MOVIE TICKET BOOKING SYSTEM

TOOLS AND TECHNOLOGIES USED:

* Mysql
* Nodejs

MYSQL SCHEMA:

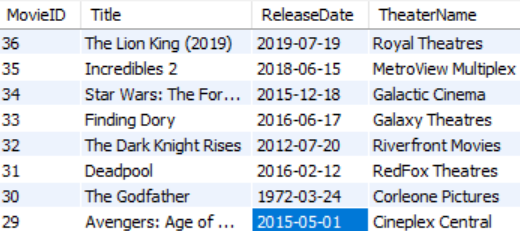
USER TABLE:

Store user data

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **UserID** | **FirstName** | **LastName** | **Email** | **PhoneNumber** | **RegistrationDate** | **Password** |
|  | 2 | John | Doe | johndoe@example.com | 123-456-7890 | 2023-09-04 19:29:46 | jdj |
|  |  |  |  |  |  |  |  |

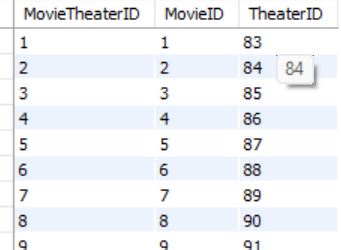
MOVIE:

Store movie and theatre name they are running in



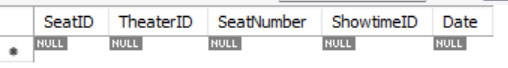
MOVIE THEATER:

Store movie and theatre name they are running in and other details also fetched regrading the address of the theatre.



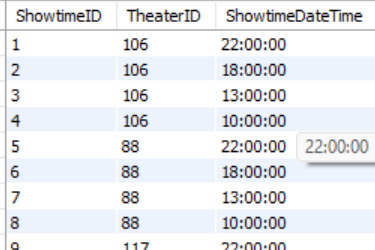
SEAT:

Store the booking of the seat in the theatre.



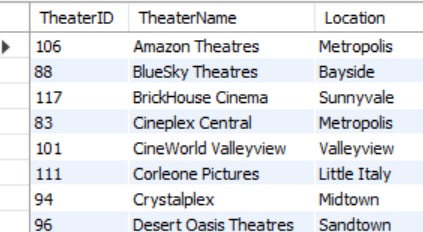
SHOW TIME DETAILS:

Store theatre time



THEATER DETAILS:

Store theatre details.



Code:

**Database connection code:**

const mysql = require('mysql2');

//database connection

const connection = mysql.createConnection({

    host: 'localhost',

    user: 'root',

    password: 'Dharani28',

    database: 'movieticket',

  });

const theaterId = 1; // Replace with the actual theater ID

const showtimes = [1, 2 ,3, 4];

// Generate and insert seat data for each showtime

for (const showtime of showtimes) {

  for (let row = 1; row <= 6; row++) { // Assuming 6 rows of seats

    for (let seatNumber = 1; seatNumber <= 10; seatNumber++) { // 10 seats per row

      const seat = `Row ${row}, Seat ${seatNumber}`;

      const query = 'INSERT INTO Seat (TheaterID, SeatNumber, ShowtimeID) VALUES (?, ?, ?)';

      const values = [theaterId, seat, showtime];

      connection.query(query, values, (err, results) => {

        if (err) {

          console.error('Error inserting seat:', err);

        } else {

          console.log(`Inserted seat: ${seat} for showtime: ${showtime}`);

        }

      });

    }

  }

}

  connection.connect((err) => {

    if (err) {

      console.error('Error connecting to MySQL: ' + err.stack);

      return;

    }

    console.log('Connected to MySQL as id ' + connection.threadId);

  });

  module.exports = connection;

**USER REGISTRATION API CODE:**

const express = require('express');

const router = express.Router();

const db = require('./database'); // Import the MySQL connection

const bodyParser = require('body-parser');

router.use(bodyParser.json());

// Define routes and API endpoints using the 'router' object

router.post('/', (req, res) => {

    const { FirstName, LastName, Email, PhoneNumber, Password } = req.body;

    // Insert the user data into the database

    const sql = 'INSERT INTO User (FirstName, LastName, Email, PhoneNumber, Password) VALUES (?, ?, ?, ?, ?)';

    const values = [FirstName, LastName, Email, PhoneNumber, Password];

    db.query(sql, values, (err, result) => {

      if (err) {

        console.error('Error inserting user:', err);

        res.status(500).json({ error: 'Internal server error' });

        return;

      }

      console.log('User registered:', result.insertId);

      res.status(201).json({ message: 'User registered successfully' });

    });

  });

// Export the router

module.exports = router;

**FETCH MOVIE RUNNING IN THEATER API:**

// api1.js

const express = require('express');

const router = express.Router();

const db = require('./database'); // Import the MySQL connection

// Define routes and API endpoints using the 'router' object

router.get('/movie', (req, res) => {

  // Use 'db' (MySQL connection) to perform database operations

  // ...

  db.query('SELECT \* FROM movies', (error, results, fields) => {

    if (error) {

      console.error('Error fetching data from MySQL: ' + error);

      res.status(500).json({ error: 'Error fetching data from the database' });

      return;

    }

    res.json(results);

  });

});

router.get('/:movieName', (req, res) => {

  const query = req.params.movieName;

  const queryText = query + '%'; // Append '%' to perform a prefix search

    db.query('SELECT MovieName FROM movies WHERE MovieName LIKE ?', [queryText], (error, results, fields) => {

        if (error) {

            console.error('Error fetching data from MySQL: ' + error);

            res.status(500).json({ error: 'Error fetching data from the database' });

            return;

        }

        res.json(results);

    });

});

// Export the router

module.exports = router;

**COLLECTING ALL API IN ONE PLACE:**

const express = require('express');

const app = express();

const mysql = require('mysql2');

//port number

const PORT = process.env.PORT || 3000;

app.listen(PORT, () => {

  console.log(`Server is running on port ${PORT}`);

});

// Import the API modules

const api1 = require('./movie');

const api2 = require('./movie');

const register = require('./UserRegister');

// Use the API modules as middleware

app.use('/movie', api1);

app.use('/movie', api2);

app.use('/register',register);