


Hackathon 3 day 4

The BestOfAll component is a dynamic product showcase page built with Next.js, Sanity CMS, and Tailwind CSS. It fetches product data from the backend and displays it in a responsive grid. Each product card includes an image, name, category, price, and availability status. Users can click on a product to view details or add it to their cart with a single button. A smooth notification pops up confirming the action using SweetAlert2. The page is interactive, visually appealing, and designed for a seamless shopping experience.


Best Products



Nike Court Vision Low Next Nature
Men's Shoes

\$4995


Just In [Add to Cart](#)



Nike Standard Issue Basketball Jersey
Women's Basketball Jersey

\$2895


Just In [Add to Cart](#)




Nike Dunk Low Retro SE
Men's Shoes

\$9695


Promo Exclusion [Add to Cart](#)



Nike Dri-FIT UV Hyverse
Men's Short-Sleeve Graphic Fitness Top

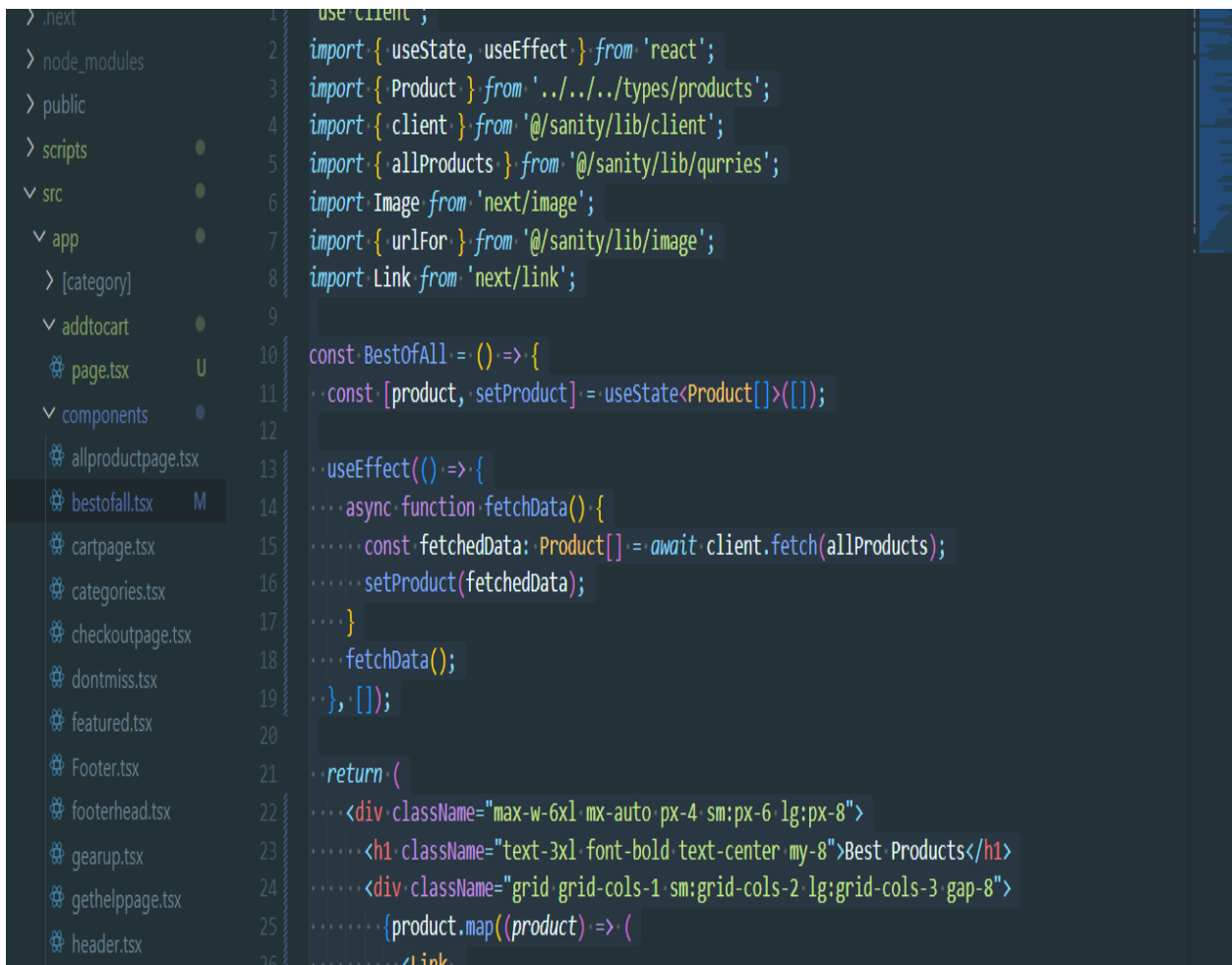


Nike Court Vision Low
Men's Shoes



Nike Pegasus 40
Men's Running Shoes

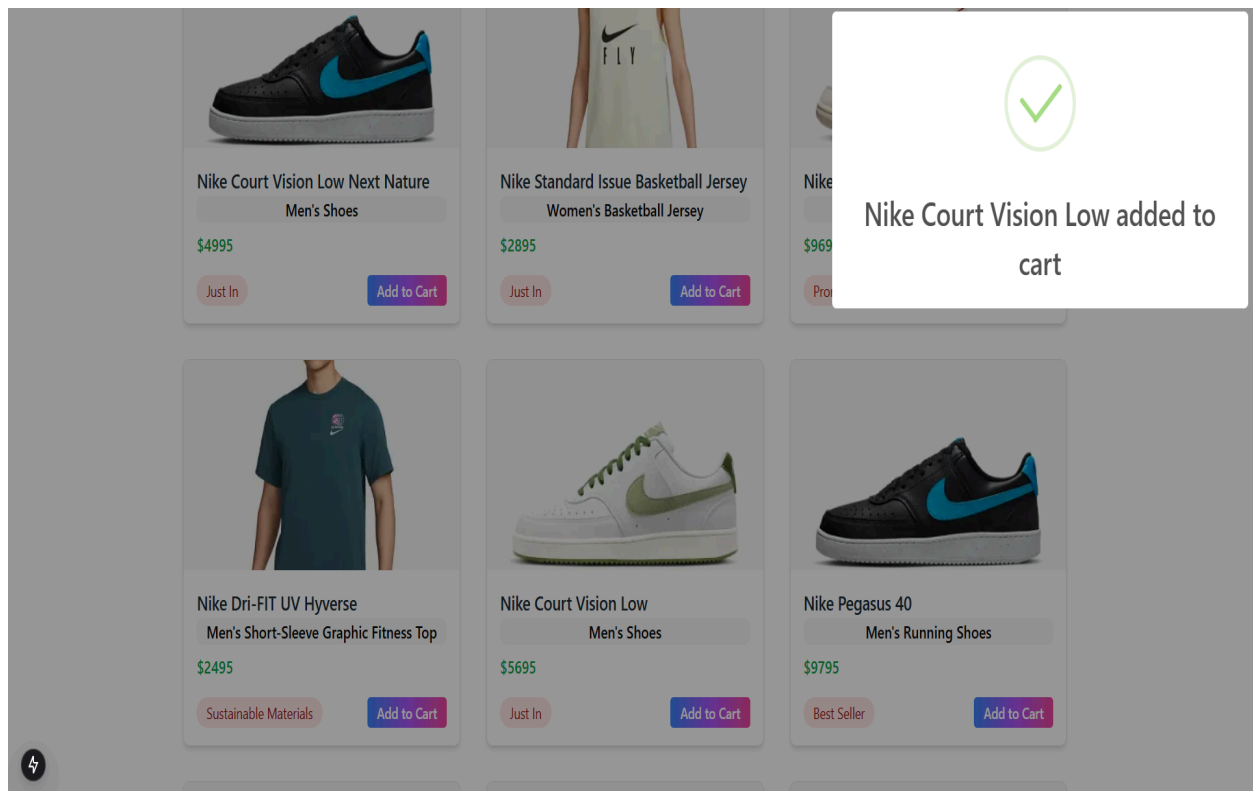
The code snippet demonstrates a React component that fetches a collection of products from a Content Management System (CMS) using the Sanity client and displays them on a web page.



```
1 use Client ;
2 import { useState, useEffect } from 'react';
3 import { Product } from '../types/products';
4 import { client } from '@sanity/lib/client';
5 import { allProducts } from '@sanity/lib/queries';
6 import Image from 'next/image';
7 import { urlFor } from '@sanity/lib/image';
8 import Link from 'next/link';
9
10 const BestOfAll = () => {
11   const [product, setProduct] = useState<Product[]>([]);
12
13   useEffect(() => {
14     async function fetchData() {
15       const fetchedData: Product[] = await client.fetch(allProducts);
16       setProduct(fetchedData);
17     }
18     fetchData();
19   }, []);
20
21   return (
22     <div className="max-w-6xl mx-auto px-4 sm:px-6 lg:px-8">
23       <h1 className="text-3xl font-bold text-center my-8">Best Products</h1>
24       <div className="grid grid-cols-1 sm:grid-cols-2 lg:grid-cols-3 gap-8">
25         {product.map((product) => (
26           <Link
```

Add To Cart Functionality

An overlay notification appears after clicking "Add to Cart," confirming that the item has been successfully added. This alert features the product name (e.g., "Nike Court Vision Low added to




cart") and a green checkmark, indicating success.

1. Image: A clear product image for easy identification.
2. Name: The product title (e.g., "Nike Court Vision Low Next Nature").
3. Category: A brief description of the product type (e.g., "Men's Shoes").
4. Price: The product price in a clear, prominent format (e.g., "\$4995").
5. Tags: Visual labels such as "Just In" or "Sustainable Materials" for product categorization.
6. Add to Cart: A button to add the product to the cart.

```
> .next 1 import exp from "constants";
> node_modules 2 import { Product } from "../../types/products";
> public 3 import { get } from "http";
> scripts 4
> src 5
  > app 6 export const addToCart = (product: Product) => {
    > [category] 7 ... let cart : Product[] = JSON.parse(localStorage.getItem('cart') || '[]');
    > actions 8 ... const existingProductIndex = cart.findIndex(item => item._id === product._id);
    > actions.tsx U 9
    > addtocart 10
    > components 11 ... if(existingProductIndex > -1){
    > allproductpage.tsx 12 ... cart[existingProductIndex].inventory += 1;
    > bestofall.tsx M 13
    > cartpage.tsx 14 ... } else {
    > categories.tsx 15 ... cart.push({...product, inventory: 1});
    > checkoutpage.tsx 16 ... }
    > dontmiss.tsx 17 ... localStorage.setItem('cart', JSON.stringify(cart));
    > featured.tsx 18 ... }
    > Footer.tsx 19
    > footerhead.tsx 20 ... export const removeFromCart = (productId: string) => {
    > gearup.tsx 21 ... let cart : Product[] = JSON.parse(localStorage.getItem('cart') || '[]');
    22 ... cart = cart.filter(item => item._id !== productId);
    23 ... localStorage.setItem('cart', JSON.stringify(cart));
    24 ...
    25
```

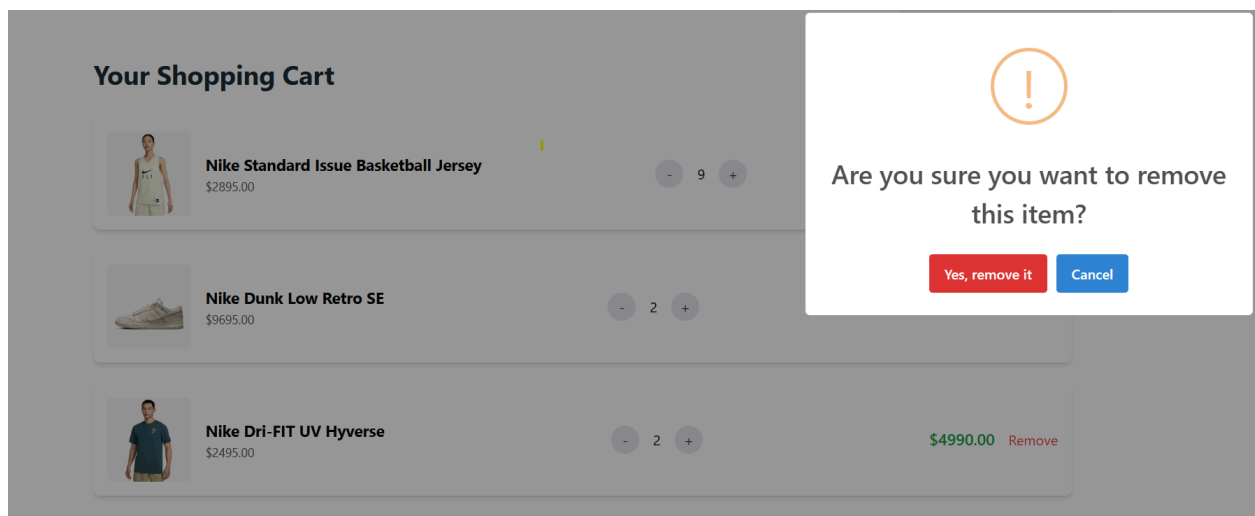
This is a shopping cart UI displaying selected products with their names, prices, quantities, total prices, and options to adjust quantities or remove items

Your Shopping Cart




	<p>Nike Standard Issue Basketball Jersey</p> <p>\$2895.00</p>	<div>- 9 +</div>	<p>\$26055.00 Remove</p>
	<p>Nike Dunk Low Retro SE</p> <p>\$9695.00</p>	<div>- 2 +</div>	<p>\$19390.00 Remove</p>
	<p>Nike Dri-FIT UV Hyverse</p> <p>\$2495.00</p>	<div>- 2 +</div>	<p>\$4990.00 Remove</p>


This **CartPage** component displays a shopping cart interface where users can view items, adjust quantities, remove products, and proceed to checkout. It uses React hooks, SweetAlert for confirmations, and Tailwind CSS for styling.

```
ECOMMERCE-WEBSITE  src > app > addtocart > page.tsx > ...
> .next
> node_modules
> public
> scripts
  src
    app
      [category]
      actions
        actions.tsx  U
      addtocart
        page.tsx  U
      components
        allproductpage.tsx
        bestofall.tsx  M
        cartpage.tsx
        categories.tsx
        checkoutpage.tsx
        dontmiss.tsx
        featured.tsx
        Footer.tsx
        footerhead.tsx
        gearup.tsx
        gethelppage.tsx
5  import Swal from 'sweetalert2';
6  import Image from 'next/image';
7  import { urlFor } from '@sanity/lib/image';
8
9  const CartPage = () => {
10    const [cart, setCart] = useState<Product[]>([]);
11
12    useEffect(() => {
13      setCart(getCartItems());
14    }, []);
15
16    const handleRemove = (id: string) => {
17      Swal.fire({
18        position: 'top-end',
19        icon: 'warning',
20        title: 'Are you sure you want to remove this item?',
21        showCancelButton: true,
22        confirmButtonText: 'Yes, remove it',
23        cancelButtonText: 'Cancel',
24        confirmButtonColor: '#d3d3',
25        cancelButtonColor: '#3085d6',
26      }).then((result) => {
27        if (result.isConfirmed) {
28          removeFromCart(id);
29          setCart(getCartItems());
30          Swal.fire('Item removed from cart', '', 'success');
```






Your Shopping Cart


	Nike Dunk Low Retro SE \$9695.00	- 2 +	\$19390.00 Remove
	Nike Dri-FIT UV Hyver \$2495.00		\$4990.00 Remove
	Nike Court Vision Low \$5695.00		\$11390.00 Remove



Item removed from cart

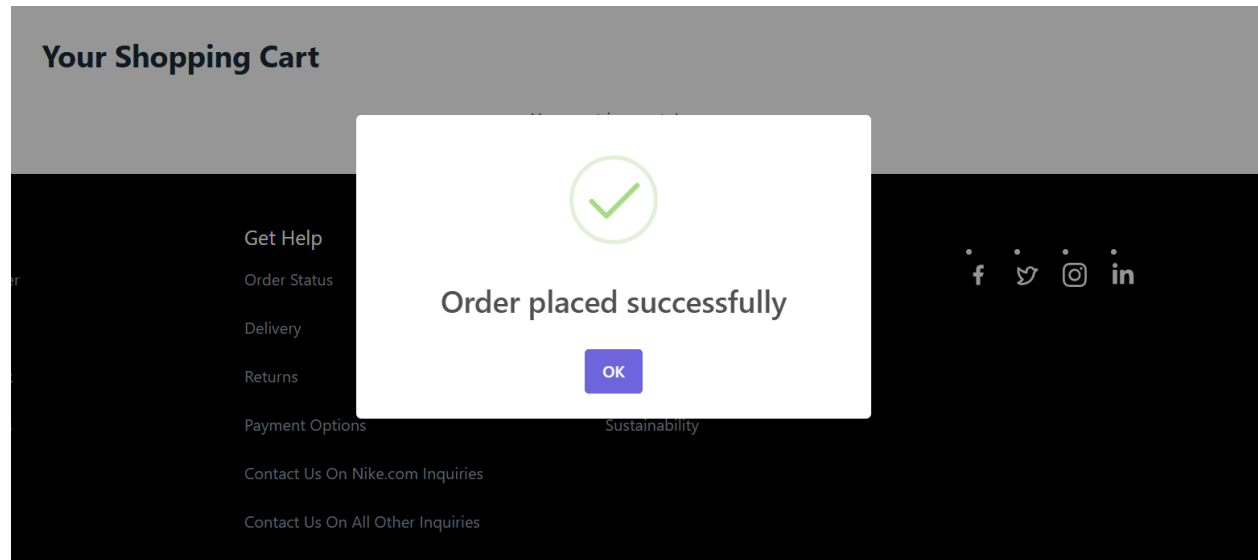
[OK](#)

	Nike Court Vision Low \$5695.00	- 2 +	
	Nike Pegasus 40 \$9795.00	- 1 +	
	Nike Renew Run 3 \$7295.00	- 1 +	\$7295.00 Remove
Total:			\$81945.00
Proceed to Checkout			



Proceed to checkout?

[Yes, proceed](#) [Cancel](#)



By **SAiF SOOMRO**

Class - Sunday 2pm to 5 pm

classTeacher - ALI JAWWAD