



JÖNKÖPING UNIVERSITY

School of Engineering

USING SQLITE IN NODE.JS

Peter Larsson-Green

Jönköping University

Autumn 2018

THE `sqlite3` PACKAGE

Distributions of SQLite exists as npm packages, e.g.:

- `npm install sqlite3`

```
const sqlite3 = require('sqlite3')
```

```
const db = new sqlite3.Database('path/to/database.db')
```

USING THE DATABASE OBJECT

Creating a table:

```
db.run("CREATE TABLE...", function(error) {  
  if(error) {  
    // Query could not be executed.  
  } else {  
    // Query successfully executed.  
  }  
})
```

USING THE DATABASE OBJECT

Inserting rows:

```
db.run("INSERT INTO...", function(error) {  
  if(error) {  
    // Query could not be executed.  
  } else {  
    // Query successfully executed.  
    const id = this.lastID  
  }  
})
```

USING THE DATABASE OBJECT

Inserting rows:

```
const query = "INSERT INTO Humans (Name) VALUES ('Alice')"  
db.run(query, function(error) {  
  // ...  
})
```

```
const name = "Alice"  
const query = "INSERT INTO Humans (Name) VALUES ('"+name+"')"  
db.run(query, function(error) {  
  // ...  
})
```

USING THE DATABASE OBJECT

Inserting rows:

```
const name = "Alice"  
const query = "INSERT INTO Humans (Name) VALUES (?)"  
db.run(query, [name], function(error) {  
  // ...  
})
```

USING THE DATABASE OBJECT

Retrieving a single row:

```
const name = "Alice"
const query = "SELECT * FROM Humans WHERE Name = ?"
db.get(query, [name], function(error, human) {
  if(error) {
    // 😞
  } else {
    // human = {Id: 1, Name: "Alice", Age: 10}
  }
})
```

Id	Name	Age
1	Alice	10
2	Bob	15
3	Chloe Clair	20

Table: Humans

Database: Test

USING THE DATABASE OBJECT

Retrieving multiple rows:

```
const maxAge = 18
const query = "SELECT * FROM Humans WHERE Age < ?"
db.all(query, [maxAge], function(error, humans) {
  /*
    humans = [
      {Id: 1, Name: "Alice", Age: 10},
      ...
    ]
  */
})
```

Id	Name	Age
1	Alice	10
2	Bob	15
3	Chloe Clair	20

Table: Humans

Database: Test

USING THE DATABASE OBJECT

Updating rows:

```
const id = 1
const newName = "Alicia"
const query = "UPDATE Humans SET Name = ? WHERE id = ?"
db.run(query, [newName, id], function(error) {
  if(error) {
    // ...
  } else {
    const numberOfUpdatedRows = this.changes
  }
})
```

USING THE DATABASE OBJECT

Deleting rows:

```
const id = 1
const query = "DELETE FROM Humans WHERE id = ?"
db.run(query, [id], function(error) {
  if(error) {
    // ...
  } else {
    const numberOfDeletedRows = this.changes
  }
})
```