MYSQL SETUP

1️⃣ Install MySQL Server

1. Go to https://dev.mysql.com/downloads/installer/
2. Download MySQL Installer for Windows (Community Edition).
3. During setup:
   - Choose Developer Default installation.
   - Keep default settings.
   - Set root password and remember it.
   - Keep port 3306.
   - Finish installation.

2️⃣ Start the MySQL Server

Open Command Prompt as Administrator:

```
net start mysql
```

If invalid, try:

```
net start "MySQL80"
```

You should see: The MySQL80 service was started successfully.

3️⃣ Login to MySQL with Local File Enabled

```
mysql --local-infile=1 -u root -p
```

Enter your root password.

4️⃣ Enable Local File Loading

Inside MySQL shell:

```
SHOW VARIABLES LIKE 'local_infile';
```

If it shows OFF, run:

```
SET GLOBAL local_infile = 1;
```

5️⃣ Create the Database

```
CREATE DATABASE IF NOT EXISTS insurance_db;
USE insurance_db;
```

6 Create and Load health_insurance Table

```
DROP TABLE IF EXISTS health_insurance;

CREATE TABLE health_insurance (
  id BIGINT AUTO_INCREMENT PRIMARY KEY,
  policy_id VARCHAR(50),
  customer_id VARCHAR(50),
  customer_name VARCHAR(150),
  age INT,
  gender VARCHAR(10),
  region VARCHAR(50),
  state VARCHAR(100),
  city VARCHAR(100),
  pre_existing_disease VARCHAR(200),
  agent_id VARCHAR(50),
  agent_name VARCHAR(100),
  policy_start_date DATE,
  policy_end_date DATE,
  policy_period_months INT,
  sum_assured BIGINT,
  premium_amount DECIMAL(12,2),
  premium_date DATE,
  premium_frequency VARCHAR(50),
  claim_id VARCHAR(50),
  claim_amount DECIMAL(12,2),
  remaining_amount DECIMAL(12,2),
  claim_date DATE,
  claim_status VARCHAR(50),
  current_status VARCHAR(50)
);
```

```
    LOAD DATA LOCAL INFILE
'C:/Users/manoj/Insurance_data/health_insurance.csv'
    INTO TABLE insurance_db.health_insurance
    CHARACTER SET utf8mb4
    FIELDS TERMINATED BY ','
    ENCLOSED BY '"'
    LINES TERMINATED BY '\n'
    IGNORE 1 LINES
    (policy_id, customer_id, customer_name, age, gender, region, state, city,
     pre_existing_disease, agent_id, agent_name, @policy_start_date,
@policy_end_date,
     policy_period_months, sum_assured, premium_amount, @premium_date,
premium_frequency,
     claim_id, claim_amount, remaining_amount, @claim_date, claim_status,
current_status)
    SET
     policy_start_date = STR_TO_DATE(@policy_start_date, '%Y-%m-%d'),
     policy_end_date = STR_TO_DATE(@policy_end_date, '%Y-%m-%d'),
     premium_date = STR_TO_DATE(@premium_date, '%Y-%m-%d'),
     claim_date = STR_TO_DATE(@claim_date, '%Y-%m-%d');
```

7️⃣ Create and Load car_insurance Table

```
---------------------------------------

    DROP TABLE IF EXISTS car_insurance;

    CREATE TABLE car_insurance (
      id BIGINT AUTO_INCREMENT PRIMARY KEY,
      policy_id VARCHAR(50),
      customer_id VARCHAR(50),
      customer_name VARCHAR(150),
      age INT,
      gender VARCHAR(10),
      region VARCHAR(50),
      state VARCHAR(100),
      city VARCHAR(100),
      agent_id VARCHAR(50),
```

```
    agent_name VARCHAR(100),
    vehicle_reg VARCHAR(50),
    vehicle_make VARCHAR(100),
    vehicle_model VARCHAR(100),
    manufacture_year INT,
    policy_start_date DATE,
    policy_end_date DATE,
    policy_period_months INT,
    sum_assured BIGINT,
    premium_amount DECIMAL(12,2),
    premium_date DATE,
    premium_frequency VARCHAR(50),
    claim_id VARCHAR(50),
    claim_amount DECIMAL(12,2),
    remaining_amount DECIMAL(12,2),
    claim_date DATE,
    claim_status VARCHAR(50),
    current_status VARCHAR(50)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

LOAD DATA LOCAL INFILE 'C:/Users/manoj/Insurance_data/car_insurance.csv'
INTO TABLE insurance_db.car_insurance
CHARACTER SET utf8mb4
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\n'
IGNORE 1 LINES
(policy_id, customer_id, customer_name, age, gender, region, state, city,
 agent_id, agent_name, vehicle_reg, vehicle_make, vehicle_model,
manufacture_year,
 @policy_start_date, @policy_end_date, policy_period_months, sum_assured,
 premium_amount, @premium_date, premium_frequency, claim_id, claim_amount,
 remaining_amount, @claim_date, claim_status, current_status)
SET
 policy_start_date = STR_TO_DATE(@policy_start_date, '%Y-%m-%d'),
 policy_end_date   = STR_TO_DATE(@policy_end_date, '%Y-%m-%d'),
```

```
    premium_date      = STR_TO_DATE(@premium_date, '%Y-%m-%d'),
    claim_date        = STR_TO_DATE(@claim_date, '%Y-%m-%d');
```

8️⃣ Verify Tables Exist

```
-----------------------
    SHOW TABLES;
You should see:
    +-----------------+
    | Tables_in_insurance_db |
    +-----------------+
    | car_insurance    |
    | health_insurance |
    +-----------------+
```

FRONTEND SETUP — React (Create React App)

This is the **dashboard interface** for your analytics tool — where users can upload CSVs, connect to MySQL, apply filters, and visualize charts using **ECharts**.

Prerequisites

- **Node.js v18+**
- **npm** (comes with Node)
- **MySQL** (installed and configured — refer to your MySQL setup guide)

Create the React App

If you are starting from scratch, open a terminal and run:

```
npx create-react-app chola-ms
cd chola-ms
```

If you already have the project folder, just `cd` into it instead.

Install Frontend Dependencies

Run this command inside your `chola-ms` folder:

```
npm install @mui/material @mui/icons-material @emotion/react
@emotion/styled echarts axios papaparse xlsx lodash react-router-dom
react-grid-layout react-resizable
```

✅ This installs:

- **Material UI (MUI)** → UI components
- **ECharts** → Charts (line, bar, pie, scatter)
- **Axios** → For backend API requests
- **PapaParse / XLSX** → CSV + Excel parsing
- **Lodash** → Utility functions
- **React Router DOM** → Navigation
- **Grid & Resizable** → Dashboard layout tools

Start the Frontend

```
npm start
```

This starts your React app at:

http://localhost:3000

BACKEND SETUP — Node.js + Express + MySQL2

Create a Backend Folder

From the project root, create a backend directory:

```
mkdir server
cd server
```

Initialize Node.js

```
npm init -y
```

Install Backend Dependencies

```
npm install express cors body-parser mysql2 dotenv
```

✅ These dependencies do the following:

express	Core web framework
cors	Allows cross-origin access from frontend
mysql2	MySQL connection and query execution
body-parser	Parses JSON bodies

Start the Backend

```
node server.js
```

You will see backend server runs,http://localhost:5000