

Работа с SQLite

Каждый miniApp может независимо от других хранить данные для своей работы в SQLite. Чтобы предотвратить потерю данных из БД при обновлении miniApp, предусмотрен механизм отдельного хранения файла базы за пределами хранилища miniApp. Предполагается вариант реализации API, сходный к интерфейсу node.js.

mapbox/node-sqlite3

Database Database#close([callback]) Database#configure(option, value) Database#run(sql, [param, ...], [callback]) Database#get(sql, [param, ...], [callback]) Database#all(sql, [param, ...], [callback])

<https://github.com/mapbox/node-sqlite3/wiki/API>

mapbox/**node-sqlite3**

Asynchronous, non-blocking SQLite3 bindings for Node.js

71 Contributors 226k Used by 5k Stars 665 Forks

Querying Data in SQLite Database from Node.js Applications

Summary: in this tutorial, you will learn how to query data from the SQLite database from a Node.js application using sqlite3 API. To query data in SQLite database from a Node.js application, you use these steps: Open a database connection. Execute a SELECT statement and process the result set.

<https://www.sqlitetutorial.net/sqlite-nodejs/query/>

То есть со стороны miniApp формируется json-запрос формата:

Команда

```
SQLITE

json
{
  "action" : "run", // SQL команда: run, get, all, each и т.д.
```

```
"sql": "", // строка SQL запроса  
}
```

Базовый API

Реализованы базовые методы:

```
`Get` / `Run` / `All`
```

Примеры тестовых запросов

```
`RUN`: CREATE TABLE Contact (Id INT PRIMARY KEY NOT NULL, Name CHAR(255));  
  
`RUN`: INSERT INTO Contact (Id, Name) VALUES (0, 'Alex');  
  
`GET` / `ALL`: SELECT * FROM Contact;  
  
`RUN`: UPDATE Contact SET Name = 'Adam' WHERE Id = 1;
```