

# Работа с базами данных при помощи языка программирования Python и СУБД SQLite3 в школьном курсе информатики

Пономарев Д.А.

*den.ponomarev2011@mail.ru*

*Саратовский государственный университет имени Н.Г. Чернышевского*

В данной статье рассмотрены примеры работы с базами данных при помощи языка программирования Python и СУБД SQLite3. Проведен анализ ситуации с изучением баз данных и введением нового учебного языка программирования. Приведенные автором примеры показывают спектр возможностей языка программирования Python, его многофункциональность, возможность работы с системами управления базами данных.

**Ключевые слова:** базы данных, SQL, SQLite3, преподавание информатики в школе, язык программирования Python.

## **Введение.**

В ходе бурных рассуждений о новом образовательном стандарте и его наполнении, возникают споры и о содержании того или иного учебного предмета. В ходе общественного обсуждения, предложенного учеными из университета ИТМО, была утверждена Концепция учебного предмета Информатика. Автора статьи в концепции заинтересовали две вещи: переход на новый учебный язык программирования и переход к языку запросов SQL для изучения баз данных.

### **1. Проблема изучения программирования и основ баз данных.**

С 2013 года в мировой IT-индустрии наблюдается крупномасштабное распространение и внедрение технологий, предназначенных для анализа данных больших объемов. В настоящее время большинство крупных IT-компаний разрабатывают решения, связанные с обработкой Больших Данных. Социальные сети, различные сервисы и даже муниципальные организации используют различные программы, которые оперируют огромными потоками информации. И эти объемы постоянно увеличиваются, создавая проблемы их управления. Поэтому уже сегодня без баз данных невозможно представить работу большинства компаний и учреждений. Впервые термин «Большие данные» появился в прессе в 2008 году, когда редактор журнала Nature Клиффорд Линч выпустил статью на тему развития будущего науки с помощью технологий работы с большим количеством данных. До 2009 года данный термин рассматривался только с точки зрения научного анализа, но после выхода еще нескольких статей пресса стала широко использовать понятие Big Data – и продолжает использовать его в настоящее время. Необходимо пояснить, что термины «Большие данные» и «Big Data» являются тождественными. Поскольку термин появился за рубежом в англоязычном журнале, то использование английского термина не является ошибкой.

С каждым годом повышается спрос на специалистов по работе с большими данными – Big Data Engineer. Согласно открытым данным компании HeadHunter в 2018 г. количество вакансий с упоминанием специальности Data Scientist (специалист по работе с данными) выросло в 7 раз по сравнению с 2015 г., а вакансий с ключевыми словами Machine Learning – в 5 раз. По мнению

аналитиков IDC, Россия является крупнейшим региональным рынком решений в Центральной Восточной Европе по аналитике Big Data.[1] И хотя ВУЗы начали проводить изучение больших данных в качестве отдельного предмета уже в 2013 году, спрос на специалистов в данной области огромен.

Подводя итог, можно сделать вывод, что школьники изучающие основы баз данных, умеющие работать с базами данных, имеют хорошие шансы поступить на специальность Data Scientist, и в последующем устроиться на престижную высокооплачиваемую работу. Однако, в настоящее время эта тема изучается при помощи устаревшей технологии Microsoft Access, которая не дает полного и современного понимания построения и работы баз данных.

Программирование, и в целом навыки работы с компьютером, становятся наиболее важными в современном мире. Поэтому ученики, изучающие программирование, овладевают современной и развивающейся профессией, которая приносит значительный доход, но и развивают логическое мышление, точность формулировок, аккуратность, работу с данными и многое другое. Но спрос опережает предложение. По оценкам аналитиков, чтобы удовлетворить потребность рынка в ИТ-кадрах, надо как минимум в 2,5 раза увеличить набор и выпуск студентов соответствующих специальностей. Этому вторят и цели национальных проектов: к 2024 году вузы должны предложить рынку 120 тыс. выпускников в сфере ИТ, а доля населения, обладающего цифровыми навыками, должна достигнуть 40%. Для достижения этих целей будет реализовано направление «Кадры для цифровой экономики».[2] Что же касается системы школьного образования, то основным языком обучения в школах до сих пор является язык программирования Pascal, который появился в 70 годах прошлого века. Этот язык обучения уже морально устарел и не может достоверно отражать изменения в сфере ИТ. И сейчас проходят полноценные дискуссии по выбору нового учебного языка программирования. Здесь можно отметить язык программирования Python, который в последнее время набирает все большую популярность, как среди профессиональных программистов, так и среди студентов и школьников. В последние годы он является языком программирования с самым быстрым темпом роста количества пользователей. [3] Автор данной статьи уже писал, что в данный момент времени наиболее благоприятным, как для системы образования, так и для детей, стоит считать введение Python как второго языка.[4] Изучение языка программирования Python должно начинаться в старших классах профильного обучения. Это позволит подготовить преподавательскую базу, учебники и учебные пособия, а также нивелирует неизбежные проблемы.

Далее в статье будут разобраны основные примеры работы с базами данных при помощи языка программирования Python. Необходимо пояснить, что приведенные далее примеры работы – это только так называемый «скелет». Каждый из этих примеров можно расширить и дополнить, в зависимости от конкретных задач, стоящих перед учителем.

## **2. Практические примеры.**

Перед практическими примерами стоит объяснить выбор СУБД SQLite 3 для связки с языком программирования Python. Во-первых, это одна из самых

популярных СУБД в мире. Главными ее конкурентами являются СУБД MySQL и PostgreSQL. Далее разберем основные минусы и плюсы представленных СУБД.

MySQL – это самая популярная на данный момент из всех крупных серверных баз данных. Это достаточно простая в использовании СУБД, к тому же в сети есть огромное количество обучающей информации. MySQL не поддерживает полностью стандарты SQL, но она предлагает достаточно широкий функционал. Неполная SQL-стандартизированность – это главный минус данной СУБД, и существенный повод отказаться от нее в учебном процессе. Такая система управления подойдет для уже опытных пользователей, знакомых со всеми тонкостями стандартов SQL.

PostgreSQL – это СУБД, обладающая широким функционалом, ориентирующаяся на полное соответствие стандартам SQL и расширяемость системы. PostgreSQL – одна из самых передовых систем баз данных с открытым исходным кодом, в ней много возможностей, широко используемых как разработчиками, так и системными администраторами. Это очень хорошая, мощная СУБД, но функционал PostgreSQL не будет использован полностью в школьном курсе. Ее использование можно будет сравнить с пальбой из пушки по воробьям.

SQLite 3 обладает следующими качествами, которые выгодно выделяют его на фоне других СУБД:

- SQLite свободно распространяемая СУБД;
- SQLite стандартизированная СУБД;
- SQLite файловая СУБД;
- СУБД проста в установке и использовании.

Все эти пункты делают SQLite 3 приоритетным выбором в качестве учебной системы управления базами данных.

Первый пример, который будет разобран – это создание и вставка таблиц. Необходимо напомнить, что язык программирования Python имеет встроенную поддержку SQLite базы данных, для этого вам не надо ничего дополнительно устанавливать, достаточно в скрипте указать импорт стандартной библиотеки.

Пример:

```
# импорт модуля sqlite3
import sqlite3
# подключение к созданной базе данных 'WORK.db'
connection = sqlite3.connect('WORK.db')
# подключение курсора
cursor = connection.cursor()
# создание таблицы workers при помощи команды SQL CREATE TABLE
sql_command = CREATE TABLE workers(
#задаются параметры столбцов таблицы. В данном случае задается
столбец 'number' хранящий целочисленные значения и являющийся
первичным ключом.
number INTEGER PRIMARY KEY,
#Задается столбец 'fname', содержащий текстовые значения.
fname TEXT,
#Задается столбец 'lname', содержащий текстовые значения.
```

```

lname TEXT,
#Задается столбец 'gender', содержащий текстовые значения.
gender TEXT,
);
#выполнение инструкции
cursor.execute(sql_command)
#вставка записи в таблицу при помощи SQL команды INSERT INTO
sql_command = INSERT INTO workers VALUES (35, 'Andrey',
'Elagin', M');
# выполнение инструкции
cursor.execute(sql_command)
# вставка записи в таблицу при помощи SQL команды INSERT INTO
sql_command = INSERT INTO workers VALUES (11, 'Alexander',
'Kulagin', 'M');
# выполнение инструкции
cursor.execute(sql_command)
#сохранение изменений в таблице
connection.commit()
#закрытие соединения
connection.close()

```

В данном примере, создается таблица, задаются параметры ее столбцов и в таблицу заносят две записи. Главное, не забыть сохранить результат операций, поскольку автоматического сохранения нет. Учитель может расширить данный пример путем добавления столбцов, содержащих данные других типов или создать несколько таблиц, объединив их одной из существующих связей.

Далее рассмотрим извлечение данных из созданной таблицы. Это одно из самых часто проводимых действий с базами данных. Извлечение будет проводиться в переменную 'ans', и затем выводиться при помощи цикла.

**Пример:**

```

#импорт модуля sqlite3
import sqlite3
#подключение к созданной базе данных 'WORK.db'
connection = sqlite3.connect('WORK.db')
cursor = connection.cursor()
#выполнение извлечения всех данных из таблицы при помощи SQL
команды SELECT.
cursor.execute('SELECT * FROM workers')
#сохранение данных в переменную ans.
ans = cursor.fetchall()
#вывод данных таблицы при помощи цикла for.
for i in ans:
print(i)

```

Учитель может расширить данный пример, извлекая данные из конкретного столбца или данные, подходящие по заданным параметрам. Для этого используется SQL конструкция SELECT WHERE.

Затем рассмотрим обновление записей в таблице и их удаление.

**Пример:**

```

#импорт модуля sqlite3
import sqlite3
#подключение к созданной базе данных 'WORK.db'

```

```

conn = sqlite3.connect('WORK.db')
#обновление записи таблицы, удовлетворяющей заданному условию.
connection.execute('UPDATE workers lname = 'Elagin' WHERE
number= '4''')
#сохранение изменений в таблице.
connection.commit()
#вывод количества измененных строк.
print 'Общее количество обновленных строк:', connection.
total_change'
#удаление записей таблицы, удовлетворяющих заданному условию.
cursor.execute('DELETE FROM workers WHERE fname = 'Oleg''')
#сохранение изменений в таблице.
connection.commit()
#закрытие соединения.
connection.close()

```

Расширить и усложнить данное задание можно следующим образом. Для обновления записи таблицы можно задать несколько условий, используя логические операторы OR, AND. Удаление данных из таблицы также можно усложнить при помощи логических операторов.

### **Вывод**

В данной статье рассмотрены операции с базами данных, проводимые при помощи языка программирования Python. Были рассмотрены: создание таблицы, создание столбцов таблицы с заданными параметрами, рассмотрено заполнение таблицы, изменение и удаление записей таблицы. Эти операции являются базовыми для работы с базами данных, это тот необходимый минимум знаний, который должен быть изучен учащимся. Использование в данном случае языка программирования Python позволяет вывести практическое применение полученных знаний на новый уровень. Ученик может в качестве индивидуального проекта или подготовки к научной конференции создать приложение, веб-сайт или, как сейчас стало очень модно, telegram-бота. Все эти направления работают с данными, соответственно, их нужно где-то хранить, обрабатывать и изменять. И полученные на таких уроках знания позволят ученикам уже при поступлении в ВУЗы иметь более высокий уровень подготовки.

### **Список литературы**

- [1] Официальный сайт компании Daily Comm «Коммуникации в IT-бизнесе» [Электронный ресурс] / URL: <http://www.dailycomm.ru/m/44819/>
- [2] Официальный сайт национальной программы «Цифровая экономика» [Электронный ресурс] URL: <https://data-economy.ru/2024>
- [3] Ежегодный интерактивный рейтинг лучших языков программирования IEEE Spectrum [Электронный ресурс] URL: <https://jobs.ieee.org/jobs/content/The-Top-Programming-Languages-2019-2019-09-09>
- [4] Пономарев Д.А. Варианты изучения языка программирования Python в школьном курсе информатики // Образование. Технологии. Качество.: Материалы Всеросс. Научно-практ. конф. – М: Издательство «Перо», 2019. – 130-136 с.