TYPE THE CODE

```jsx
// src/components/MedicalShop.jsx

import React, { useState } from 'react';
const MedicalShop = () => {
  const [inventory, setInventory] = useState([
    { name: 'Paracetamol', quantity: 50, price: 0.5 },
    { name: 'Aspirin', quantity: 30, price: 0.8 },
    { name: 'Band-Aid', quantity: 100, price: 0.2 },
  ]);
  const [formData, setFormData] = useState({
    name: '',
    quantity: '',
    price: '',
  });
  const [search, setSearch] = useState('');
  const handleChange = (e) => {
    const { placeholder, value } = e.target;
    setFormData((prev) => ({
      ...prev,
      [placeholder.toLowerCase()]: value,
    }));
  };
  const handleAddMedicine = () => {
    const { name, quantity, price } = formData;
    if (!name || !quantity || !price) {
      alert('Please fill in all fields');
      return;
    }
    const newMedicine = { name, quantity, price };
    setInventory((prev) => [...prev, newMedicine]);
    setFormData({ name: '', quantity: '', price: '' }); // reset form
  };
  const filteredInventory = inventory.filter((item) =>
    item.name.toLowerCase().includes(search.toLowerCase())
  );
  return (
    <div className="container my-4">
      <h2 className="text-center mb-4">Medical Shop Inventory</h2>
      {/* Search */}
      <input
        type="text"
        placeholder="Search Medicine"
        className="form-control mb-3"
        value={search}
        onChange={(e) => setSearch(e.target.value)}
      />
      {/* Form */}
      <div className="row mb-3">
        <div className="col-md-4">
          <input
            type="text"
            placeholder="Name"
            className="form-control mb-2"
            value={formData.name}
            onChange={handleChange}
          />
        </div>
        <div className="col-md-4">
          <input
            type="number"
            placeholder="Quantity"
            className="form-control mb-2"
```

```jsx
                value={formData.quantity}
                onChange={handleChange}
              />
            </div>
            <div className="col-md-4">
              <input
                type="number"
                placeholder="Price"
                step="0.01"
                className="form-control mb-2"
                value={formData.price}
                onChange={handleChange}
              />
            </div>
          </div>
          <button
            className="btn btn-success w-100 mb-4"
            onClick={handleAddMedicine}
          >
            Add Medicine
          </button>
          {/* Table */}
          <table className="table table-striped table-bordered">
            <thead className="table-dark">
              <tr>
                <th>Name</th>
                <th>Quantity</th>
                <th>Price</th>
              </tr>
            </thead>
            <tbody>
              {filteredInventory.length > 0 ? (
                filteredInventory.map((item, index) => (
                  <tr key={index}>
                    <td>{item.name}</td>
                    <td>{item.quantity}</td>
                    <td>{item.price}</td>
                  </tr>
                ))
              ) : (
                <tr>
                  <td colSpan="3" className="text-center">
                    No medicines found
                  </td>
                </tr>
              )}
            </tbody>
          </table>
        </div>
    );
};
export default MedicalShop;

// src/App.js

import React from 'react';
import MedicalShop from './components/MedicalShop';
import './App.css';
function App() {
    return (
        <div>
            <MedicalShop />
        </div>
    );
```

```
}
export default App;


PRACTICE AT HOME

// src/components/Bookstore.jsx

import React, { useState } from 'react';
const initialInventory = [
  { id: 1, title: 'The Great Gatsby', author: 'F. Scott Fitzgerald', stock: 12,
price: 10.5 },
  { id: 2, title: '1984', author: 'George Orwell', stock: 8, price: 9.0 },
  { id: 3, title: 'To Kill a Mockingbird', author: 'Harper Lee', stock: 15,
price: 12.0 },
];
export default function Bookstore() {
  const [inventory, setInventory] = useState(initialInventory);
  const [newBook, setNewBook] = useState({
    title: '',
    author: '',
    stock: '',
    price: '',
  });
  const handleChange = (e) => {
    const { name, value } = e.target;
    setNewBook((prev) => ({ ...prev, [name]: value }));
  };
  const handleAddBook = () => {
    const { title, author, stock, price } = newBook;
    if (
      title.trim() === '' ||
      author.trim() === '' ||
      stock === '' ||
      price === ''
    ) {
      alert('Please fill in all fields');
      return;
    }
    const nextId =
      inventory.length === 0
        ? 1
        : Math.max(...inventory.map((b) => b.id)) + 1;
    const book = {
      id: nextId,
      title: title.trim(),
      author: author.trim(),
      stock: Number(stock),
      price: Number(price),
    };
    setInventory((prev) => [...prev, book]);
    // clear inputs
    setNewBook({
      title: '',
      author: '',
      stock: '',
      price: '',
    });
  };
  const formatPrice = (n) =>
    `$${Number(n).toFixed(2)}`;
  return (
    <div className="container my-5">
      <h1 className="fw-bold text-primary text-center">Bookstore Inventory</h1>
```

```jsx
      <p className="text-muted text-center">
        Manage your book collection with ease
      </p>
      <h2 className="mt-4 mb-3">Available Books</h2>
      <div className="row g-3">
        {inventory.map((book) => (
          <div className="col-md-4" key={book.id}>
            <div className="card h-100 shadow-sm">
              <div className="card-body">
                <h5 className="card-title text-success">{book.title}</h5>
                <p className="card-subtitle mb-2 text-muted">by
{book.author}</p>
                <p className="mb-1">
                  <span className="fw-bold">Stock:</span> {book.stock}
                </p>
                <p className="mb-0">
                  <span className="fw-bold">Price:</span>
{formatPrice(book.price)}
                </p>
              </div>
              <div className="card-footer bg-transparent border-0">
                <span className="badge bg-info">ID: {book.id}</span>
              </div>
            </div>
          </div>
        ))}
      </div>
      <h2 className="mt-5 mb-3">Add a New Book</h2>
      <div className="card shadow-sm">
        <div className="card-body">
          <div className="row g-3">
            <div className="col-md-3">
              <input
                className="form-control"
                type="text"
                name="title"
                placeholder="Book Title"
                value={newBook.title}
                onChange={handleChange}
              />
            </div>
            <div className="col-md-3">
              <input
                className="form-control"
                type="text"
                name="author"
                placeholder="Author"
                value={newBook.author}
                onChange={handleChange}
              />
            </div>
            <div className="col-md-2">
              <input
                className="form-control"
                type="number"
                name="stock"
                placeholder="Stock"
                min="0"
                value={newBook.stock}
                onChange={handleChange}
              />
            </div>
            <div className="col-md-2">
              <input
```

```jsx
                    className="form-control"
                    type="number"
                    name="price"
                    placeholder="Price ($)"
                    step="0.01"
                    min="0"
                    value={newBook.price}
                    onChange={handleChange}
                  />
                </div>
                <div className="col-md-2 d-grid">
                  <button
                    className="btn btn-success w-100"
                    onClick={handleAddBook}
                  >
                    Add Book
                  </button>
                </div>
              </div>
            </div>
          </div>
        </div>
      );
}

// src/App.js

import React from 'react';
import Bookstore from './components/Bookstore';
export default function App() { return <Bookstore />; }


CHALLENGE YOURSELF

// src/components/MusicStore.jsx

import React, { useState } from 'react';
const initialItems = [
  { id: 1, name: 'Acoustic Guitar', quantity: 10, price: 150 },
  { id: 2, name: 'Electric Keyboard', quantity: 5, price: 300 },
  { id: 3, name: 'Drum Set', quantity: 3, price: 500 },
];
export default function MusicStore() {
  const [inventory, setInventory] = useState(initialItems);
  const [newItem, setNewItem] = useState({ name: '', quantity: '', price: '' });
  const onChange = (e) => {
    const { name, value } = e.target;
    setNewItem((prev) => ({ ...prev, [name]: value }));
  };
  const addItem = () => {
    const { name, quantity, price } = newItem;
    if (name.trim() === '' || quantity === '' || price === '') {
      alert('Please fill in all fields');
      return;
    }
    const nextId =
      inventory.length === 0 ? 1 : Math.max(...inventory.map((i) => i.id)) + 1;
    setInventory((prev) => [
      ...prev,
      {
        id: nextId,
        name: name.trim(),
        quantity: Number(quantity),
        price: Number(price),
```

```jsx
          },
        ]);
        setNewItem({ name: '', quantity: '', price: '' });
      };
      const money = (n) => `$${Number(n).toFixed(2)}`;
      return (
        <div className="container my-4">
          {/* Header bar styled to resemble screenshot */}
          <div className="card shadow-sm mb-3">
            <div className="card-body bg-dark text-warning text-center">
              <h1 className="m-0">Music Store Inventory</h1>
            </div>
          </div>
          <div className="card h-100 shadow-sm">
            <div className="card-body">
              <div className="table-responsive">
                <table className="table table-bordered table-hover shadow-sm m-0">
                  <thead className="table-warning">
                    <tr>
                      <th scope="col">ID</th>
                      <th scope="col">Instrument / Album</th>
                      <th scope="col">Quantity</th>
                      <th scope="col">Price ($)</th>
                    </tr>
                  </thead>
                  <tbody>
                    {inventory.map((item) => (
                      <tr key={item.id}>
                        <td>
                          {item.id}
                        </td>
                        <td>{item.name}</td>
                        <td>{item.quantity}</td>
                        <td>{money(item.price)}</td>
                      </tr>
                    ))}
                  </tbody>
                </table>
              </div>
              {/* Add form */}
              <h2 className="text-info text-center mt-4 mb-3">Add New
Instrument</h2>
              <div className="row g-2 align-items-stretch">
                <div className="col-md-5">
                  <input
                    className="form-control border-info"
                    type="text"
                    name="name"
                    placeholder="Instrument / Album Name"
                    value={newItem.name}
                    onChange={onChange}
                  />
                </div>
                <div className="col-md-2">
                  <input
                    className="form-control border-info"
                    type="number"
                    name="quantity"
                    placeholder="Quantity"
                    min="0"
                    value={newItem.quantity}
                    onChange={onChange}
                  />
                </div>
```

```jsx
              <div className="col-md-3">
                <input
                  className="form-control border-info"
                  type="number"
                  name="price"
                  placeholder="Price ($)"
                  step="0.01"
                  min="0"
                  value={newItem.price}
                  onChange={onChange}
                />
              </div>
              <div className="col-md-2 d-grid">
                <button className="btn btn-warning w-100 fw-bold"
onClick={addItem}>
                  Add Item
                </button>
              </div>
            </div>
          </div>
        </div>
      </div>
    );
}


// src/App.js

import React from 'react';
import MusicStore from './components/MusicStore';
export default function App() {
  return <MusicStore />;
}


FIX THE CODE

// src/components/FoodOrder.jsx

import React, { useState } from 'react';
import 'bootstrap/dist/css/bootstrap.min.css';
const initialOrders = [
  { id: 1, item: 'Burger', quantity: 2, price: 150 },
  { id: 2, item: 'Pizza', quantity: 1, price: 300 },
  { id: 3, item: 'Pasta', quantity: 3, price: 200 },
];
export default function FoodOrder() {
  const [orders, setOrders] = useState(initialOrders);
  const [newOrder, setNewOrder] = useState({ item: '', quantity: '', price:
'' });
  const handleNewOrderChange = (e) => {
    const { name, value } = e.target;
    setNewOrder((prev) => ({ ...prev, [name]: value }));
  };
  const handleAddOrder = () => {
    const { item, quantity, price } = newOrder;
    if (item.trim() === '' || quantity === '' || price === '') {
      alert('Please fill in all fields');
      return;
    }
    const nextId = orders.length === 0 ? 1 : Math.max(...orders.map((o) =>
o.id)) + 1;
    const toAdd = {
      id: nextId,
      item: item.trim(),
```

```jsx
      quantity: Number(quantity),
      price: Number(price),
    };
    setOrders((prev) => [...prev, toAdd]);
    setNewOrder({ item: '', quantity: '', price: '' });
  };
  return (
    <div className="container my-4">
      <h1 className="text-primary text-center mb-4">Food Order Management</h1>
      <div className="table-responsive">
        <table className="table table-striped table-bordered" role="table">
          <thead>
            <tr>
              <th>ID</th>
              <th>Food Item</th>
              <th>Quantity</th>
              <th>Price ($)</th>
            </tr>
          </thead>
          <tbody>
            {orders.map((o) => (
              <tr key={o.id}>
                <td>{o.id}</td>
                <td>{o.item}</td>
                <td>{o.quantity}</td>
                <td>{o.price}</td>
              </tr>
            ))}
          </tbody>
        </table>
      </div>
      <h2 className="text-success text-center mt-5 mb-3">Add New Food Order</h2>
      <div className="row g-3">
        <div className="col-md-5">
          <input
            type="text"
            className="form-control"
            placeholder="Food Item"
            name="item"
            value={newOrder.item}
            onChange={handleNewOrderChange}
          />
        </div>
        <div className="col-md-3">
          <input
            type="number"
            className="form-control"
            placeholder="Quantity"
            name="quantity"
            value={newOrder.quantity}
            onChange={handleNewOrderChange}
            min="0"
          />
        </div>
        <div className="col-md-2">
          <input
            type="number"
            className="form-control"
            placeholder="Price ($)"
            name="price"
            value={newOrder.price}
            onChange={handleNewOrderChange}
            step="0.01"
            min="0"
```

```
          />
        </div>
        <div className="col-md-2 d-grid">
          <button className="btn btn-success w-100" onClick={handleAddOrder}>
            Add Order
          </button>
        </div>
      </div>
    </div>
  );
}

// src/App.js

import React from 'react';
import FoodOrder from './components/FoodOrder';
export default function App() {
  return <FoodOrder />;
}
```

MCQ

1. What is the purpose of 'npm start' command?
To start the React application

2. What is JSX?
A way to write HTML code in JavaScript

3. What is JSX and why is it used in React?
JSX is a syntax extension for JavaScript that allows us to write HTML in our
JavaScript code and is used in React for rendering templates

4. What is the React Hook `useState` used for?
It is used to manage the state of a React component

5. What is the difference between a functional component and a class component
in React?
Functional components are stateless, whereas class components are stateful

6. What is the role of 'ReactComponent'
To create a new React component

7. What is the purpose of the 'console.log' statement in React?
To debug a React application

8. What is the purpose of the `index.js` file in a React project?
It contains the entry point of the application

9. What is the purpose of the 'import' keyword in React?
To import a module

10. What is the default file extension for a React component?
jsx

11. What is the primary purpose of 'npm init' command?
To create a package.json file

12. What is the purpose of the `key` attribute in JSX?
It is used to identify the React component in the DOM

13. What is the React component lifecycle method `componentDidUpdate` used for?
It is used after a React component has been updated but its props have not
changed

14. What is the role of a 'ReactDOM.render' in a React application?
To render a React component to the DOM

15. Which of the following commands sets up a new React project using Create React App?
npx create-react-app my-app