Main.js Documentation

Overview

This JavaScript file `main.js` serves as the main entry point for a Node.js application that uses the Express.js framework to create a server. The application interfaces with a MySQL database to retrieve and manipulate data related to users and medicines. It provides API endpoints, some of which are secure and require an API key to access.

Dependencies

The application requires the following npm packages:

- **express**: Web application framework for Node.js
- **body-parser**: Middleware to parse incoming request bodies
- **mysql**: Node.js driver for MySQL to interact with the database

Setup and Middleware

The application sets up Express and the middleware needed for parsing JSON data.

```
const express = require('express');
const bodyParser = require('body-parser');
const app = express();
const port = 3000;

// Middleware to parse JSON data
app.use(bodyParser.json());
```

Database Connection

The application establishes a connection with the MySQL database using the provided credentials.

```
const db = mysql.createConnection({
  host: 'localhost',
  user: 'root',
  password: '',
  database: 'matadrugs'
});

db.connect(err => {
  if (err) {
    console.error('Database connection error:', err.message);
  } else {
    console.log('Connected to the database');
  }
});
```

API Key Verification Middleware

The `verifyApiKey` middleware function is used to authenticate requests using an API key.

```
const verifyApiKey = (req, res, next) => {
  // ... API key verification logic ...
};
```

API Endpoints

The application provides several endpoints, categorized as follows:

User-related Endpoints (Secure)

These endpoints require an API key and allow clients to retrieve user information.

Endpoint	HTTP Method	Description
`/users`	GET	List all users.
`/users/:identifier`	GET	Retrieve one user's details by email or ID.
`/users/gender/:gender`	GET	List all users by gender.
`/users/purchased/:category/:item`	GET	List users who purchased a specific drug or item.
`/users/purchased/date/:date`	GET	List users who purchased a drug on a specific date.
`/users/last-login`	GET	List all users by last login time.

Medicine-related Endpoints (Insecure)

These endpoints do not require an API key and provide information about medicines.

Endpoint	HTTP Method	Description
`/medicine`	GET	List all medicines/items.
`/medicine/:id`	GET	Retrieve drug information by ID.
`/medicine/category/:category`	GET	List all drugs by category/subcategory.
`/medicine/user/:userId`	GET	List drugs by a user.

Server Initialization

The server listens on the specified port for incoming connections.

```
app.listen(port, () => {
  console.log(`Server listening on port ${port}`);
});
```

How to Run the Application

1. Ensure Node.js and npm are installed.
2. Install the required npm packages using `npm install`.
3. Run the server using `node main.js`.

For more detailed instructions, consider creating a `README.md` file alongside the codebase.

Emoji Legend:

- 🔑: Secure endpoint (requires API key)
- 🧑‍⚕️: User-related endpoint
- 💊: Medicine-related endpoint

Remember that the provided code is a template and requires a running MySQL server with the appropriate schema and data. Adjustments might be made to the database connection settings, and API keys should be stored securely, not hard-coded.