**Лабораторная работа №6**

**РАЗРАБОТКА ПРИЛОЖЕНИЯ РАБОТЫ С БАЗОЙ ДАННЫХ**

**Вариант 9.**

**Цель работы:** изучить возможности взаимодействия Python с реляционными базами данных с помощью DB-API 2.0.

Краткая теория

Почти каждая программа, которая работает с данными, требует места для их хранения. В качестве такого хранилища могут выступать обычные текстовые файлы, файлы JSON или XML. Однако для хранения большого количества данных и удобной организации работы с ними существуют СУБД. В данной лабораторной работе не рассматриваются понятия и термины баз данных, синтаксис SQL. Здесь будут рассмотрены механизмы для работы с базами данных с помощью Python.

DB-API 2.0

Несмотря на существующий стандарт SQL (ISO/IEC 9075) каждая СУБД имеет ряд отличий от других. Для того, чтобы программисту не вникать в их реализации в Python существует общее API, описанное стандартом PEP 249. Стоит учесть, что PEP 249 это всего лишь спецификация, реализация которой выполняется самостоятельно. Однако существуют уже готовые реализации данного API для ряда СУБД. Стоит отметить, что Python имеет в качестве встроенного модуля одну из реализаций данного API для работы с SQLite. Ниже приведены некоторые реализации для других СУБД:

− Oracle: cx\_Oracle, mxODBC, pyodbc;

− MySQL: mysql-python, PyMySQL;

− MS SQL Server: adodbapi, pymssql, mxODBC, pyodbc;

− PostgreSQL: psycopg2, txpostgres.

**Задание. Вариант 9.**

Написать функцию, которая принимает путь к файлу с расширением .txt, который содержит данные для заполнения таблицы, и по указанным в файле данным создает таблицу или таблицы и заполняет их данными.

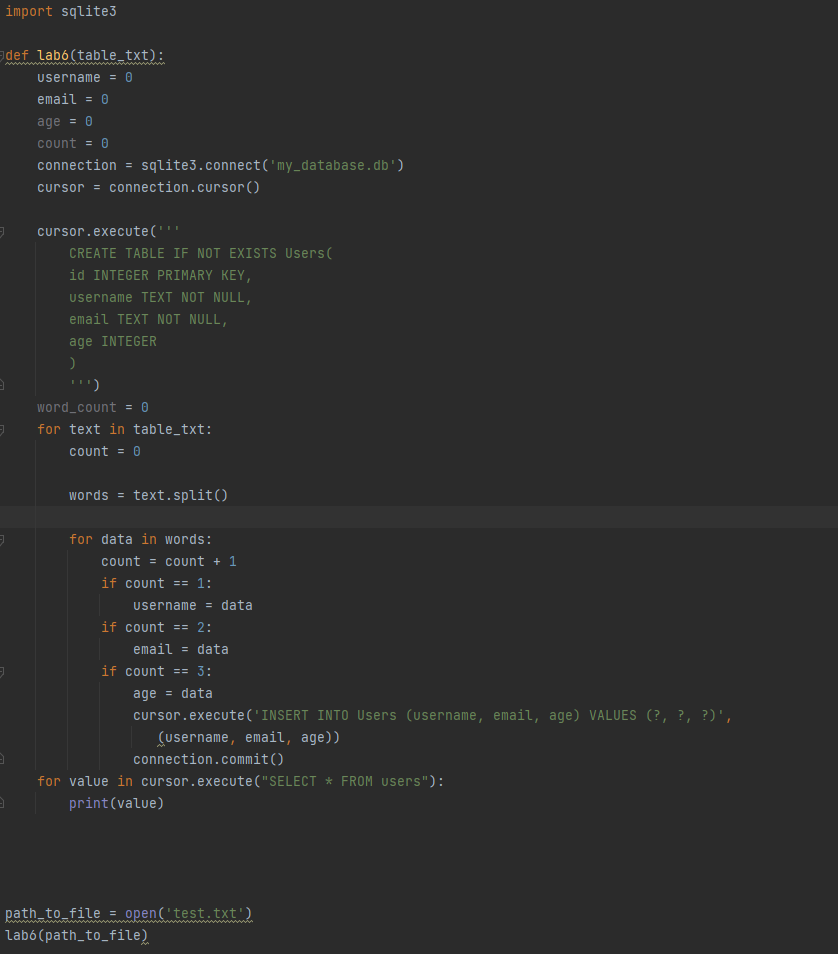


Рисунок 6.1 - код программы

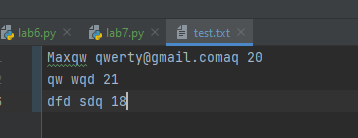
****

Рисунок 6.2 - текстовый файл

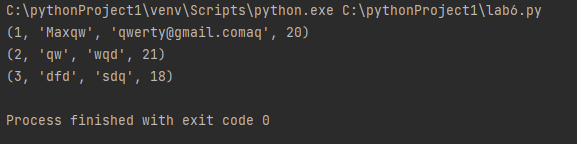


Рисунок 6.3 - результат работы

**Вывод:** в ходе выполнения лабораторной работы были изучены возможности взаимодействия Python с реляционными базами данных с помощью DB-API 2.0.

|  |  |
| --- | --- |
| Выполнил | Нестеренко М.С. |
| Проверил | Елкин Н.С. |