем имя «TestRepo»);

Description – описание (можно оставить пустым);

Тип репозитория – публичный/приватный (выбираем публичный);

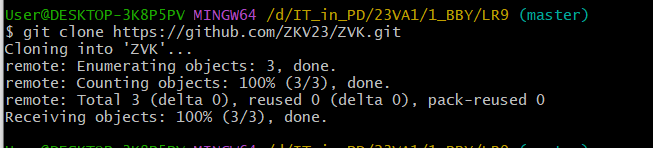
README file – стандартный для репозиториев файл с описанием (ставим галочку);

gitignore – файл со списком игнорируемых расширений, файлов, папок, описанных в нем.

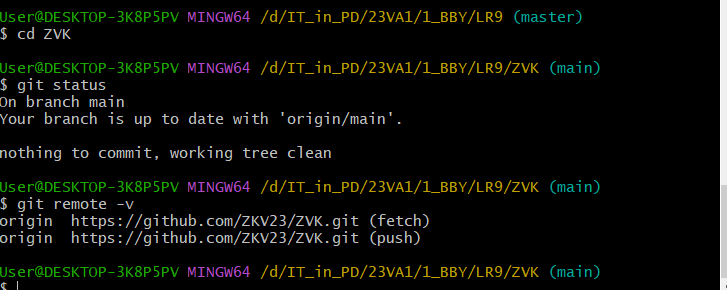


#### 4) Клонирование репозитория

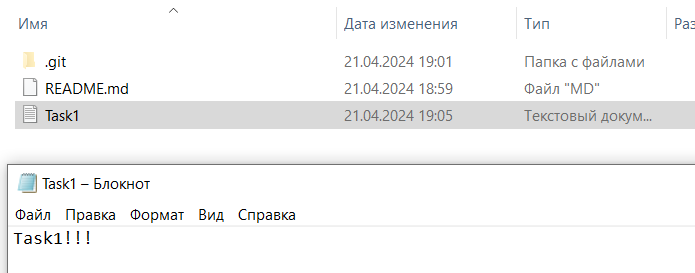
прописываем команду git clone «ссылка на репозиторий»



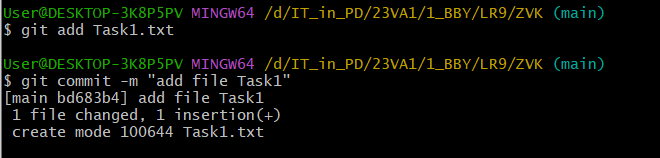
Автоматически Github подключит к локальному репозиторию удалённый и назовёт его origin. Перейдите в папку TestRepo с помощью команды cd и проверьте это командой git remote –v . В нашем локальном репозитории автоматически создалась локальная ветка main, которую связана с веткой main в удаленном репозитории на Github. Git автоматически создает ветку master, а Github при связывании создает main и клонирует её в локальный репозиторий, т.е. у нас сразу появляется века с именем main и ничего дополнительно делать не нужно. Проверьте это командой git status.



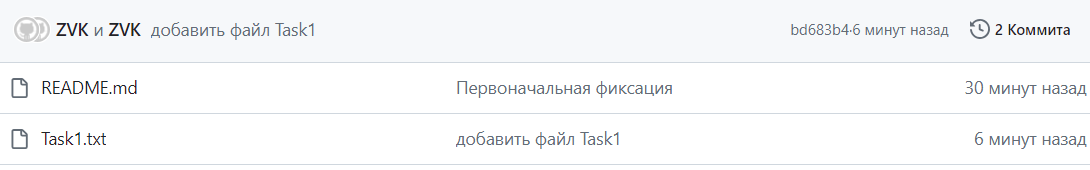
Теперь рассмотрим, как синхронизировать локальный репозиторий (клон) с оригиналом, который находится на Github. В клонированной папке TestRepo создайте локально файл Task1.txt, напишите в нем текст «Task1!!!», закройте его, сохранив изменения.



Для того чтобы зафиксировать изменения, выполните команду git add Task1.txt и создайте коммит

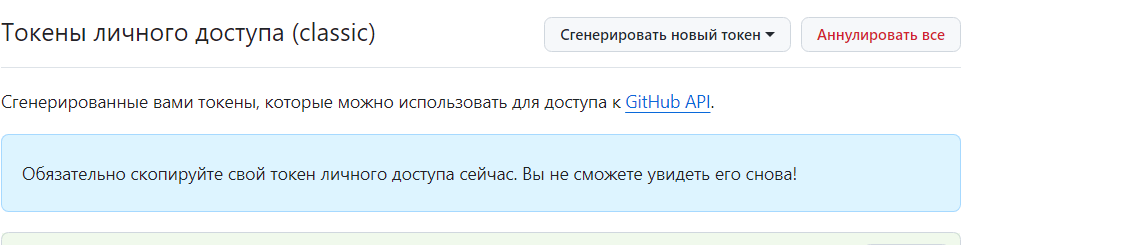


Выполните команду git push. Эта команда берет имеющиеся локальные изменения и отправляет их в удалённый репозиторий.



#### 5) Создание токена персонального доступа

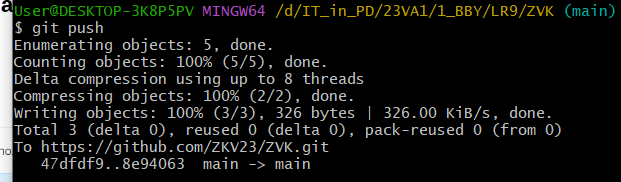
Создали токен



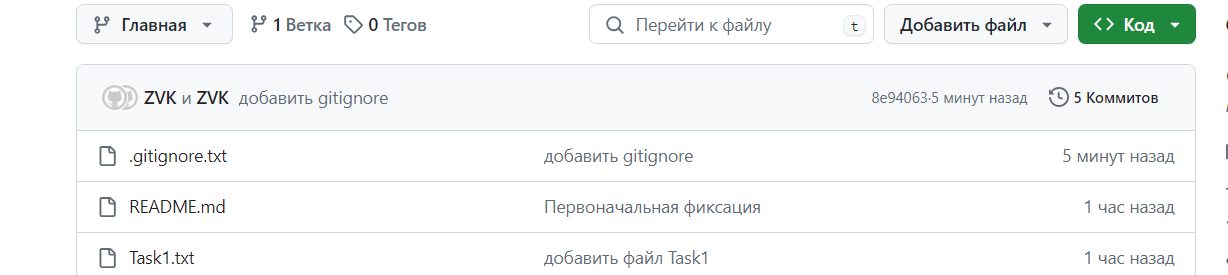
#### 6) Добавление .gitignore

Добавили файл

Проиндексируем изменения и сделаем коммит. После выполним команду git push

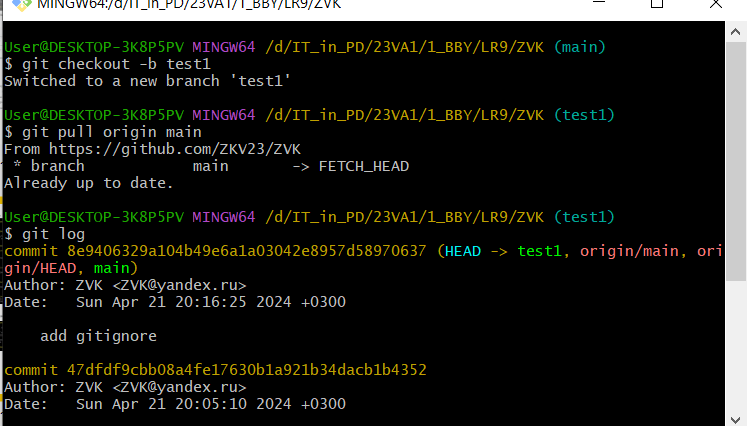


Проверим репозиторий на GitHub.

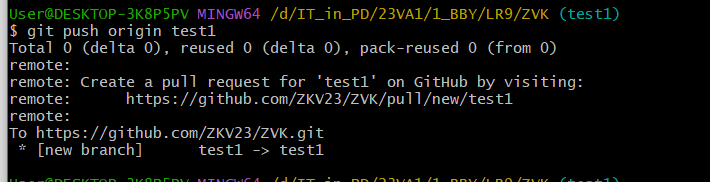


#### 7) Получение данных

Для того чтобы получить данные с удаленного репозитория используется команда git pull – извлечь из указанного удаленного репозитория копию текущей ветки и немедленно объединить ее с локальной копией. Для примера создадим новую ветку и получим изменения из главной ветки, проверим командой git log.

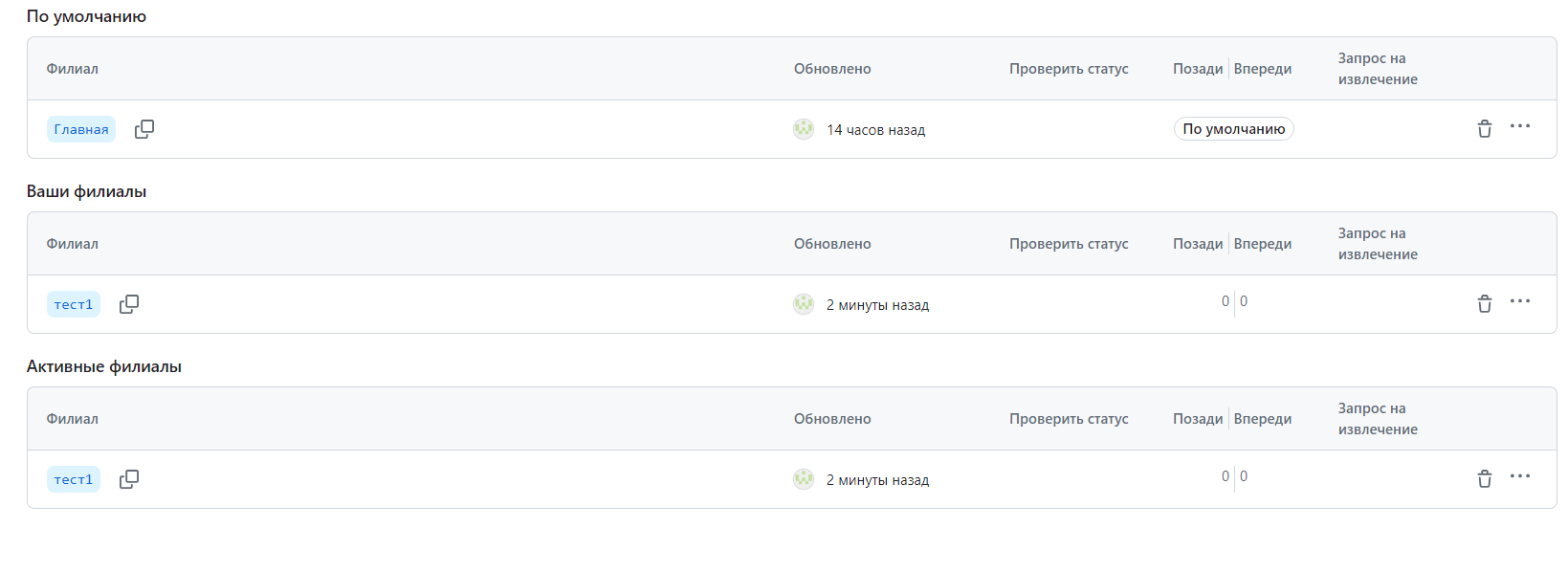


После отправим эту ветку на удаленный репозиторий.



Проверим репозиторий на GitHub. Видим, что в репозитории 2 ветки. Сейчас активна ветка main. Выберем из выпадающего списка ветку test1.

Здесь можно увидеть все ветки, какая из них активна в данный момент.



Вывод: научились работать с веб-сервисом для хостинга проектов и их совместной разработки GitHub. Создали свой публичный репозиторий.