**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ПЕРВОЕ ВЫСШЕЕ ТЕХНИЧЕСКОЕ УЧЕБНОЕ ЗАВЕДЕНИЕ РОССИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«САНКТ-ПЕТЕРБУРГСКИЙ ГОРНЫЙ УНИВЕРСИТЕТ ИМЕНИ ИМПЕРАТРИЦЫ ЕКАТЕРИНЫ II»**

**Кафедра автоматизации технологических процессов и производств**

Отчет по дисциплине «Программирование и алгоритмизация»

Практическое задание №2

«Взаимодействие с базой данных»

Выполнил: студент гр. АПН-22 \_\_\_\_\_\_\_\_\_ /Скрябнев А.В./

(подпись) (Ф.И.О.)

Проверил: доцент \_\_\_\_\_\_\_\_\_ /Фёдорова Э. Р./

(подпись) (Ф.И.О.)

Санкт-Петербург

2024

Цель работы: Изучение основ баз данных, а также способов взаимодействия с ними.

1. Основные команды языка SQL
2. Создание новой базы данных: «CREATE DATABASE [name];», где вместо [name] вписываем имя новой базы данных. Для проверки что команда выполнилась можно написать: «SHOW DATABASES;». Для дальнейшего использования созданной базы данных, используется команда: «USE [name]».
3. Реляционная база данных (РБД) – это набор данных с предопределенными связями между ними. Эти данные организованны в виде набора таблиц, состоящих из столбцов и строк. В таблицах хранится информация об объектах, представленных в базе данных. Для создания таблицы используется команда (Рисунок 23):

«CREATE TABLE [name] (

[nameColumn1] [typeOfValue] [flags]»

[nameColumn2] [typeOfValue] [flags]»

…);

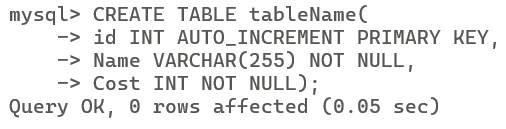


Рисунок 23 – Пример использования команды

typeOfValue – тип переменной. Пример: INT, VARCHAR(255), …

flags:

NOT NULL – Автоматическое заполнение значение not null;

PRIMARY KEY - это поле, которое однозначно идентифицирует каждую запись в таблице.

AUTO\_INCREMENT - это атрибут, который автоматически генерирует уникальные целочисленные значения для столбца при вставке новой строки в таблицу.

Внесение значений в таблицу используется команда INSERT (Рисунок 24).

«INSER [nameTable] ([nameValue1, nameValue2…]) VALUES («Value1», «Value2»…), …»



Рисунок 24 – Пример использования команды.

Получение значений с таблицы осуществляется командой SELECT:

SELECT [nameColumn1], [nameColumn2], … FROM [nameTable];

Вместо перечисления названия столбцов можно напечатать символ «\*», для получения всех столбцов (Рисунок 25).

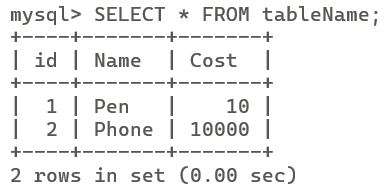


Рисунок 25 – Пример использования команды

1. Использование коннекторов (библиотек) для взаимодействия с ними через языки программирования (Python).

Данные об дифференциальном уравнение мы храним в табличном виде для удобного переноса информации в базу данных. Для это используется библиотека «pandas».

Для хранения данных в табличном виде используется метод «DataFrame». На рисунке 26 показано хранение коэффициентов уравнения и массив значений «x», «y».

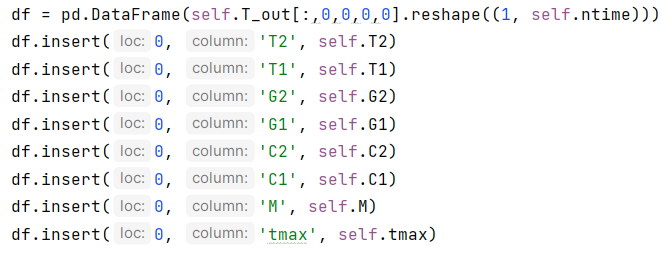


Рисунок 26 – пример применения метода «DataFrame» библиотеки «Pandas»

Для переноса таблицы в базу данный используется библиотека «SQLAlchemy». Данная библиотека работает по структуре, изображенной на рисунке 27.

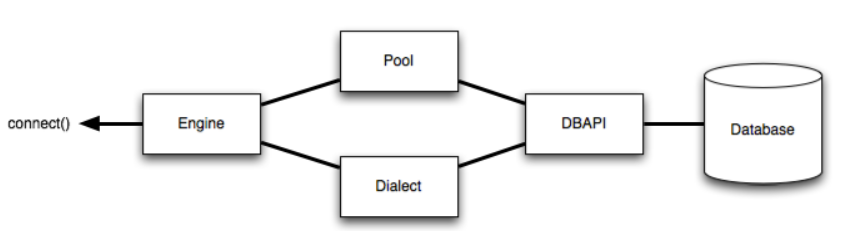


Рисунок 27 – Структура взаимодействия базы данных с языком Python

Engine является отправной точкой для любого приложения «SQLAlchemy» [3]. Представляет из себя URL, необходимое для хранения и дальнейшей передачи параметров подключения и взаимодействия с базой данной в DBAPI. В URL указывается название базы данных и название API (DataBaseAPI), к которой мы будем подключаться (Рисунок 28).



Рисунок 28 – Пример URL

К параметрам, которые необходимы для подключения относятся:

* Пользователь (Регистрируется при скачивании, вместе с паролем)
* Пароль
* Host - Адрес или же порт сервера, на котором хранится Database. Если программа, при помощи которой происходит подключение к базе данных и сама база данных хранится на одном и том же сервере (компьютере), то для этого существует стандарт. Он разнится от использования API. В нашем случае: «localhost:3306».
* Название базы данных.

Для определения необходимого API используются модули «Pool» и «Dialect». Они интерпретируют функции модуля DBAPI, а также поведение базы данных [3].

«Engine» является одним из параметров метода «to\_sql» библиотеки «pandas». Этот метод необходим для переноса целой таблицы в базу данных (Рисунок 29).

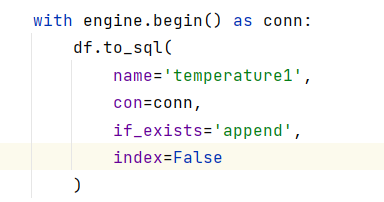


Рисунок 29 – Использование метода «to\_sql»

Он создает таблицу в базе данных аналогичную переменной «df» под название «temperature1». Параметр «if\_exists» позволяет не создавать при повторном обращении уже существующую таблицу, а добавление в нее новых значений.

Метод «with … as» автоматически разрывает соединение с базой данных, для предотвращения не нужных ошибок.

Вывод: Научились пользоваться базами данных, а также способы взаимодействия с ними через язык программирования Python.