

Отчет по лабораторной работе №1

Организация рабочего окружения и работа с Git

Дата: 03-12-2025

Семестр: 2 курс, 1 семестр

Группа: ПИН-Б-О-24-2

Дисциплина: Технологии программирования

Студент: Осипов Александр Сергеевич

Цель работы

Приобрести практические навыки настройки среды разработки в Ubuntu, освоить базовые операции системы контроля версий Git, изучить стратегии ветвления Git Flow/GitHub Flow и настроить автоматизированный контроль качества с помощью pre-commit хуков.

Теоретическая часть

Git - распределенная система контроля версий, позволяющая отслеживать изменения в коде и организовывать командную работу. **GitHub** - веб-сервис для хостинга проектов.

Стратегии ветвления: * **Git Flow:** Сложная модель с ветками `main`, `develop`, `feature/*`, `release/*`, `hotfix/*`. * **GitHub Flow:** Упрощенная модель с веткой `main` и feature-ветками.

Pre-commit hooks: Скрипты для автоматической проверки качества кода перед КОММИТОМ.

Практическая часть

Выполненные задачи

- [x] Часть 0: Предварительная настройка (SSH)
- [x] Часть 1: Настройка Git и инициализация репозитория
- [x] Часть 2: Настройка pre-commit хуков
- [x] Часть 3: Основной workflow (GitHub Flow)
- [x] Часть 4: Работа с удаленным репозиторием
- [x] Часть 5: Имитация Git Flow
- [x] Часть 6: Финальные штрихи

Часть 1-3. Реализация скриптов

В ходе работы были созданы скрипты на языке Python.

Код программы `hello.py`

```
print("Hello, Word!")  
print ("Im here!")
```

Код программы `greeting.py`

```
имя_пользователя = input("Ваше имя пользователя: ")
```

Часть 2. Настройка pre-commit

Был создан файл конфигурации `.pre-commit-config.yaml` для автоматической проверки кода.

```
repos:  
  - repo: https://github.com/pre-commit/pre-commit-hooks  
    rev: v4.4.0  
    hooks:  
      - id: trailing-whitespace  
      - id: end-of-file-fixer  
      - id: check-yaml  
      - id: check-added-large-files  
  - repo: https://github.com/psf/black  
    rev: 23.3.0  
    hooks:
```

```
- id: black  
  language_version: python3
```

Часть 5. Имитация Git Flow

Были созданы ветки `develop` и `feature/calculator` для демонстрации работы с Git Flow.

Выводы

1. Настроено рабочее окружение и SSH-аутентификация для GitHub.
2. Изучены и применены на практике стратегии `GitHub Flow` и `Git Flow`.
3. Настроен автоматический контроль качества кода с помощью `pre-commit`.
4. Создан проект под управлением Git с историей коммитов и ветвлением.