

## App.js

```
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;
}
```


Shopping App




**Fjallraven - Foldsack No. 1 Backpack, Fits 15 Laptops**  
Category: men's clothing  
Price: \$109.95




**Mens Casual Premium Slim Fit T-Shirts**  
Category: men's clothing  
Price: \$22.3




**Mens Cotton Jacket**  
Category: men's clothing  
Price: \$55.99




**Mens Casual Slim Fit**  
Category: men's clothing  
Price: \$15.99




**John Hardy Women's Legends Naga Gold & Silver Dragon Station Chain Bracelet**  
Category: jewelry  
Price: \$695




**Solid Gold Petite Micropave**  
Category: jewelry  
Price: \$168




**White Gold Plated Princess**  
Category: jewelry  
Price: \$9.99




**Pierced Owl Rose Gold Plated Stainless Steel Double**  
Category: jewelry  
Price: \$10.99




**WD 2TB Elements Portable External Hard Drive - USB 3.0**  
Category: electronics  
Price: \$64



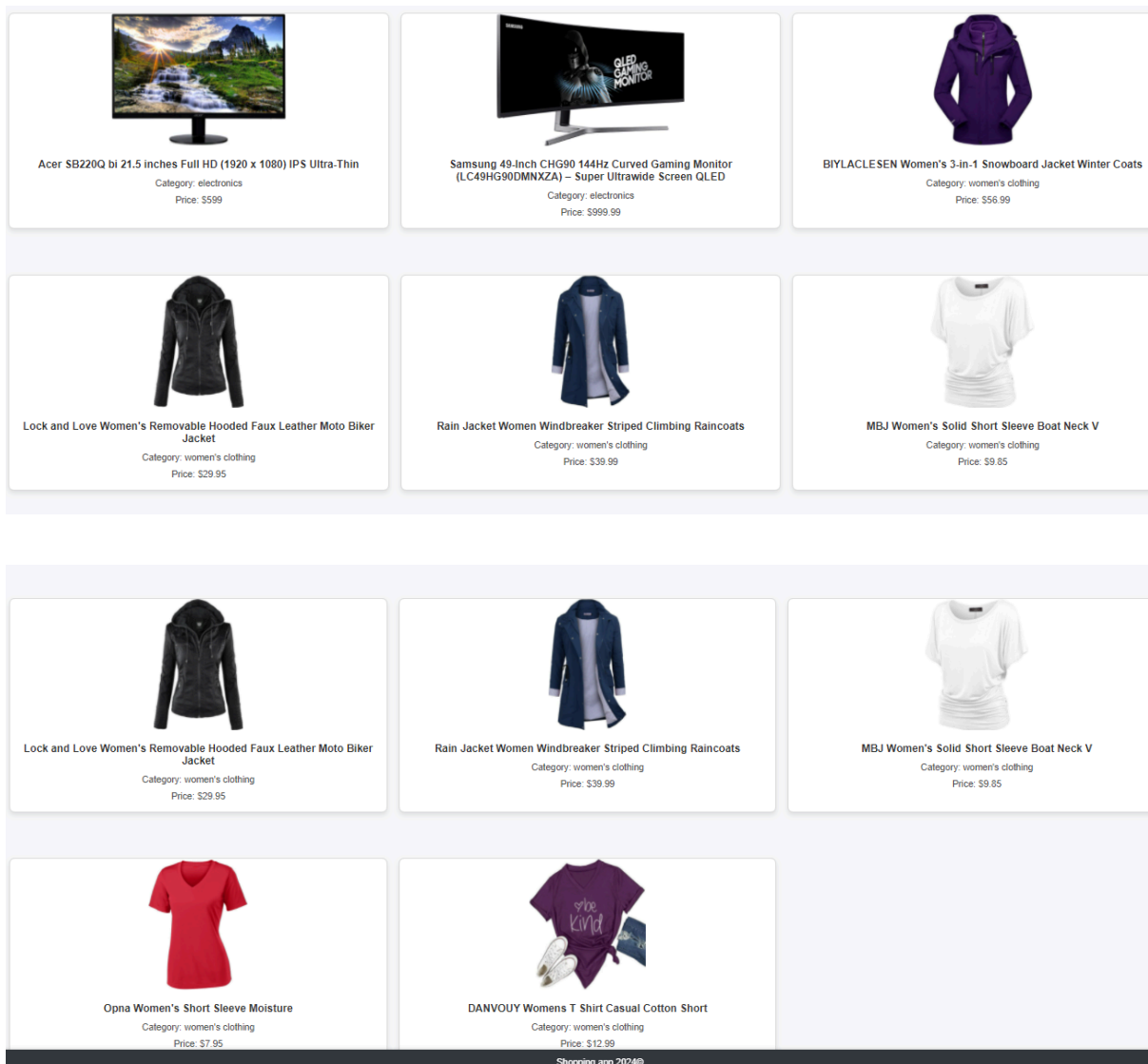
**SanDisk SSD PLUS 1TB Internal SSD - SATA III 6 Gb/s**  
Category: electronics  
Price: \$109



**Silicon Power Z56GB SSD 3D NAND A55 SLC Cache Performance Boost SATA III 2.5**  
Category: electronics  
Price: \$109



**WD 4TB Gaming Drive Works with Playstation 4 Portable External Hard Drive**  
Category: electronics  
Price: \$114



Restaurant app:

App.js:

```
import React, { useState } from 'react';
import './App.css';

const App = () => {
  const [customerName, setCustomerName] = useState("");
  const [customerNumber, setCustomerNumber] = useState("");
  const [selectedItem, setSelectedItem] = useState('1');
  const [itemSize, setItemSize] = useState('Medium');
  const [quantity, setQuantity] = useState(1);
  const [specialInstructions, setSpecialInstructions] = useState("");
  const [toppings, setToppings] = useState([]);
  const [orderType, setOrderType] = useState('pickup');
```

```

const [termsAccepted, setTermsAccepted] = useState(false);
const [orderDetails, setOrderDetails] = useState(null);
const [errors, setErrors] = useState({});

const menuItems = [
  { id: '1', name: 'Pizza' },
  { id: '2', name: 'Burger' },
  { id: '3', name: 'Pasta' },
];

const images = [
  '/images/pizza.jpg',
  '/images/burger.jpg',
  '/images/pasta.jpg',
];

const handleSubmit = (e) => {
  e.preventDefault();

  const newErrors = {};
  if (customerNumber.length !== 10) newErrors.customerNumber = 'Contact number must be exactly 10 digits.';
  if (quantity < 1 || quantity > 3) newErrors.quantity = 'Quantity must be between 1 and 3.';
  if (!termsAccepted) newErrors.terms = 'You must accept the terms and conditions.';

  if (Object.keys(newErrors).length > 0) {
    setErrors(newErrors);
    return;
  }

  const order = {
    customerName,
    customerNumber,
    menuItem: menuItems.find(item => item.id === selectedItem).name,
    itemSize,
    quantity,
    toppings,
    specialInstructions,
    orderType,
    totalOrder: quantity
  };

  setOrderDetails(order);
  setErrors({});
};

const handleReset = () => {
  setCustomerName('');

```



```

    setCustomerNumber("");
    setSelectedItem('1');
    setItemSize('Medium');
    setQuantity(1);
    setSpecialInstructions("");
    setToppings([]);
    setOrderType('pickup');
    setTermsAccepted(false);
    setOrderDetails(null);
    setErrors({});
  };

```

```

const handleToppingsChange = (e) => {
  const { value, checked } = e.target;
  if (checked) {
    setToppings([...toppings, value]);
  } else {
    setToppings(toppings.filter(topping => topping !== value));
  }
};

```

```

return (
  <div className="App">
    <header className="App-header">
      <h1>Restaurant Page</h1>
    </header>
    <main className="container">
      <section id="order">
        <div className="form-container">
          <h1>Restaurant Order Form</h1>
          <form onSubmit={handleSubmit}>
            <label>
              Customer Name:
              <input type="text" value={customerName} onChange={(e) =>
setCustomerName(e.target.value)} />
            </label>
            <label>
              Customer Number:
              <input type="text" value={customerNumber} onChange={(e) =>
setCustomerNumber(e.target.value)} maxLength="10" />
              {errors.customerNumber && <p
className="error">{errors.customerNumber}</p>}
            </label>
            <label>
              Menu Item:
              <select value={selectedItem} onChange={(e) => setSelectedItem(e.target.value)}>
                {menuItems.map(item => (
                  <option key={item.id} value={item.id}>{item.name}</option>

```

```

    ))}
  </select>
</label>
<label>
  Size:
  <select value={itemSize} onChange={(e) => setItemSize(e.target.value)}>
    <option value="Small">Small</option>
    <option value="Medium">Medium</option>
    <option value="Large">Large</option>
  </select>
</label>
<label>
  Quantity:
  <input type="number" value={quantity} onChange={(e) => setQuantity(Math.min(3,
Math.max(1, e.target.value)))} />
  {errors.quantity && <p className="error">{errors.quantity}</p>}
</label>
<label>
  Extra Toppings:
  <div className="toppings">
    <label>
      <input type="checkbox" value="Cheese" checked={toppings.includes('Cheese')}
onChange={handleToppingsChange} />
      Cheese
    </label>
    <label>
      <input type="checkbox" value="Pepperoni"
checked={toppings.includes('Pepperoni')} onChange={handleToppingsChange} />
      Pepperoni
    </label>
    <label>
      <input type="checkbox" value="Mushrooms"
checked={toppings.includes('Mushrooms')} onChange={handleToppingsChange} />
      Mushrooms
    </label>
  </div>
</label>
<label>
  Order Type:
  <div className="order-type">
    <label>
      <input type="radio" value="pickup" checked={orderType === 'pickup'}
onChange={(e) => setOrderType(e.target.value)} />
      Pickup
    </label>
    <label>
      <input type="radio" value="delivery" checked={orderType === 'delivery'}
onChange={(e) => setOrderType(e.target.value)} />

```

```

        Delivery
      </label>
    </div>
  </label>
  <label>
    Special Instructions:
    <textarea value={specialInstructions} onChange={(e) =>
setSpecialInstructions(e.target.value)} />
  </label>
  <label>
    <input type="checkbox" checked={termsAccepted} onChange={(e) =>
setTermsAccepted(e.target.checked)} />
    I accept the terms and conditions
  </label>
  {errors.terms && <p className="error">{errors.terms}</p>}
  <button type="submit">Order Now</button>
  <button type="button" onClick={handleReset}>Reset</button>
</form>
{orderDetails && (
  <div className="order-summary-modal">
    <div className="modal-content">
      <h2>Order Details</h2>
      <p>Customer Name: {orderDetails.customerName}</p>
      <p>Customer Number: {orderDetails.customerNumber}</p>
      <p>Menu Item: {orderDetails.menuItem} - {orderDetails.itemSize}</p>
      <p>Quantity: {orderDetails.quantity}</p>
      <p>Toppings: {orderDetails.toppings.join(', ')}</p>
      <p>Order Type: {orderDetails.orderType}</p>
      <p>Special Instructions: {orderDetails.specialInstructions}</p>
      <p>Total Order: {orderDetails.totalOrder}</p>
    </div>
  </div>
)}
</div>
</section>
</main>
<footer>
  <p>&copy; 2024 Restaurant</p>
</footer>
</div>
);
};

export default App;

App.css:

/* App.css */

```

```
body {  
  font-family: Arial, sans-serif;  
  background-color: #f8f9fa;  
  margin: 0;  
  padding: 0;  
}
```

```
.App-header {  
  background-color: #343a40;  
  color: white;  
  padding: 20px;  
  text-align: center;  
}
```

```
.container {  
  padding: 20px;  
  max-width: 1200px;  
  margin: 0 auto;  
}
```

```
nav ul {  
  list-style: none;  
  padding: 0;  
  display: flex;  
  justify-content: center;  
  margin-bottom: 20px;  
}
```

```
nav ul li {  
  margin: 0 15px;  
}
```

```
nav ul li a {  
  text-decoration: none;  
  color: #007bff;  
  font-weight: bold;  
}
```

```
nav ul li a:hover {  
  text-decoration: underline;  
}
```

```
.form-container {  
  background-color: rgb(104, 83, 201);  
  border: 1px solid #ced4da;  
  border-radius: 5px;  
  padding: 20px;  
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);  
}
```

```
}
```

```
.form-container h1 {  
  margin-top: 0;  
  margin-bottom: 20px;  
  color: #343a40;  
}
```

```
label {  
  display: block;  
  margin-bottom: 10px;  
  font-weight: bold;  
}
```

```
input[type="text"],  
input[type="number"],  
select,  
textarea {  
  width: 100%;  
  padding: 10px;  
  margin-top: 5px;  
  margin-bottom: 15px;  
  border: 1px solid #ced4da;  
  border-radius: 4px;  
  box-sizing: border-box;  
}
```

```
input[type="checkbox"],  
input[type="radio"] {  
  margin-right: 10px;  
}
```

```
input[type="checkbox"]:not(:last-child),  
input[type="radio"]:not(:last-child) {  
  margin-bottom: 10px;  
}
```

```
textarea {  
  height: 80px;  
}
```

```
button[type="submit"],  
button[type="button"] {  
  padding: 10px 20px;  
  background-color: #007bff;  
  color: white;  
  border: none;  
  border-radius: 5px;
```

```
    cursor: pointer;
    margin-right: 10px;
    margin-top: 15px;
}
```

```
button[type="button"] {
    background-color: #6c757d;
}
```

```
button:hover {
    background-color: #0056b3;
}
```

```
button[type="button"]:hover {
    background-color: #5a6268;
}
```

```
.error {
    color: red;
    font-size: 0.875rem;
    margin-top: -10px;
    margin-bottom: 10px;
}
```

```
.gallery {
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));
    gap: 15px;
}
```

```
.gallery img {
    width: 100%;
    height: auto;
    border-radius: 5px;
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
}
```

```
.order-summary-modal {
    position: fixed;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    background-color: rgba(0, 0, 0, 0.5);
    display: flex;
    align-items: center;
    justify-content: center;
}
```

```

.modal-content {
  background-color: white;
  padding: 20px;
  border-radius: 5px;
  max-width: 500px;
  width: 100%;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
}

.modal-content h2 {
  margin-top: 0;
  color: #343a40;
}

footer {
  background-color: #343a40;
  color: white;
  text-align: center;
  padding: 15px;
  position: fixed;
  bottom: 0;
  width: 100%;
  box-shadow: 0 -4px 8px rgba(0, 0, 0, 0.1);
}

```

Restaurant Page

Restaurant Order Form

Customer Name:

Customer Number:

Menu Item:

Pizza

Size:

Medium

Quantity:

1

Extra Toppings:

☐ Cheese

☐ Pepporoni

☐ Mushrooms

Order Type:

☐ Pickup

☐ Delivery

Special Instructions:

☐ I accept the terms and conditions

Order Now

Reset

© 2024 Restaurant

