

Restaurant:

```
import React, { useState } from 'react';
import './App.css';
function App() {
  const [formData, setFormData] = useState({
    customerName: "",
    contactNumber: "",
    menuItem: "",
    itemQuantity: 1,
    specialInstructions: "",
    termsAccepted: false,
  });
  const [successMessage, setSuccessMessage] = useState("");
  const [errorMessage, setErrorMessage] = useState("");
  const [menuItems] = useState(['Pizza', 'Burger', 'Pasta', 'Salad', 'Sushi']);

  const handleInputChange = (e) => {
    const { name, value, type, checked } = e.target;
    setFormData({
      ...formData,
      [name]: type === 'checkbox' ? checked : value,
    });
  };

  const validateForm = () => {
    setErrorMessage(""); // Reset error message
```

```
    if (!formData.customerName || !formData.contactNumber ||  
    !formData.menuItem || !formData.termsAccepted) {  
        setErrorMessage('Please fill out all required fields.');
```

```
        return false;
```

```
    }
```

```
    if (formData.contactNumber.length !== 10) {
```

```
        setErrorMessage('Contact number must be 10 digits.');
```

```
        return false;
```

```
    }
```

```
    if (formData.itemQuantity < 1 || formData.itemQuantity > 3) {
```

```
        setErrorMessage('Quantity must be between 1 and 3.');
```

```
        return false;
```

```
    }
```

```
    return true;
```

```
};
```

```
const handleSubmit = (e) => {
```

```
    e.preventDefault();
```

```
    if (validateForm()) {
```

```
        setSuccessMessage(`Order successful! You ordered  
${formData.itemQuantity} ${formData.menuItem}(s).`);
```

```
        setErrorMessage("");
```

```
    } else {
```

```
    setSuccessMessage("");
  }
};

const handleReset = () => {
  setFormData({
    customerName: "",
    contactNumber: "",
    menuItem: "",
    itemQuantity: 1,
    specialInstructions: "",
    termsAccepted: false,
  });
  setSuccessMessage("");
  setErrorMessage("");
};

return (
  <div className="App">
    <header>
      <h1>Restaurant Menu Order</h1>
    </header>
    <div className="content">
      <form onSubmit={handleSubmit}>
        <div className="form-group">
          <label htmlFor="customerName">Customer Name:</label>
          <input
            type="text"
```

```
      id="customerName"
      name="customerName"
      value={formData.customerName}
      onChange={handleInputChange}
      required
    />
  </div>
```

```
<div className="form-group">
  <label htmlFor="contactNumber">Contact Number:</label>
  <input
    type="text"
    id="contactNumber"
    name="contactNumber"
    value={formData.contactNumber}
    onChange={handleInputChange}
    required
    maxLength="10"
  />
</div>
```

```
<div className="form-group">
  <label htmlFor="menuItem">Menu Item:</label>
  <select
    id="menuItem"
    name="menuItem"
```

```
      value={formData.menuItem}
      onChange={handleInputChange}
      required
    >
    <option value="">Select an item</option>
    {menuItems.map((item, index) => (
      <option key={index} value={item}>{item}</option>
    ))}
  </select>
</div>
```

```
<div className="form-group">
  <label htmlFor="itemQuantity">Item Quantity:</label>
  <input
    type="number"
    id="itemQuantity"
    name="itemQuantity"
    value={formData.itemQuantity}
    onChange={handleInputChange}
    min="1"
    max="3"
    required
  />
</div>
```

```
<div className="form-group">
```

```
<label htmlFor="specialInstructions">Special Instructions:</label>
```

```
<textarea
```

```
  id="specialInstructions"
```

```
  name="specialInstructions"
```

```
  value={formData.specialInstructions}
```

```
  onChange={handleInputChange}
```

```
></textarea>
```

```
</div>
```

```
<div className="form-group">
```

```
  <label>
```

```
    <input
```

```
      type="checkbox"
```

```
      name="termsAccepted"
```

```
      checked={formData.termsAccepted}
```

```
      onChange={handleInputChange}
```

```
      required
```

```
    />
```

```
    I agree to the Terms and Conditions
```

```
  </label>
```

```
</div>
```

```
<div className="form-group">
```

```
  <button type="submit" className="btn">Order Now</button>
```

```
  <button          type="button"          className="btn          reset"
onClick={handleReset}>Reset</button>
```

```
</div>
```

```
</form>
```

```
{errorMessage &&
```

```
<div className="error-message">{errorMessage}</div>}
```

```
{successMessage &&
```

```
<div className="success-message">{successMessage}</div>}
```

```
</div>
```

```
<footer>
```

```
<p>© 2024 Restaurant Name. All rights reserved.</p>
```

```
</footer>
```

```
</div>
```

```
);
```

```
}
```

```
export default App;
```

```
App.css
```

```
body {
```

```
font-family: Arial, sans-serif;
```

```
margin: 0;
```

```
padding: 0;
```

```
background-color: #f4f4f4;
```

```
color: #333;
```

```
}
```

```
.App {
```

```
max-width: 800px;
```

```
margin: 0 auto;
```

```
padding: 20px;
background-color: #fff;
border-radius: 8px;
box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}
```

```
header {
  text-align: center;
  padding: 10px 0;
  background-color: #4CAF50;
  color: white;
  border-radius: 8px;
}
```

```
h1 {
  margin: 0;
}
```

```
.content {
  margin: 20px 0;
}
```

```
.form-group {
  margin-bottom: 15px;
}
```



```
.form-group label {  
  display: block;  
  margin-bottom: 5px;  
  font-weight: bold;  
  color: #4CAF50;  
}
```

```
.form-group input,  
.form-group select,  
.form-group textarea {  
  width: 100%;  
  padding: 8px;  
  margin: 4px 0;  
  border: 1px solid #ccc;  
  border-radius: 4px;  
  box-sizing: border-box;  
}
```

```
.form-group textarea {  
  height: 80px;  
  resize: vertical;  
}
```

```
.btn {  
  display: inline-block;  
  padding: 10px 20px;
```

```
margin: 5px 0;
color: white;
background-color: #007BFF;
border: none;
border-radius: 4px;
cursor: pointer;
}
```

```
.btn.reset {
  background-color: #dc3545;
}
```

```
.btn:hover {
  opacity: 0.9;
}
```

```
.success-message {
  margin: 20px 0;
  color: #28a745;
  font-weight: bold;
}
```

```
.error-message {
  margin: 20px 0;
  color: #dc3545;
  font-weight: bold;
}
```

```
}
```

```
footer {  
  text-align: center;  
  margin-top: 20px;  
  padding: 10px 0;  
  background-color: #f1f1f1;  
  border-radius: 8px;  
}
```

```
footer p {  
  margin: 0;  
  color: #666;  
}
```

The screenshot shows a web browser window with the address bar displaying "127.0.0.1:5500/OrderForm.html". The page has a yellow header bar with the text "Restaurant Name". Below the header, there is a form with the following fields: "Customer Name:", "Contact Number:", "Menu Item:", "Item Quantity:", and "Special Instructions:". Below these fields, there is a checkbox for "I agree to the [Terms and Conditions](#)". At the bottom of the form, there are two buttons: "Order Now" and "Reset". Below the form, there is a section titled "Popular Dishes" which contains three items: "Dish 1", "Dish 2", and "Dish 3", each with a small image icon. At the very bottom of the page, there is a dark gray footer bar with the text "© 2024 Restaurant Name. All rights reserved."

### Shopping app:

```
import './App.css';  
  
import Navbar from './components/Navbar';  
import Footer from './components/Footer';  
import Products from './components/Products';
```

```
function App() {  
  return (  
    <div className="App">  
      <Navbar/>  
      <Products/>  
      <Footer/>  
    </div>  
  );  
}  
  
export default App;  
  
Navbar.js  
  
import React from 'react';  
import './App.css'  
  
const Navbar = () => {  
  return (  
    <div className='Navbar'>  
      <h1>Shopping App</h1>  
    </div>  
  );  
}  
export default Navbar;
```

Products.js

```
import React, { useState, useEffect } from 'react';
```

```
import '../App.css';
```

```
import axios from 'axios';
```

```
const Products = () => {
```

```
  const [loading, setLoading] = useState(false);
```

```
  const [data, setData] = useState([]);
```

```
  useEffect(() => {
```

```
    setLoading(true);
```

```
    axios({
```

```
      method: "GET",
```

```
      url: "https://fakestoreapi.com/products"
```

```
    })
```

```
    .then(res => {
```

```
      console.log(res.data);
```

```
      setData(res.data);
```

```
    })
```

```
    .catch(e => console.log(e))
```

```
    .finally(() => setLoading(false));
```

```
  }, []);
```

```
  return (
```

```
    <div className='products-container'>
```

```
      {loading && (
```

```
        <div>Loading...</div>
```

```
      )}
```

```
      {!loading && data.length > 0 && (
```

```

    data.map(product => (
      <div key={product.id} className="card">
        <div>
          <img src={product.image} alt={product.title} />
        </div>
        <div className='card-description'>
          <h6>{product.title}</h6>
          <p>Category: {product.category}</p>
          <p>Price: ${product.price}</p>
        </div>
      </div>
    ))
  )}
</div>

);
}

export default Products;

Footer.js

import React from 'react';
import '../App.css'

const Footer = () => {
  return (
    <div className='Footer'>
      <p>Shopping app 2024&copy;</p>
    </div>
  );
}

```

```
}  
export default Footer;  
App.css  
body {  
  font-family: 'Helvetica Neue', Arial, sans-serif;  
  margin: 0;  
  padding: 0;  
  background-color: #f8f9fa;  
}  
.Navbar {  
  background-color: #007bff;  
  color: white;  
  padding: 15px 20px;  
  text-align: center;  
  box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);  
}  
.Navbar h1 {  
  margin: 0;  
  font-size: 24px;  
  font-weight: 500;  
}  
.Footer {  
  background-color: #343a40;  
  color: white;  
  padding: 10px 20px;  
  text-align: center;
```

```
position: fixed;
width: 100%;
bottom: 0;
box-shadow: 0 -2px 4px rgba(0, 0, 0, 0.1);
}

.Footer p {
margin: 0;
font-size: 14px;
}

.products-container {
padding: 20px;
display: grid;
grid-template-columns: repeat(3, 1fr);
gap: 20px;
background-color: #f8f9fa;
min-height: calc(100vh - 120px);
}

.card {
background-color: #fff;
border: 1px solid #ddd;
border-radius: 8px;
box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
overflow: hidden;
text-align: left;
transition: transform 0.3s ease, box-shadow 0.3s ease;
cursor: pointer;
```



```
}  
.card:hover {  
  transform: translateY(-5px);  
  box-shadow: 0 8px 16px rgba(0, 0, 0, 0.2);  
}  
.card img {  
  width: 100%;  
  height: 200px;  
  object-fit: contain;  
}  
.card-description {  
  padding: 15px;  
  display: flex;  
  flex-direction: column;  
  gap: 10px;  
}  
.card-description h6 {  
  margin: 0;  
  font-size: 16px;  
  font-weight: 600;  
  color: #333;  
}  
.card-description p {  
  margin: 0;  
  color: #555;  
  font-size: 14px;
```

```

}

.card-description .price {
  font-size: 18px;
  font-weight: bold;
  color: #007bff;
}

.products-container > div {
  font-size: 18px;
  color: #333;
  text-align: center;
  width: 100%;
  margin-top: 50px;
}

```



