Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Ульяновский государственный технический университет»

Кафедра «Вычислительная техника»

**Разработка профессиональных приложений**

**Лабораторная работа №5**

**GIT**

Выполнил

Студент группы ИВТИИИбд-22

Лысак В.И

Ульяновск

2025

**Задание по варианту**

1. Установить Git

2. Командами в КОНСОЛИ:

* 2.1. Создать локальный репозиторий
* 2.2. Добавить несколько файлов
* 2.3. Сделать один из файлов игнорируемым
* 2.4. Зафиксировать изменения
* 2.5. Внести изменения в файлы и зафиксировать изменения
* 2.6. Создать несколько бранчей, редактировать файлы в разных бранчах и фиксировать изменения. Осуществить слияние двух веток в одну.
* 2.7. Создать аккаунт github (если его нет, а если уже имеется, то использовать его), отправить исходные коды в удаленный репозиторий.
* 2.8. Склонировать удаленный репозиторий в папку, отличную от оригинальной.
* 2.9. Добиться конфликта файлов при применении изменений в оригинальном репозитории.
* 2.10. Разрешить конфликт.
* 2.11 Подготовить отчет о проделанной работе и прикрепить его в качестве ответа на это задание в learn. Отчет должен содержать список выполняемых в процессе выполнения лабораторной работы команд с их описанием, а также результат выполнения команд, скриншоты страниц с удаленного репозитория, чтоб были видны ветки и история операций (адрес репозитория также необходимо предоставить).

**Описание реализации**

**2.1. Создали локальный репозиторий**

cd rpp5\_git

git init

**2.2. Нужно добавить в локальный репозиторий несколько файлов, для этого нам понадобиться команда touch**

git add lab4rpp # Основной файл с кодом

git add data.csv # Файл с данными для тестирования

git add output.csv # Выходной файл (будет добавлен, но позже добавим его в .gitignore)

**2.3. Сделаем один файл игнорируемым, использовав команду touch и добавив имя файла в новый созданный файл**

# Добавляем .gitignore в репозиторий

git add .gitignore

**2.4. Теперь нужно зафиксировать все изменения**

git commit -m "Initial commit: added lab4rpp.py, data.csv and .gitignore"

**2.5. Нужно внести изменения в файлы и вновь зафиксировать изменения**

# Добавляем измененный файл

git add main.py

git commit -m "Added comments to main.py"

**2.6. Создали несколько веток, добавили и отредактировали файлы в каждой ветке, зафиксировали изменения, слили в одну ветку**

Внесем изменения в main.py (добавим новый метод в класс DirectoryAnalyzer)

Фиксируем изменения:

git add main.py

git commit -m "Added get\_checks\_by\_item\_name method to CheckCollection"

Вернемся в основную ветку и создадим другую ветку:

git checkout main

git branch bugfix/fix-amount-validation

git checkout bugfix/fix-amount-validation

Внесем изменения в валидацию суммы в классе Check:

git add lab4rpp.py

git commit -m "Improved amount validation in Check class"

Сливаем изменения из bugfix-ветки в основную:

git checkout main

git merge bugfix/fix-amount-validation

Теперь попробуем смержить feature-ветку:

git merge feature/add-new-methods

**2.7. Отправим данные в удаленный репозиторий**

# Добавляем удаленный репозиторий

git remote add origin https://github.com/Ragooku/rpp2025

# Отправляем изменения на сервер

git push -u origin main

**2.8. Клонирования репозитория**

# Клонируем репозиторий в другую папку

cd C:/Users/vladislavlysak/Desktop/rpp2025

git clone https://github.com/Ragooku/labrpp4 lab4-copy

**2.9. Добьемся конфликта файлов при применении изменений в оригинальном репозитории**

В оригинальной папке вносим изменения в main.py и пушим:

# Вносим изменения (например, добавляем новый метод)

git add lab4rpp.py

git commit -m "Added new method to analyze checks"

git push origin main

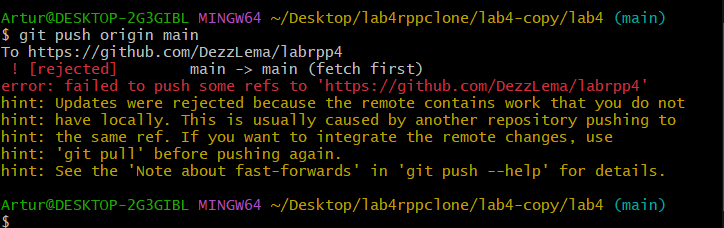
В клонированной папке также вносим изменения в тот же файл:

# Редактируем main.py

git add lab4rpp.py

git commit -m "Modified check analysis"

git push origin main # Получим ошибку, так как удаленный репозиторий изменился



**2.10. Разрешения конфликта.**

# Сначала получаем изменения с сервера

git pull origin main

# Git сообщит о конфликте, откроем файл и разрешим конфликт

# После разрешения конфликта:

git add lab4rpp.py

git commit -m "Merged changes from remote"

git push origin main

**Список выполненных команд**

git init - инициализация нового репозитория

git add - добавление файлов в индекс

Создание и настройка .gitignore

git commit -m "message" - фиксация изменений

git branch, git checkout - работа с ветками

git merge - слияние веток

git remote add, git push - работа с удаленным репозиторием

git clone - клонирование репозитория

git pull, разрешение конфликтов - работа с изменениями из удаленного репозитория

**Вывод**

В ходе выполнения лабораторной работы по Git была успешно создана локальная репозиторий, в который были добавлены несколько файлов. Один из файлов был сделан игнорируемым с помощью файла .gitignore. Все изменения были зафиксированы с использованием команды git commit. После этого были созданы несколько веток, в каждой из которых были внесены изменения и зафиксированы. Ветви были успешно слиты в одну. Далее был создан аккаунт на GitHub, и исходные коды были отправлены в удаленный репозиторий. Удаленный репозиторий был склонирован в новую папку. Для демонстрации конфликта изменений были внесены изменения в оригинальном репозитории, что привело к конфликту. Конфликт был разрешен с помощью текстового редактора nano. В результате выполнения лабораторной работы были освоены основные команды Git, а также принципы работы с удаленными репозиториями.