Сообщить об ошибке.

РЕКЛАМА .

ХОЧУ ПОМОЧЬ ПРОЕКТУ

Драйвер для баз данных SQLite3

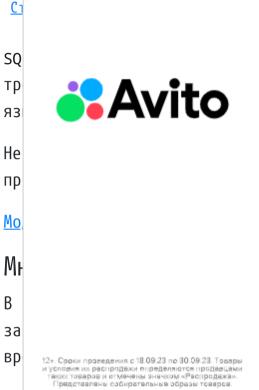


mango-office.ru

Виртуальная АТС Расширенная

2000₽

Узнать больше



/ Драйвер для баз данных SQLite3

С, которая предоставляет легковесную дисковую базу данных. База данных SQLite не процесса и позволяет получить доступ к базе данных, используя нестандартный вариант

пользовать SQLite для внутреннего хранения данных. Также возможно создать прототип QLite, а затем перенести код в большую базу данных, такую как PostgreSQL или Oracle. нтерфейс SQL, совместимый со спецификацией DB-API 2.0, описанной в PEP 249.

кали проблемы с разделением соединений между потоками. По этому <u>модуль sqlite3</u> Python

я и курсоры между потоками. Если попытаться это сделать, то получим исключение во

Единственным методом, имеющим смысл вызывать только из другого потока является вызов метода connect.interrupt().

Примеры использования:

Чтобы использовать модуль, необходимо сначала создать объект <u>Connection</u>, который представляет базу данных. В примерах, данные будут храниться в файле example.db:

```
import sqlite3
conn = sqlite3.connect('example.db')
```

Можно также указать специальное имя :memory: для создания базы данных в оперативной памяти.

Получив coeдинение <u>Connection</u>, можно создать <u>объект Cursor</u> и вызвать его метод <u>cursor.execute()</u> для выполнения команд SQL:

Данные, которые были сохранены, являются постоянными и доступны в следующих сеансах:

```
import sqlite3
connect('example.db')
conn.cursor()
```

Обычно операции SQL могут использовать значения из переменных в Python. Нет необходимости собирать запрос, используя <u>строковые операции</u> Python, потому что это небезопасно. Это делает программу уязвимой для атаки SQL-инъекцией.

Вмєреклама гого используйте подстановку параметров DB-API. Поставьте символ '?' в качестве заполнителя везде, где вы хотите использовать значение переменной, а затем предоставьте кортеж значений в качестве второго аргумента метода курсора cursor.execute(). Другие модули базы данных могут использовать другой заполнитель, такой как '%s' или ':1'.

Например:

Чтобы получить данные после выполнения оператора SELECT, можно либо обработать курсор как итератор, вызвать метод курсора <u>cursor.fetchone()</u>, чтобы получить единственную совпадающую строку, либо вызвать <u>cursor.fetchall()</u>, чтобы получить <u>список</u> совпадающих строк.

В этом примере используется форма <u>итератора</u>:

```
>>> for row in cursor.execute('SELECT * FROM stocks ORDER BY price'):
... print(row)
...
# ('2020-01-05', 'BUY', 'RHAT', 100, 35.14)
# ('2020-03-28', 'BUY', 'IBM', 1000, 45.0)
# ('2020-04-06', 'SELL', 'IBM', 500, 53.0)
# ('2020-04-05', 'BUY', 'MSFT', 1000, 72.0)
```

Содержание раздела:

- КРАТКИЙ ОБЗОР МАТЕРИАЛА.
- <u>Типы SQLite3 и Python</u>
- <u>Хранение типов Python в базах данных SQLite3</u>
- <u>Пример преобразования типов между SQLite и Python</u>
- <u>Встроенные адаптеры и конвертеры модуля sqlite3</u>
- <u>Управление транзакциями в модуле sqlite3 Python</u>
- <u>Эффективное использование sqlite3</u>
- Функция connect() модуля sqlite3
- <u>Функция register converter() модуля sqlite3</u>
- Функция register adapter() модуля sqlite3
- <u>Методы объекта Connection модуля sqlite3</u>
- Методы объекта Cursor модуля sqlite3
- Объект Row модуля sqlite3
- <u>Сравнение кириллицы в SQLite без учета регистра</u>
- <u>Функции и константы модуля sqlite3</u>
- <u>Исключения модуля sqlite3</u>
- Объект Blob() модуля sqlite3
- Вверх 'SV и XLSX(XLS) файла в sqlite3, экспорт данных в CSV

<u>DOCS-Python.ru</u>™, 2023 г.

(Внимание! При копировании материала ссылка на источник обязательна)

@docs_python_ru

РЕКЛАМА

Вверх