

Урок 4. Работа с базой данных

Схематично работа с базой данных может выглядеть примерно так:

1. Осуществить подключение к базе данных (вызов `connect()` с получением объекта-соединения)
2. Создание одного или нескольких курсоров (вызов метода объекта-соединения `cursor()` с получением объекта-курсора)
3. Исполнение команды или запроса (вызов метода `execute()` или его вариантов)
4. Получение результатов запроса (вызов метода `fetchone()` или его вариантов)
5. Завершение транзакции или ее откат (вызов метода объекта-соединения `commit()` или `rollback()`)
6. Когда все необходимые транзакции произведены, подключение закрывается вызовом метода `close()` объекта-соединения

Задание 1. Подключение к MS Excel с помощью библиотеки **pandas** и функции `read_excel()`.

1. Проанализировать код

```
import pandas
excelDataDF = pandas.read_excel('flowers.xlsx', sheet_name='winter')
# напечатать всё содержимое листа
print(excelDataDF)
```

2. Создать файл MS Excel «flowers». На листе «winter» создать таблицу с полями: номер п/п, название цветка, цена, кто любит цветок, количество, стоимость (цена * на количество).
3. Заполнить таблицу произвольными значениями (не менее 10 строк)
4. Вывести информацию на экран консоли

Задание 2. Подключение к БД SQLite с помощью библиотеки **sqlite3**.

1. Проанализировать и реализовать код

```
import sqlite3
# Попытаться открыть соединение к базе
try:
    # осуществить подключение к БД sqlitePy
    connection = sqlite3.connect('sqlitePy.db')

    # создать курсор для выполнения запросов
    cursor = connection.cursor()
    print("База данных создана и успешно подключена к SQLite")

    # вывести версию базы
    select_query = "select sqlite_version();"
    cursor.execute(select_query)

    # вернуть все строки в виде списка
    record = cursor.fetchall()
    print("Версия базы данных SQLite: ", record)
    # закрыть курсор
    cursor.close()

except sqlite3.Error as error:
    print("Ошибка при подключении к sqlite", error)
finally:
    # закрыть соединение с базой
    if (connection):
        connection.commit()
        connection.close()
        print("Соединение с SQLite закрыто")
```

2. Создать свою базу **city.db** с таблицей city и полями: номер п/п (счетчик), название города.
3. Используя методы execute(), executemany() и запрос INSERT INTO, заполнить таблицу произвольными значениями (не менее 10 строк)
4. Применить методы fetchone(), fetchmany(), fetchall() вывести значения из базы.

Задание 3. Самостоятельно создайте БД publ.db SQLite с таблицами, установите взаимосвязи между таблицами, заполните данными и отобразите.

1. Таблицы Пользователи (tUsers), Публикации (tPubl), Комментарии (tComments).
2. Таблица Пользователи содержит поля: id, имя, возраст, пол;
Таблица Публикации содержит поля: id, заголовок, описание, id_человека;
Таблица Комментарии содержит поля: id, текстКомментария, id_публикации, id_человека
3. Установить отношения один ко многим между таблицами:
Один человек может создавать несколько публикаций. Один человек может оставлять много комментариев, а одна публикация может иметь несколько комментариев (схема БД представлена на Рисунок 1).
4. Заполните таблицы произвольными данными (не менее 10 строк)
5. Запустите выборки данных

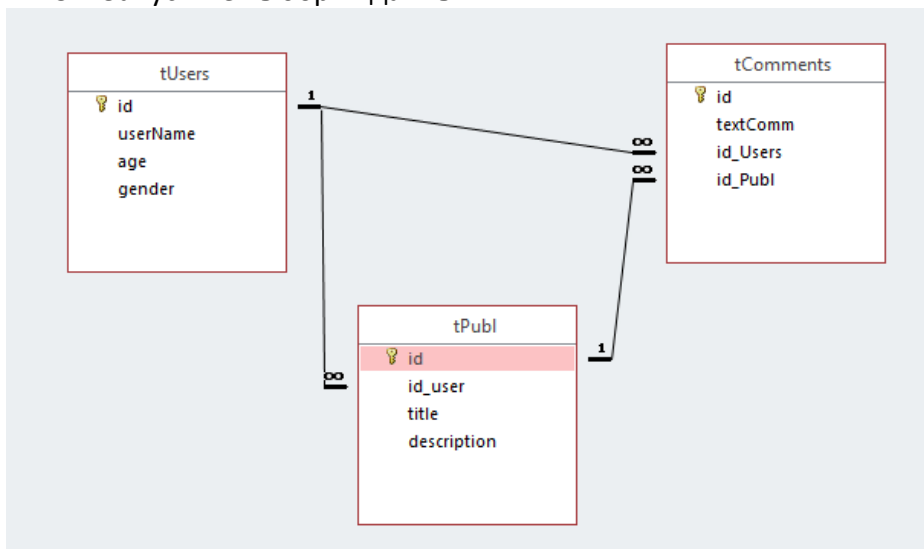


Рисунок 1. Схема БД publ.db