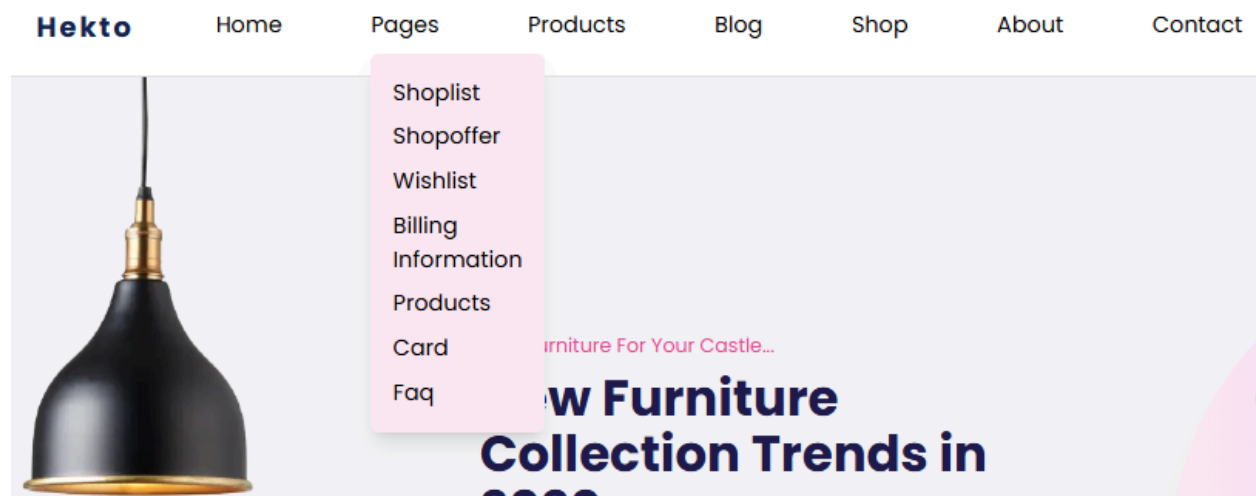


# Day 4

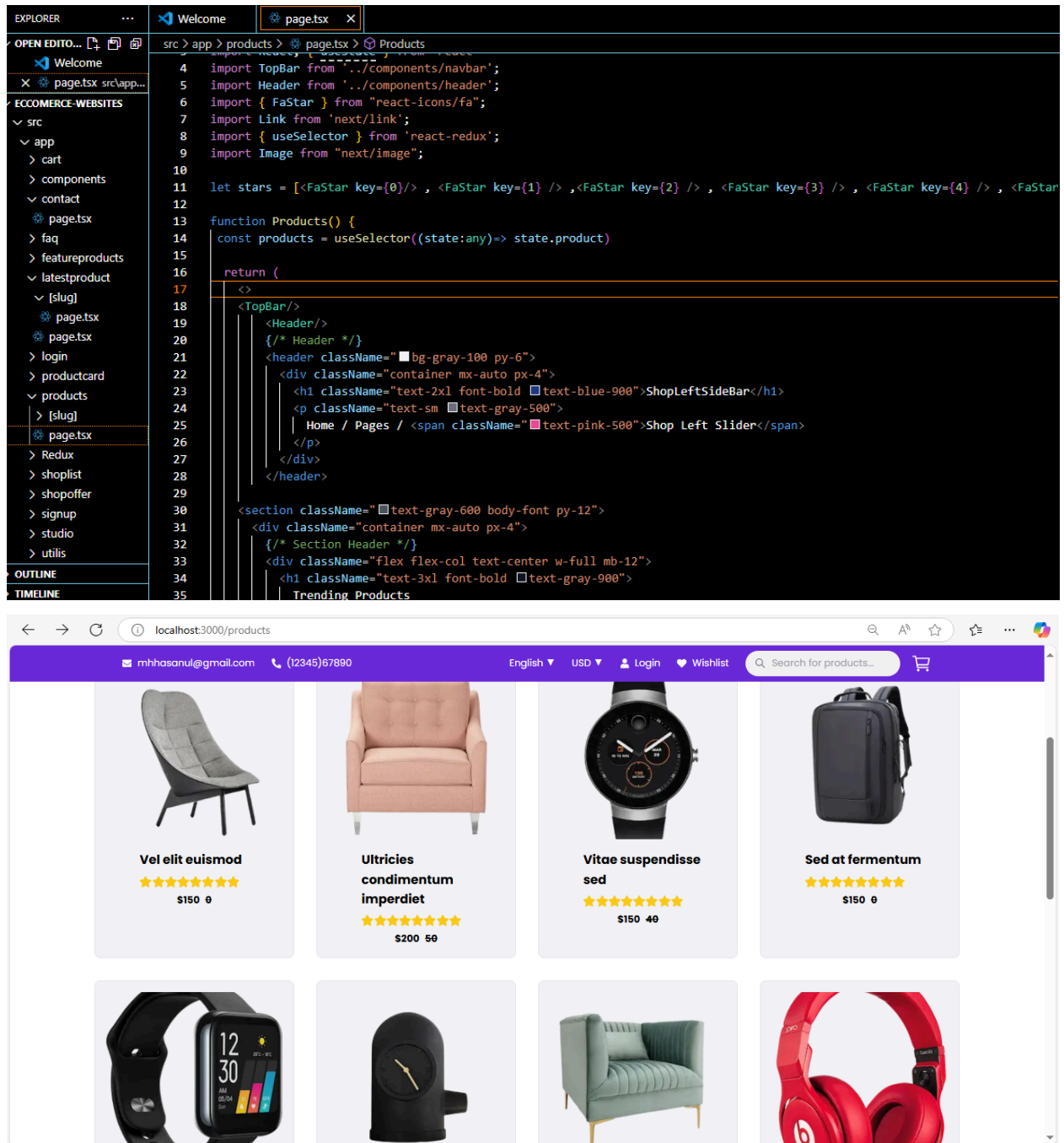
## Overview Eccommerce website

An eCommerce website is an online platform for buying and selling products or services. Its purpose is to provide customers with a convenient shopping experience, accessible anytime and anywhere. It helps businesses expand their market reach, increase sales, and build brand visibility. The platform enables secure transactions, personalized marketing, and efficient operations. Overall, it bridges the gap between businesses and customers in the digital marketplace.



## Products Listing

Browse our collection of must-have items, from fashion and accessories to electronics and home essentials. Find top-rated products at unbeatable prices, all in one convenient place. Start shopping now!



## DynamicRouting Product Page

### Product:1

Dynamic routing for an e-commerce website allows product pages to be generated based on unique identifiers (slugs) in the URL. Each product has a dynamic route, such as `/products/[slug]`, where `[slug]` corresponds to the product's name or ID. This enables the display of individual product details by fetching data from a database or API based on the slug. Dynamic routing enhances SEO, scalability, and user experience by providing clean, shareable URLs and allowing the addition of new products without manual route configuration.



```
15 function SlugPage({ params }: { params: { slug: string } }) {
23   title: items.title,
24   image: items.image,
25   slug: items.slug,
26   price: items.price,
27   size: items.size,
28   color: items.color,
29   description: items.description,
30   qty: items.qty,
31   discount: items.discount,
32
33 })
34
35
36 return (
37   <>
38     <TopBar/>
39     <Header/>
40
41     <div className='mt-24 flex flex-col lg:flex-row justify-around'>
42       <div className='sm:space-y-3 space-x-3 sm:space-x-0 p-2 sm:p-0 order-2 sm:order-1 w-full lg:w-[200px] flex lg:fle
43         <Image src={items.image[0]} className='w-[70%]' alt={items.title} width={100} height={100}></Image>
44         <Image src={items.image[0]} className='w-[70%]' alt={items.title} width={100} height={100}></Image>
45         <Image src={items.image[0]} className='w-[70%]' alt={items.title} width={100} height={100}></Image>
46       </div>
47       { /* mid div */ }
48       <div className='lg:w-[500px] flex justify-center mt-4 sm:mt-0 order-1 sm:order-2'>
49         <Image src={items.image[0]} className='w-[80%]' alt={items.title} width={100} height={100}></Image>
50       </div>
51       { /* all text */ }
52       <div className='lg:w-[500px] space-y-2 p-3 sm:p-0 mt-3 sm:mt-0 order-3'>
53         <h1 className='text-xl font-bold lg:text-3xl'>{items.title}</h1>
```



## Sollicitudin amet orci

★★★★★

1500

Best quality sofas for home and offices.

Select Colors



Select Size



- 1 + Add to cart!

## Cart Page

Product added Successfully

Home

Pages

Products

Blog

Shop

About

Contact

Search...

Q

English


USD

Login

Wishlist


Search for products...

2




Size:S  
Color:#ffffff  
**\$1500**

- 1 +




Size:M  
Color:#ffffff  
**\$160**

- 1 +



Size:S  
Color:#ffffff  
**\$350**

- 1 +



Size:S  
Color:#fafa0a  
**\$100**

- 1 +

**Order Summary**

Subtotal

\$2110

Discount (-20%)

-\$0

Delivery Fee

\$0

**Total**



**\$2110**

Add promo code

Apply

Go to Checkout

## Product:2



### Lectus vulputate faucibus

★★★★★  
150

Best quality headset for women.

Select Colors

☐ ☒ ☐

Select Size

☒ S ☐ M ☐ XL

— 1 + Add to cart!

Product added Successfully

Size:S

Color:#fafa

\$150

— 1 +

English USD Login Wishlist

Search for products...

2

Hekto Home Pages Products Blog Shop About Contact

Search...

#### Order Summary

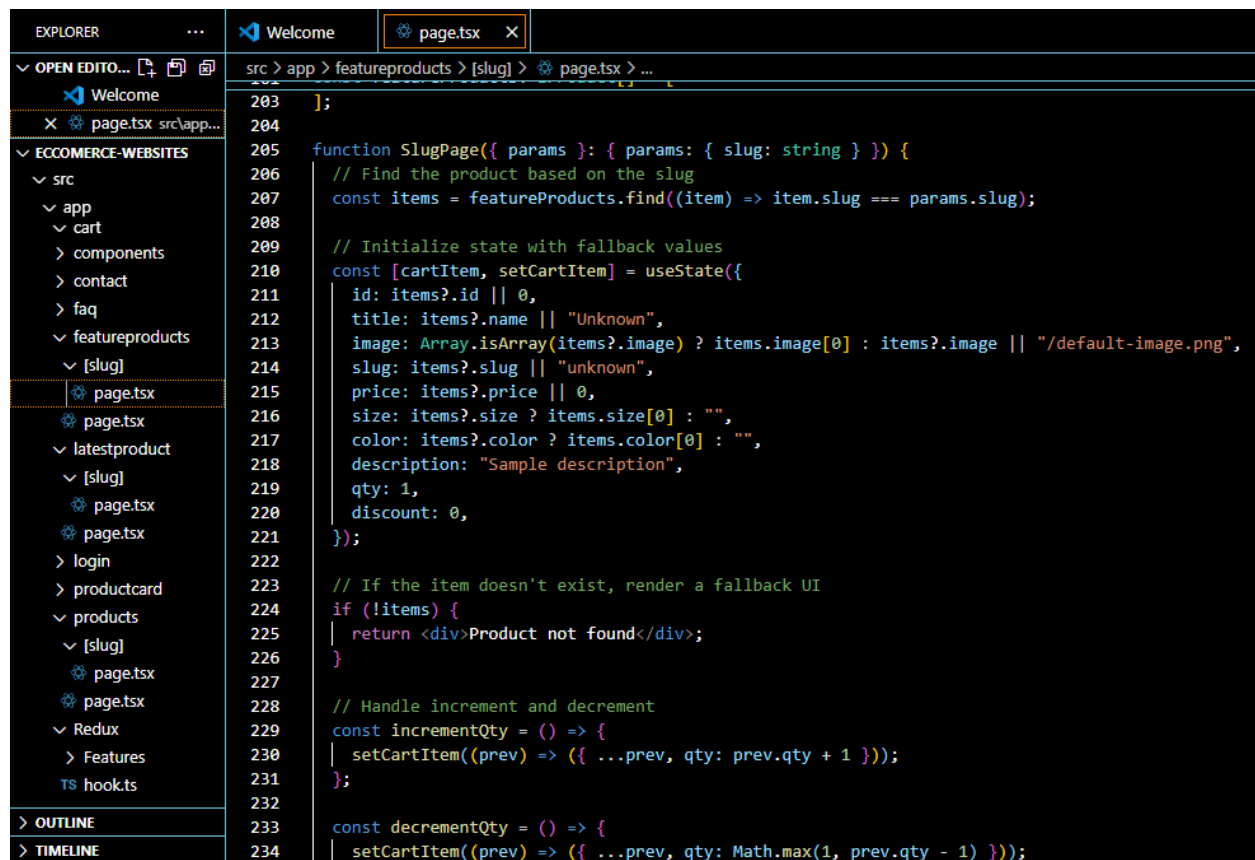
Subtotal	\$150
Discount (-20%)	-\$0
Delivery Fee	\$0
<b>Total</b>	<b>\$150</b>

## Feature Products

Featured products on an e-commerce website are carefully selected items showcased to attract attention and drive sales, typically displayed prominently on the homepage or dedicated sections. These products often include special offers, new arrivals, or bestsellers to engage customers. The feature status of a product is often managed through backend logic, allowing merchants to highlight specific products based on sales performance or seasonal promotions. This helps enhance user experience by making it easier for shoppers to find high-quality or in-demand items.



Dynamic routing for featured products on an e-commerce website allows for the creation of dedicated pages for products marked as featured, accessible through routes like `/featured/[slug]`. Each featured product's route is dynamically generated based on its unique identifier (slug) or category. This method enables the display of curated product lists or specific featured items, ensuring easy updates and scalability. By utilizing dynamic routes, featured products can be easily added, removed, or updated without modifying the overall structure of the site.



The screenshot displays a code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project structure for an e-commerce website, including a `src` directory with `app`, `cart`, `components`, `contact`, `faq`, `featureproducts`, `login`, `productcard`, `products`, `Redux`, and `hooks`. The `featureproducts` directory is expanded, showing a `[slug]` directory with a `page.tsx` file. The code editor shows the implementation of the `SlugPage` function in `page.tsx`. The function takes `params` as an argument and uses `useState` to manage the `cartItem` state. It finds the product based on the slug and renders a fallback UI if the product is not found. It also handles increment and decrement actions for the quantity.

```
203 ];
204
205 function SlugPage({ params }: { params: { slug: string } }) {
206   // Find the product based on the slug
207   const items = featureProducts.find((item) => item.slug === params.slug);
208
209   // Initialize state with fallback values
210   const [cartItem, setCartItem] = useState({
211     id: items?.id || 0,
212     title: items?.name || "Unknown",
213     image: Array.isArray(items?.image) ? items.image[0] : items?.image || "/default-image.png",
214     slug: items?.slug || "unknown",
215     price: items?.price || 0,
216     size: items?.size ? items.size[0] : "",
217     color: items?.color ? items.color[0] : "",
218     description: "Sample description",
219     qty: 1,
220     discount: 0,
221   });
222
223   // If the item doesn't exist, render a fallback UI
224   if (!items) {
225     return <div>Product not found</div>;
226   }
227
228   // Handle increment and decrement
229   const incrementQty = () => {
230     setCartItem((prev) => ({ ...prev, qty: prev.qty + 1 }));
231   };
232
233   const decrementQty = () => {
234     setCartItem((prev) => ({ ...prev, qty: Math.max(1, prev.qty - 1) }));
```



Featured Products

## Cantilever Chair

★★★★☆

Fam locavore kickstarter distillery...

Select Colors



Select Size



- 1 +

Add to cart!

## Latest Product

Latest products on an e-commerce website are newly added items that have recently been stocked or launched, often displayed to keep customers updated with fresh offerings. These products are typically organized in a "New Arrivals" section, allowing shoppers to easily browse the newest items. Highlighting the latest products helps increase customer engagement and drives curiosity about new trends or innovations. By promoting these products, e-commerce platforms aim to attract repeat visitors and boost sales through novelty and exclusivity.



```
EXPLORER  ...  Welcome  page.tsx  X
OPEN EDITO...  src > app > latestproduct > page.tsx > ...
Welcome
page.tsx  src/app...
ECCOMERCE-WEBSITES
  src
  app
  cart
  components
  contact
  faq
  featureproducts
    [slug]
    page.tsx
    page.tsx
  latestproduct
    [slug]
    page.tsx
    page.tsx
  login
  productcard
  products
    [slug]
    page.tsx
    page.tsx
  Redux
  Features
  TS hooks
OUTLINE
SUMMARY

133 function LatestProduct() {
136   useEffect(() => {
137     async function fetchProduct() {
141       } catch (error) {
142         console.error('Failed to fetch products:', error);
143       }
144     }
145     fetchProduct();
146   }, []);
147
148   return (
149     <section className="text-gray-600 body-font py-12">
150       <div className="container mx-auto">
151         <div className="flex flex-col text-center w-full mb-8">
152           <h1 className="text-3xl font-bold text-gray-900">Latest Products</h1>
153         </div>
154
155         <div className="grid grid-cols-1 sm:grid-cols-2 lg:grid-cols-3 gap-8 mb-8">
156           {product.slice(0, 9).map((product) => (
157             <div
158               key={product.id}
159               className="p-4 border border-gray-200 rounded-lg hover:shadow-lg transition-shadow duration-300"
160             >
161               <div className="flex flex-col items-center text-center">
162                 <Link href={`/${latestproduct}/${product.slug.current}`}>
163                   {product.image ? (
164                     <Image
165                       src={urlFor(product.image).url()}
166                       alt={product.name}
167                       className="rounded-lg w-full h-80 object-cover bg-gray-100 mb-4"
168                     />
169                   ) : null}
170                 <div>
```



Armchair Tortuga  
\$850 0



Stylish Golden Metal Legs Mint Blue  
Fabric Velvet Sofa Leisure Armchair  
\$780 0



Rapson Thirty-Nine Guest Chair  
\$1300 ~~\$105.00~~

## Dynamic Page of latest Product:

```
EXPLORER  ...  Welcome  page.tsx ...latestproduct  page.tsx ...[slug] X
OPEN EDITO...  src > app > latestproduct > [slug] > page.tsx > ...
Welcome
page.tsx src/app...
page.tsx src/app...
ECCOMERCE-WEBSITES
src
app
cart
components
contact
faq
featureproducts
  [slug]
    page.tsx
    page.tsx
  latestproduct
    [slug]
      page.tsx
      page.tsx
  login
  productcard
  products
    [slug]
      page.tsx
      page.tsx
  Redux
  Features
OUTLINE
TIMELINE

61  const ProductDetails: React.FC<{ product: Product }> = ({ product }) => {
99    <div className="text-gray-900 text-5xl title-font font-medium mb-1">
100      {product.name}
101    </div>
102    <p className="leading-relaxed">{product.description}</p>
103
104    /* Color Selection */
105    {product.color && product.color.length > 0 && (
106      <div className="mt-6">
107        <p className="text-gray-400">Select Color</p>
108        <div className="flex space-x-3">
109          {product.color.map((color, i) => (
110            <button
111              key={i}
112              onClick={() => setCartItem({ ...cartItem, color })}
113              className={`w-10 h-10 rounded-full border-2 ${
114                cartItem.color === color ? "border-black" : "border-gray-300"
115              }`}
116              style={{ backgroundColor: color }}
117            </button>
118          ))}
119        </div>
120      </div>
121    )}
122
123    /* Size Selection */
124    {product.size && product.size.length > 0 && (
125      <div className="mt-6">
126        <p className="text-gray-400">Select Size</p>
127        <div className="flex space-x-3">
128          {product.size.map((size, i) => (
129            <button
```

#### Latest Products

### Stylish Golden Metal Legs Mint Blue Fabric Velvet Sofa Leisure Armchair

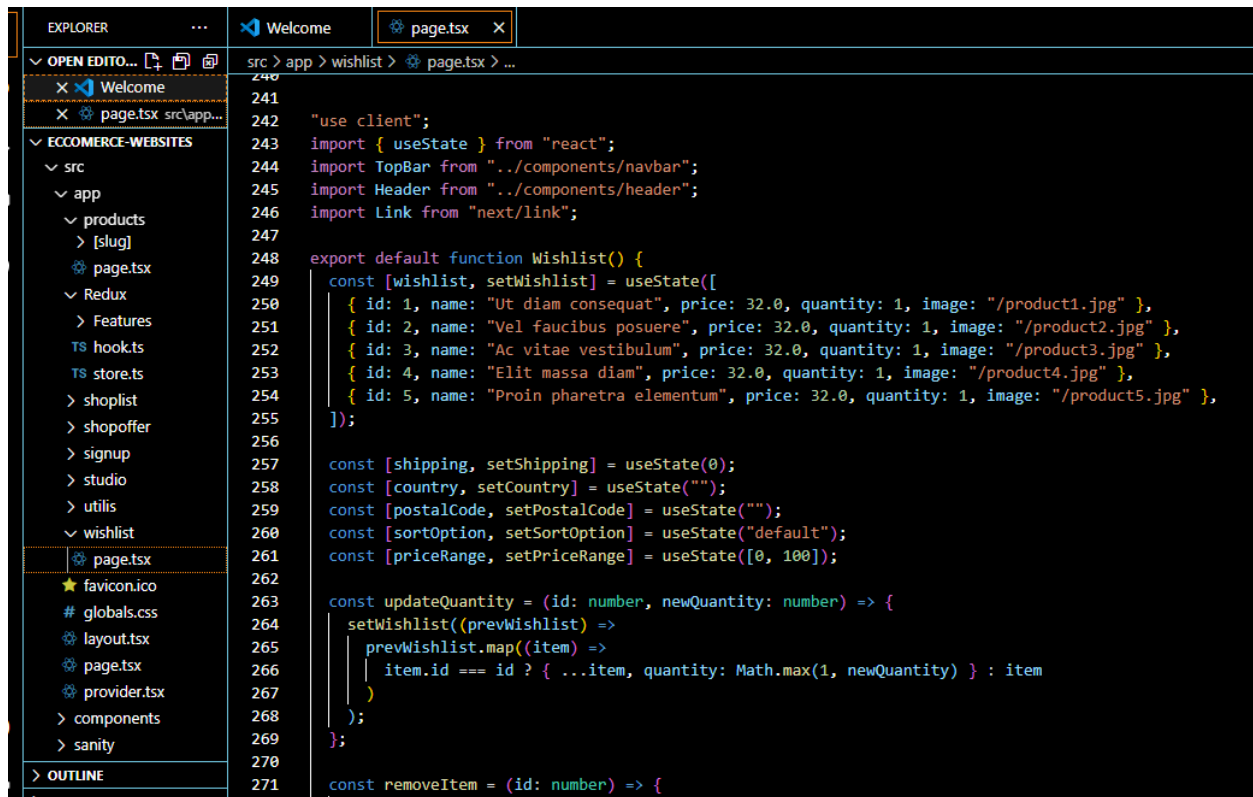
A luxurious armchair with velvet fabric and golden metal legs.

**\$780.00**


- 1 +

Add to cart!

# Wish List page



```
240
241
242 "use client";
243 import { useState } from "react";
244 import TopBar from "../components/navbar";
245 import Header from "../components/header";
246 import Link from "next/link";
247
248 export default function Wishlist() {
249   const [wishlist, setWishlist] = useState([
250     { id: 1, name: "Ut diam consequat", price: 32.0, quantity: 1, image: "/product1.jpg" },
251     { id: 2, name: "Vel faucibus posuere", price: 32.0, quantity: 1, image: "/product2.jpg" },
252     { id: 3, name: "Ac vitae vestibulum", price: 32.0, quantity: 1, image: "/product3.jpg" },
253     { id: 4, name: "Elit massa diam", price: 32.0, quantity: 1, image: "/product4.jpg" },
254     { id: 5, name: "Proin pharetra elementum", price: 32.0, quantity: 1, image: "/product5.jpg" },
255   ]);
256
257   const [shipping, setShipping] = useState(0);
258   const [country, setCountry] = useState("");
259   const [postalCode, setPostalCode] = useState("");
260   const [sortOption, setSortOption] = useState("default");
261   const [priceRange, setPriceRange] = useState([0, 100]);
262
263   const updateQuantity = (id: number, newQuantity: number) => {
264     setWishlist((prevWishlist) =>
265       prevWishlist.map((item) =>
266         item.id === id ? { ...item, quantity: Math.max(1, newQuantity) } : item
267       )
268     );
269   };
270
271   const removeItem = (id: number) => {
```



Ut diam consequat

Price: \$32


-

1

+

Total: \$32

Remove



Vel faucibus posuere

Price: \$32


-

1

+

Total: \$32

Remove



Ac vitae vestibulum

Price: \$32


-

1

+

Total: \$32

Remove



Elit massa diam

Price: \$32


-

1

+

Total: \$32

Remove



Proin pharetra elementum

Price: \$32

-

1

+

Total: \$32

Remove

Cart Totals

Subtotal: \$160.00

Shipping: \$0

Total: \$160.00

Proceed to Checkout

Calculate Shipping

Country

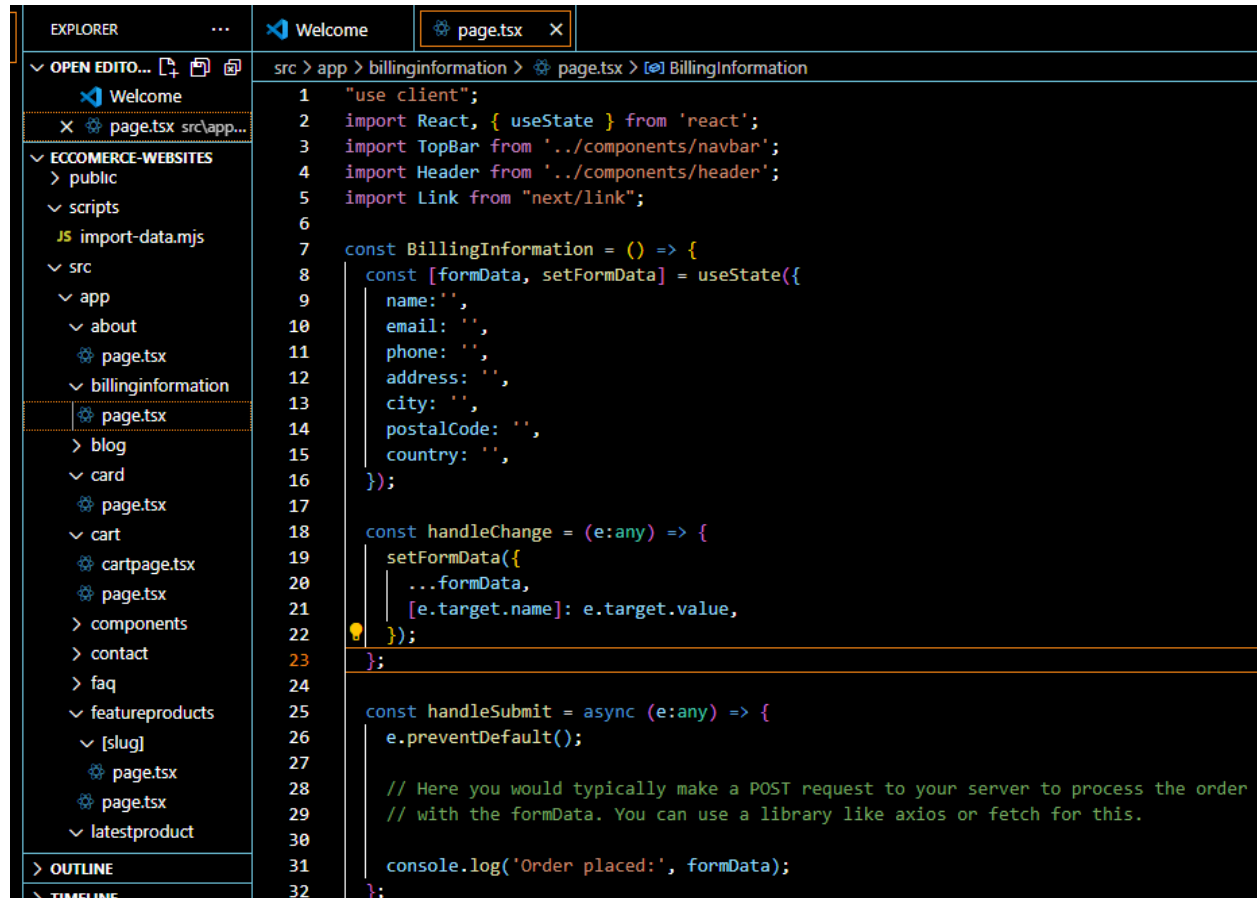
Enter country

Postal Code

Enter postal code

Calculate Shipping

# Billing Information



The image shows a VS Code editor interface. On the left is the Explorer sidebar showing a project structure for 'ECCOMERCE-WEBSITES'. The main editor area displays the 'page.tsx' file for the 'BillingInformation' component. The code uses 'use client' and 'useState' to manage form data for a billing form. It includes fields for name, email, phone, address, city, postal code, and country. There are two functions: 'handleChange' for updating form fields and 'handleSubmit' for processing the form submission, which includes a comment about making a POST request to a server.

```
1  "use client";
2  import React, { useState } from 'react';
3  import TopBar from '../components/navbar';
4  import Header from '../components/header';
5  import Link from "next/link";
6
7  const BillingInformation = () => {
8    const [formData, setFormData] = useState({
9      name: '',
10     email: '',
11     phone: '',
12     address: '',
13     city: '',
14     postalCode: '',
15     country: '',
16   });
17
18   const handleChange = (e:any) => {
19     setFormData({
20       ...formData,
21       [e.target.name]: e.target.value,
22     });
23   };
24
25   const handleSubmit = async (e:any) => {
26     e.preventDefault();
27
28     // Here you would typically make a POST request to your server to process the order
29     // with the formData. You can use a library like axios or fetch for this.
30
31     console.log('Order placed:', formData);
32   };
33 }
```

## Billing Information

Name

Email

Phone

Address

City

Postal Code

Country

## Order Completed



The screenshot shows a code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project structure for 'ECCOMERCE-WEBSITES' with a 'src' directory containing 'app', 'card', and 'page.tsx'. The code editor shows the 'Order' function in 'page.tsx'.

```
7 export default function Order() {  
19  
20  
21  
22   /* Main Content Section */  
23   <div className="flex flex-col items-center mt-16 space-y-6">  
24     /* Tick Image */  
25     <Image  
26       src="/tick.png"  
27       alt="Order Completed Tick"  
28       width={120}  
29       height={80}  
30       className="rounded-full object-cover"  
31     />  
32  
33     /* Description */  
34     <div className="text-center px-4">  
35       <h2 className="text-[#151875] font-bold text-2xl">  
36         Your Order Is Completed  
37       </h2>  
38       <p className="mt-4 text-[#8D92A7] text-sm leading-relaxed">  
39         Thank you for your order! Your order is being processed and will be  
40         completed within 3-6 hours. You will receive an email confirmation when  
41         your order is completed.  
42       </p>  
43     </div>  
44  
45     /* Continue Shopping Button */  
46     <Link href="/shoplist">  
47       <button className="mt-6 px-6 py-3 bg-[#FF3EB2] text-white text-sm font-semibol  
48     </button>  
49     </Link>  
50   </div>  
51 }  
52
```

# Order Completed

Home . Pages . [Order Completed](#)



## Your Order Is Completed

er is being processed and will be completed within 3-6 hours. You will receive an email confirmation when your

[Continue Shopping](#)



## About

```
EXPLORER  Welcome  page.tsx X
src > app > about > page.tsx > AboutUs
4 import Header from "../components/header";
5
6 const AboutUs: React.FC = () => {
7   return (
8     <div className="bg-gray-50 text-gray-800">
9       <TopBar />
10      <Header />
11
12      { /* Header Section */ }
13      <header className="text-center py-8 bg-gray-100">
14        <h1 className="text-4xl font-bold">About Us</h1>
15        <p className="text-sm text-gray-600 mt-2">Home /Pages/ About Us</p>
16      </header>
17
18      { /* About Section */ }
19      <section className="text-gray-700 body-font">
20        <div className="container mx-auto flex flex-col-reverse md:flex-row px-5 py-24 items-center">
21          <div className="lg:max-w-lg lg:w-full md:w-1/2 w-5/6 mb-10 md:mb-0">
22            <Image
23              className="object-cover object-center rounded-lg mt-5"
24              alt="hero"
25              src="/boys.jpg"
26              width={400}
27              height={300}
28            />
29          </div>
30          <div className="lg:flex-grow md:w-1/2 lg:pl-24 md:pl-16 flex flex-col items-center md:items-start">
31            <h1 className="title-font sm:text-4xl text-3xl mb-4 font-medium text-gray-900">
32              Know About Our E-commerce Business & History
33            </h1>
34            <p className="mb-8 leading-relaxed">
```



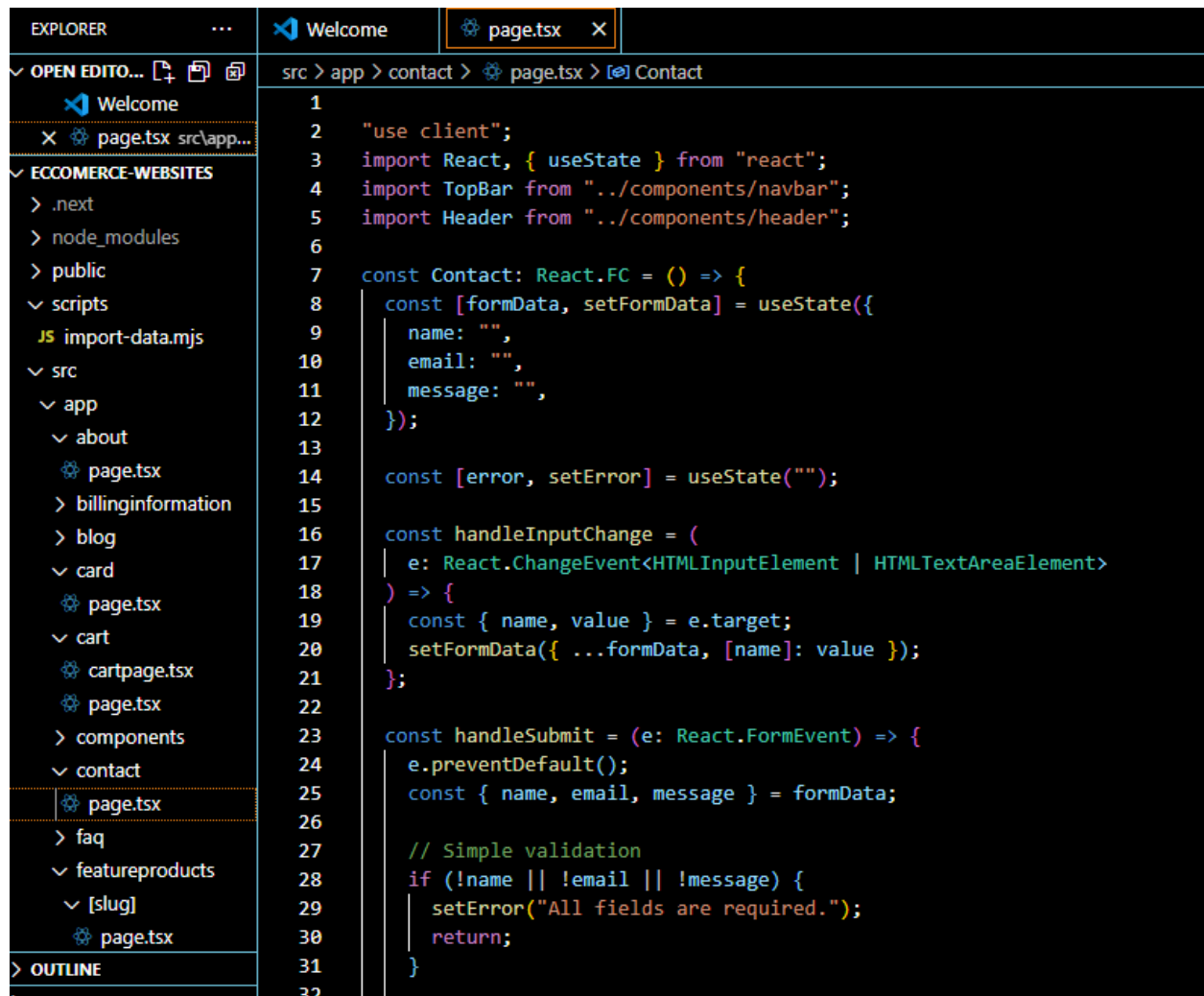
## Know About Our E-commerce Business & History

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Non duis ultr vel dui sollicitudin aliquet id arcu. Nam vitae a enim nunc, sed sapien ac nam. Tristique ultrices dolor aliquam lacus volutpat praesent.

Contact Us



## Contact us



```
1
2 "use client";
3 import React, { useState } from "react";
4 import TopBar from "../components/navbar";
5 import Header from "../components/header";
6
7 const Contact: React.FC = () => {
8   const [formData, setFormData] = useState({
9     name: "",
10    email: "",
11    message: "",
12  });
13
14   const [error, setError] = useState("");
15
16   const handleInputChange = (
17     e: React.ChangeEvent<HTMLInputElement | HTMLTextAreaElement>
18   ) => {
19     const { name, value } = e.target;
20     setFormData({ ...formData, [name]: value });
21   };
22
23   const handleSubmit = (e: React.FormEvent) => {
24     e.preventDefault();
25     const { name, email, message } = formData;
26
27     // Simple validation
28     if (!name || !email || !message) {
29       setError("All fields are required.");
30       return;
31     }
32   }
33 }
```

20 Margaret St, LC  
Great Britain, 3NM

## An illustration of two people, a woman with purple hair and a man with blue hair, working together at a computer. The woman is on the left, wearing a yellow shirt and holding a blue phone to her ear. The man is on the right, wearing a red shirt and a headset, typing on a keyboard. A large blue monitor is between them. The background is light blue with several circular icons: a calendar, a clock showing 24/7, a mail envelope, a gear for settings, and a speech bubble. The entire scene is framed by concentric circles and small blue dots, suggesting a digital or network environment.

Send Message

The screenshot displays a code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with folders like 'src', 'app', 'card', 'pagetx', 'cart', 'components', 'contact', 'faq', 'featureproducts', 'slug', 'latestproduct', and 'products'. The code editor shows the content of 'src > app > productcard > page.tsx', which is a React component for a product card. The code includes a const ProductCard definition, a header with product details, a main image area with three placeholder images, and a section for a product description and price.

## Product Details

[Home](#) / [Pages](#) / [Product Details](#)



### Playwood Arm Chair

★★★★★ (22)

A stylish and comfortable armchair for modern interiors.

**\$32.00** ~~\$40.00~~

[Add to Cart](#)

**Categories**

**Tags**

**Share**



## Login page

### Login

Please login using your account details below.

Email Address

saima

Password

.....

[Sign In](#)

Don't have an account? [Sign up](#)

The screenshot shows a VS Code editor with a file explorer on the left and a code editor on the right. The file explorer, titled 'ECCOMERCE-WEBSITES', shows a project structure with folders like 'src', 'app', 'card', 'cart', 'components', 'contact', 'faq', 'featureproducts', 'latestproduct', 'login', 'productcard', and 'products'. The 'login' folder is expanded, showing 'page.tsx' selected. The code editor shows the content of 'page.tsx' with the following code:

```
src > app > login > page.tsx > ...
6  import Header from "../components/header";
7
8  function Login() {
9      const [email, setEmail] = useState("");
10     const [password, setPassword] = useState("");
11     const [error, setError] = useState("");
12     const [success, setSuccess] = useState("");
13
14     const handleSubmit = async (e: any) => {
15         e.preventDefault();
16
17         // Validation
18         if (!email || !password) {
19             setError("Please fill in all fields.");
20             setSuccess("");
21             return;
22         }
23
24         try {
25             // Simulated API call for login
26             const response = await fetch("/api/login", {
27                 method: "POST",
28                 headers: { "Content-Type": "application/json" },
29                 body: JSON.stringify({ email, password }),
30             });
31
32             if (!response.ok) {
33                 throw new Error("Invalid email or password. Please try again.");
34             }
35
36             const data = await response.json();
37             setSuccess("Login successful! Redirecting...");
```

## SignUp Page

EXPLORER	...	Welcome	page.tsx
OPEN EDITOR...		src > app > signup > page.tsx > ...	
Welcome		8	function SignUp() {
page.tsx	src/app...	16	const handleSubmit = async (e: any) => {
ECCOMERCE-WEBSITES		21	setError("Please fill in all fields.");
src		22	setSuccess("");
app		23	return;
featureproducts		24	}
latestproduct		25	
[slug]		26	if (password !== confirmPassword) {
page.tsx		27	setError("Passwords do not match.");
login		28	setSuccess("");
productcard		29	return;
page.tsx		30	}
products		31	
[slug]		32	try {
page.tsx	M	33	// Simulated API call for signup
page.tsx		34	const response = await fetch("/api/signup", {
Redux		35	method: "POST",
Features		36	headers: { "Content-Type": "application/json" },
hook.ts		37	body: JSON.stringify({ name, email, password }),
store.ts		38	});
shoplist		39	
shopoffer		40	if (!response.ok) {
signup		41	throw new Error("Failed to sign up. Please try again.");
page.tsx		42	}
studio		43	
utilis		44	const data = await response.json();
OUTLINE		45	setSuccess("Account created successfully! Please log in.");
TIMELINE		46	setError("");
		47	
		48	// Clear form fields
		49	setName("");
		50	setEmail("");
		51	setPassword("");

**Create an Account**

Please sign up by filling in your details below.

Full Name

Email Address

Password

Confirm Password

**Sign Up**

Already have an account? [Log in](#)

### **Conclusion of website:**

**Ecommerce website is a powerful tool for businesses to connect with a global audience, streamline operations, and drive growth. It offers customers convenience, variety, and a seamless shopping experience. By leveraging data insights and technology, businesses can deliver personalized services and build lasting relationships. An eCommerce platform also helps reduce costs while maximizing reach and efficiency. Ultimately, it plays a vital role in shaping the future of commerce in the digital era.**

