**ЛАБОРАТОРНАЯ РАБОТА**  
**Тема:** Система контроля версий  
**Специальность:** Информационные системы и программирование  
**Курс:** 3

**Цель работы:**

Ознакомиться с системой контроля версий (СКВ), научиться настраивать и использовать Git в среде разработки PyCharm.

**ЧАСТЬ 1. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ**

**1.1. Определение системы контроля версий**

Система контроля версий (СКВ) — это инструмент для отслеживания изменений в коде, позволяющий управлять версиями файлов, сохранять историю изменений и работать в команде.

**1.2. Основные принципы работы с Git**

* **Локальный репозиторий** — хранится на компьютере разработчика.
* **Удаленный репозиторий** — хранится на сервере (например, GitHub, GitLab, Bitbucket).
* **Коммиты (commits)** — фиксируют изменения в файлах.
* **Ветки (branches)** — позволяют вести разработку независимо.
* **Слияние (merge)** — объединяет изменения из разных веток.
* **Pull и Push** — команды для получения и отправки изменений в удаленный репозиторий.

**1.3. Основные команды Git**

* git init — инициализация репозитория.
* git clone <URL> — клонирование удаленного репозитория.
* git add <file> — добавление файлов в отслеживаемые.
* git commit -m "сообщение" — фиксация изменений.
* git push — отправка изменений в удаленный репозиторий.
* git pull — получение обновлений с удаленного репозитория.
* git branch — просмотр и создание веток.
* git merge — слияние веток.

**ЧАСТЬ 2. ПРАКТИЧЕСКАЯ РАБОТА**

**2.1. Задание**

Настроить систему контроля версий Git в PyCharm, создать локальный и удаленный репозиторий, выполнить базовые операции с файлами.

**2.2. Этапы выполнения работы**

**1. Установка Git (если не установлен)**

1. Скачать и установить Git с [официального сайта](https://git-scm.com/).
2. Проверить установку с помощью команды:
3. git --version

**2. Настройка Git в PyCharm**

1. Открыть PyCharm и загрузить или создать проект.
2. Перейти в **File → Settings → Version Control → Git**.
3. Указать путь к исполняемому файлу Git (обычно C:\Program Files\Git\bin\git.exe на Windows или /usr/bin/git на Linux/Mac).
4. Нажать **Test**, убедиться, что Git работает, и нажать **OK**.

**3. Инициализация локального репозитория**

1. Открыть **VCS → Enable Version Control Integration…**.
2. Выбрать **Git** и нажать **OK**.
3. Проверить, что папка .git появилась в корне проекта.

**4. Создание и фиксация коммита**

1. Добавить новый Python-файл в проект.
2. Перейти в **VCS → Git → Add** или нажать правой кнопкой на файле и выбрать **Git → Add**.
3. Перейти в **VCS → Git → Commit**.
4. Ввести сообщение (например, "Первый коммит") и нажать **Commit**.

**5. Подключение к удаленному репозиторию**

1. Создать новый репозиторий на [GitHub](https://github.com/) или другом хостинге.
2. Скопировать URL удаленного репозитория.
3. Открыть **VCS → Git → Remotes…**.
4. Нажать **+**, вставить URL и нажать **OK**.

**6. Отправка изменений в удаленный репозиторий**

1. Перейти в **VCS → Git → Push**.
2. Выбрать удаленный репозиторий и нажать **Push**.
3. Проверить, что изменения появились на GitHub.

**7. Получение изменений из удаленного репозитория**

1. Внести изменения в файлы на GitHub.
2. В PyCharm выбрать **VCS → Git → Pull**.
3. Убедиться, что изменения загрузились в локальный проект.

**2.3. Ожидаемый результат**

* Настроен Git в PyCharm.
* Создан локальный и удаленный репозиторий.
* Выполнены базовые операции (commit, push, pull).
* Проверена синхронизация с удаленным репозиторием.