



17. MySQLの接続

mysql2 モジュールの基本

`mysql2` モジュールで、DB接続やクエリーの実行など **MySQL** の基本的な操作できます。

DB接続設定

`createConnection()` 接続情報を設定して DB接続します。接続情報は「host」「user」「port」「password」「database」などを設定します。

```
mysql.createConnection({  
  host: ホスト名,  
  database: データベース名,  
  user: ユーザ名,  
  port: ポート,  
  password: パスワード,  
});
```

DB接続

`connect()` でDB接続すると、コールバック関数が実行されます。

```
con.connect((err) => { });
```

クエリー

`query()` で **SQL** を実行します。第2引数にデータを受け渡すこともできます。

```
con.query(SQL, データ);
```

DB終了処理

DB処理をしたら最後に終了処理をします。 終了処理をしておかないとプログラムが完了しないので注意しましょう。

```
con.end();
```

DB 作成

phpMyAdmin や MySQL コマンドなどで my_page データベースを作成します。

MySQLコマンドの場合

```
% create database my_page;
```

ユーザテーブル作成

以下の SQL で users と items テーブルを作成します。

```
CREATE TABLE IF NOT EXISTS users (  
  id bigint UNSIGNED AUTO_INCREMENT PRIMARY KEY,  
  name varchar(255) NOT NULL,  
  email varchar(255) UNIQUE NOT NULL,  
  password varchar(255) NOT NULL,  
  hobby text NULL,  
  created_at DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,  
  updated_at DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP  
);  
  
CREATE TABLE IF NOT EXISTS items (  
  id bigint UNSIGNED AUTO_INCREMENT PRIMARY KEY,  
  name varchar(255) NOT NULL,  
  price int NOT NULL,  
  stock int DEFAULT 0,  
  created_at DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,  
  updated_at DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP  
);
```

モジュールインストール

mysql2 モジュールをインストールします。

```
% npm i mysql2
```

.env の設定

.env にDB接続設定を記述します。ポートはMySQLのデフォルト 3306とします。ユーザ名、パスワードは環境にあわせてください。

```
HOST=localhost
PORT=3000

DB_NAME=my_page
DB_HOST=localhost
DB_PORT=3306
DB_USER=root
DB_PASSWORD=
```

DB接続テスト

DB接続テストファイル test/connect.js ファイルを作成し、mysql2 モジュールで接続確認します。

モジュール読み込み

mysql2 モジュールを読み込みます。

```
const mysql = require('mysql2');
```

DB接続設定読み込み

dotenv でDB接続情報を読み込んで、db_info に代入します。

```
const dotenv = require('dotenv');
dotenv.config();

const db_info = {
  host: process.env.DB_HOST,
```

```
user: process.env.DB_USER,  
port: process.env.DB_USER,  
password: process.env.DB_PASSWORD,  
}
```

DB接続

`db_info` でDB接続をします。

```
const con = mysql.createConnection(db_info);  
con.connect((err) => {  
  if (err) throw err;  
  console.log('DB connect!!')  
})  
con.end();
```

接続確認

`test/connect.js` を実行して、DB接続を確認します。

```
% node test/connect.js  
DB connect!!
```

アクセス許可がない場合

エラーが発生すると、例外処理でエラーが表示されます。以下はユーザ名、一パスワードが間違った場合です。

```
Error: Access denied for user '@'localhost' (using password: NO)
```

ユーザデータ挿入

`mysql2` で `MySQL` データを挿入します。パスワードハッシュ化モジュール `bcrypt` をインストールします。

```
% npm i bcrypt
```

メインプログラム

`lib/db.js` を作成し、`inserts()` メソッドを実装します。テーブル名とデータを使って、INSERT処理を繰り返し実行します。`query()` の第2引数のデータは「?」にバインドされます。

```
exports.inserts = (table, posts) => {
  let sql = `INSERT INTO ${table} SET ?`;
  const con = mysql.createConnection(db_info)
  posts.forEach((post) => {
    con.query(sql, post);
  })
  con.end();
}
```

ハッシュパスワード

ストレッチング回数を指定して、パスワードをハッシュ化します。

```
bcrypt.hashSync(パスワード, ストレッチング回数);
```

パスワード「password」を、ストレッチング10回でハッシュ化します。

```
const password = bcrypt.hashSync('password', 10);
```

ユーザデータの生成

`test/insert_users.js` に `INSERT` するデータを配列で定義します。

```
const password = bcrypt.hashSync('password', 10);
let posts = [
  { name: 'Adam', email: 'adam@test.com', password: password, 'hobby': '旅行,グルメ' },
  { name: 'Bob', email: 'bob@test.com', password: password, 'hobby': 'スポーツ,音楽' },
  { name: 'Chris', email: 'chris@test.com', password: password, 'hobby': '映画,読書' },
  { name: 'David', email: 'david@test.com', password: password, 'hobby': 'スポーツ,ゲーム' },
  { name: 'Eliza', email: 'eliza@test.com', password: password, 'hobby': 'ゲーム,グルメ' },
];
```

INSERTの実行

`lib/db.js` を読み込んで、`users` テーブルに `inserts()` を実行します。

```
const db = require('../lib/db');
db.inserts('users', posts);
```

INSERTの確認

`test/insert_users.js` を実行して、 `INSERT` 処理を確認します。

```
% node test/insert_users.js
```

演習

商品データの INSERT

以下の商品データを、 `items` テーブルに `INSERT` してみましょう。

```
let posts = [
  { name: 'コーヒー', price: 120, stock: 0, },
  { name: '紅茶', price: 150, stock: 0, },
  { name: 'ほうじ茶', price: 100, stock: 0, },
];
```

当サイトの教材をはじめとするコンテンツ（テキスト、画像等）の無断転載・無断使用を固く禁じます。これらのコンテンツについて権利者の許可なく複製、転用等する事は法律で禁止されています。尚、当ウェブサイトの内容をWeb、雑誌、書籍等へ転載、掲載する場合は「ロジコヤ」までご連絡ください。