Міністерство освіти і науки України

Кременчуцький національний університет   
імені Михайла Остроградського

Навчально-науковий інститут електричної інженерії   
та інформаційних технологій

Кафедра автоматизації та інформаційних систем

НаВчальна дисципліна  
«**АЛГОРИТМИ І СТРУКТУРИ ДАНИХ**»

Звіт

З ЛАБОРАТОРНОЇ РОБОТИ №1

Виконав:

студент групи КН-24-1

Більда П. В.

Перевірив:

доцент кафедри АІС

Сидоренко В. М.

Кременчук 2025

**Тема:** Налаштування середовища розробки та системи контролю версій Git.

**Мета:** Встановити та налаштувати необхідне програмне забезпечення (Git, Python, PyCharm), створити акаунт на GitHub, навчитися працювати з віддаленими репозиторіями та освоїти базові команди Git для ефективної розробки та управління версіями проєктів.

**Хід роботи:**

1. Створення акаунту на GitHub та налаштування доступу

Зареєструвався на GitHub за посиланням https://github.com/

Створив новий репозиторій з назвою "PavloBilda" з файлом README та налаштував доступ до репозиторію через HTTPS протокол

2. Встановлення Python

Зайшов на офіційний сайт Python (python.org)

Завантажив актуальну версію Python.

Встановив інтерпретатор з параметрами за замовчуванням, включно з додаванням Python до змінної PATH

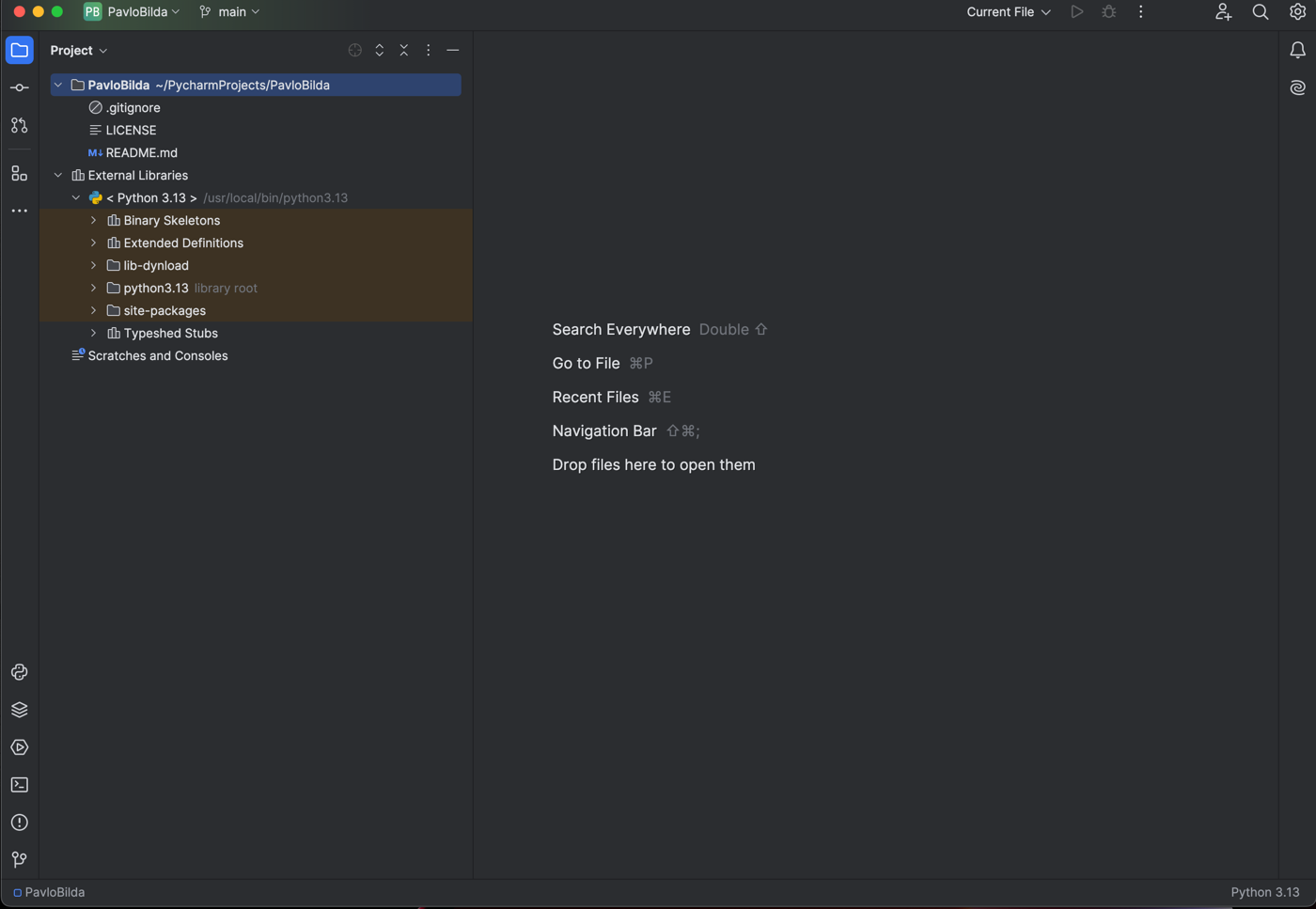
Перевірив коректність встановлення, запустивши команду python в командному рядку

3. Встановлення та налаштування IDE PyCharm

Завантажив безкоштовну Community версію PyCharm з офіційного сайту

Встановив IDE з налаштуванням змінної PATH

Запустив PyCharm і переконався у коректному запуску середовища



Інтерфейс PyCharm

4. Створення Python-проєкту та зв'язування з Git-репозиторієм

Створив новий проєкт у PyCharm з відповідними налаштуваннями:

Вибрав віртуальне середовище

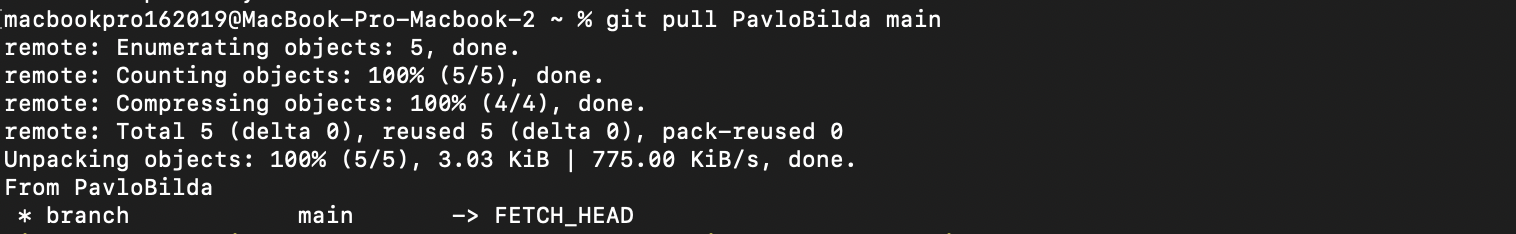
Визначив розташування проєкту

Вибрав версію інтерпретатора Python

Ініціалізував Git-репозиторій у проєкті через меню VCS → Create Git Repository

Зв'язав локальний репозиторій з віддаленим на GitHub через меню Git → Manage Remotes, вказавши URL мого репозиторію

5. Створення та комітування файлу, та команда PULL.



6. Відправлення змін на GitHub

Відправив зміни на віддалений репозиторій через меню Git → Push

Перевірив наявність змін на GitHub

**Контрольні питання**

1. Що таке IDE PyCharm? – це інтегроване середовище розробки, для

мови програмування Pythonб створене компанією JetBrains.

2. Що таке система контролю версій (СКВ) і для чого вона призначена? –

це програмне забезпечення яке дозволяє зберігати історію змін у коді або

будь-яких інших файлах, відстежувати, хто і коли зробив зміни, а також

повертатися до попередніх версій у разі потреби.

У чому полягає принципова відмінність СКВ SVN і GIT? –

СКВ – це інструмент для збереження історії змін файлах відстеження

правок спільної роботи над проектом та можливості повернення до

попередніх версій.

SVN – централізована система контролю версій де всі зміни

зберігаються на центральному сервері. Користувачі підключаються до

сервера для отримання або відправлення змін. Робота з гілками повільна

та менш зручна.

GIT – це розподілена система контролю версій, де кожен користувач має

локальну копію репозиторію з усією історією. Дозволяє працювати

offline, швидко створювати та зливати гілки. Операції виконуються

локально, що забезпечує високу швидкість і гнучкість.

3. Що відбувається унаслідок виконання команди git add .? – Додає зміни

у файлах до індексу (staging area), готуючи їх до коміту.

4. Що відбувається унаслідок виконання команди git commit? – Ця

команда зберігає всі додані в індекс зміни у вигляді окремої

версії (коміту) в локальному репозиторії.

5. Якою Git-командою виконується розміщення локальної копії Git-

репозиторію проєкта на віддаленому, що знаходиться на GitHub?-

Команда (git push -u origin main) де origin – це ім’я віддаленого

репозиторію, а main – гілка, в яку ми хочемо відправити якісь зміни.

**Висновки:**

В результаті виконання лабораторної роботи я:

Cтворив та налаштував аккаунт на GitHub, та створив віддалені

репозиторії для зберігання коду.

Встановив та налаштував “Python” та ще одне середовище розробки

“PyCharm” для ефективної роботи над проєктами.

Розібрався з командами у Git для контролю версій: ініціалізацію

репозиторію, додавання файлів, комітування та синхронізацію з

віддаленим репозиторієм.

Розібрався з основами середовища розробки PyCharm.

Отримані знання стануть в основу для подальшої праці, навчання і

створення більш цікавих та великих проектів.