1. **Project Setup:**
   * Initiate the project by setting up the necessary tools and dependencies, including Node.js and npm for server-side development and React for the frontend.
   * Establish a directory structure for the project to organize server-side (Node.js/Express) and client-side (React) code.
2. **Backend Development (Node.js/Express):**
   * Set up an Express.js server to handle HTTP requests and responses.
   * Connect the server to MongoDB using Mongoose, creating models for the necessary data entities (e.g., products, users, orders).
   * Implement CRUD (Create, Read, Update, Delete) operations to manage data in the MongoDB database.
3. **Authentication and Authorization:**
   * Integrate user authentication using techniques like JWT (JSON Web Tokens) for secure user sessions.
   * Implement user roles and permissions to manage access control within the application.
4. **RESTful API Creation:**
   * Develop RESTful APIs to expose endpoints for functionalities like product listing, user registration, cart management, and order processing.
   * Ensure proper validation and error handling in API routes.
5. **Frontend Development (React):**
   * Set up a React application for the client-side, creating components for different sections of the e-commerce website (e.g., product listings, user authentication, shopping cart).
   * Use React Router for handling navigation and creating a seamless, single-page application experience.
6. **State Management:**
   * Implement state management using tools like Redux to efficiently manage and share application state between React components.
   * Utilize state to handle user interactions, such as adding items to the cart or updating user details.
7. **Responsive Design:**
   * Ensure a responsive design by employing CSS frameworks or libraries to adapt the user interface to different screen sizes and devices.

**8.for admin login**

* Please enter 1 for email and 1 for password