Hekto   [What are you looking for?] 🔍   Home   Pages   Blog   Shop   Contact   🛒 Cart   ♡ Wishlist   ☰

Your Shopping Cart

Varmora Plastic Chair Solid
Price: $100

| - | + | | Remove |

Total: **$NaN**

Proceed to Checkout

Shop List

Ecommerce Accessories & Fashion Items

About 9,620 results (0.62 seconds)

Per Page: [   ]   Sort By: Best Match ▾   View: [   ]

Tribù Elio Chair

1200

A sleek outdoor chair with natural wooden elements and a modern look.

★★★★★

♡ 🛒 👁

Stylish Golden Metal Legs Mint Blue Fabric Velvet Sofa Leisure Armchair

780

A luxurious armchair with velvet fabric and golden metal legs.

★★★★★

♡ 🛒 👁

Nautilus Lounge Chair

1450

A lounge chair with a shell-inspired design and premium upholstery.

★★★★★

♡ 🛒 👁

Alpha Chair – Solid Ebonised Oak

900

A sturdy oak chair with a sleek and minimalist design.

★★★★★

♡ 🛒 👁

Futuristic Sleek Modern Chair

2000

A modern chair with a carbon fiber frame and bold red accents.

★★★★★

♡ 🛒 👁

Cozy Armchair
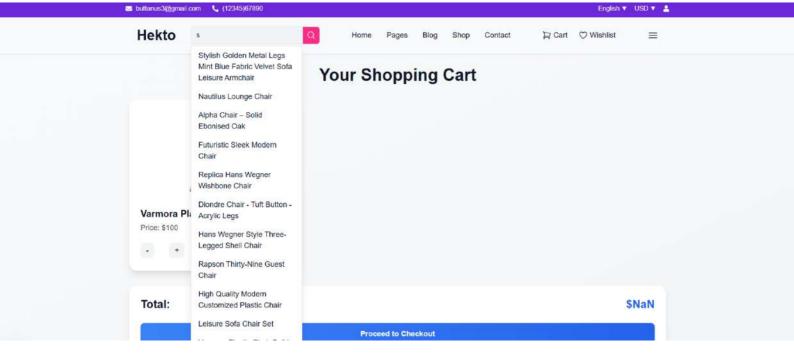
520

Compact armchair with a cozy design for small spaces.

★★★★★

♡ 🛒 👁

Hekto

s   🔍

Home   Pages   Blog   Shop   Contact   🛒 Cart   ♡ Wishlist   ☰

Stylish Golden Metal Legs
Mint Blue Fabric Velvet Sofa
Leisure Armchair

Nautilus Lounge Chair

Alpha Chair – Solid
Ebonised Oak

Futuristic Sleek Modern
Chair

Replica Hans Wegner
Wishbone Chair

Diondre Chair - Tuft Button -
Acrylic Legs

Hans Wegner Style Three-
Legged Shell Chair

Rapson Thirty-Nine Guest
Chair

High Quality Modern
Customized Plastic Chair

Leisure Sofa Chair Set

Your Shopping Cart

Varmora Pla
Price: $100

-    +

Total: **$NaN**

Proceed to Checkout

src > app > shoplist > page.tsx > ...

```tsx
1   "use client";
2   import Image from "next/image";
3   import Link from "next/link";
4   import swal from "sweetalert2";
5   import {
6     AiOutlineHeart,
7     AiOutlineShoppingCart,
8     AiOutlineEye,
9   } from "react-icons/ai";
10  import Header from "../components/Header";
11  import Footer from "../components/Footer";
12  import { client } from "@/sanity/lib/client";
13  import { allProducts } from "@/sanity/lib/queries";
14  import { useState, useEffect } from "react";
15  import { Product } from "../../../types/products";
16  import { addToCart } from "../actions/actions";
17  import { urlFor } from "@/sanity/lib/image";
18
19  const ShopListProducts = () => {
20    const [products, setProducts] = useState<Product[]>([]);
21
22    useEffect(() => {
23      async function fetchProduct() {
24        try {
25          const fetchedProducts: Product[] = await client.fetch(allProducts);
26          setProducts(fetchedProducts);
27        } catch (error) {
28          console.error("Error fetching products:", error);
29        }
```

```tsx
// components/AddToCartButton.tsx
'use client'

import Swal from "sweetalert2";
import { Product } from "../../../types/products";
import { addToCart } from "@/app/actions/actions";

interface AddToCartButtonProps {
  product: Product;
}

const AddToCartButton: React.FC<AddToCartButtonProps> = ({ product }) => {
  const handleAddToCart = (e: React.MouseEvent) => {
    e.preventDefault();
    Swal.fire({
      position: "top-end",
      icon: "success",
      title: `${product.name} added to cart`,
      showConfirmButton: false,
      timer: 1000,
    });
    addToCart(product);
  };

  return (
    <button
      onClick={handleAddToCart}
      className="w-full py-3 bg-blue-600 text-white font-semibold rounded-lg hover:bg-blue-700 transition-col
    >
```

```tsx
import React, { useState, useEffect } from "react";
import { MagnifyingGlassIcon } from "@heroicons/react/24/outline";
import Link from "next/link";
import { client } from "@/sanity/lib/client"; // Adjust the import based on your setup
import { allProducts } from "@/sanity/lib/queries"; // Import your predefined query

// Define product type
interface Product {
  _id: string;
  name: string;  // Use 'name' instead of 'title'
  slug: { current: string }; // Assuming your Sanity schema includes a slug
}

const SearchBar: React.FC = ' ' > '
  const [searchTerm, setSea        const setFilteredProducts: React.Dispatch<React.SetStateAction<Product[]>>
  const [filteredProducts, setFilteredProducts] = useState<Product[]>([]);
  const [products, setProducts] = useState<Product[]>([]); // Store fetched products
  const [showResults, setShowResults] = useState(false);
  const [activeIndex, setActiveIndex] = useState<number>(-1);

  // Fetch products from Sanity when the component mounts
  useEffect(() => {
    async function fetchProducts() {
      try {
        const fetchedProducts: Product[] = await client.fetch(allProducts); // Using your allProducts query
        console.log("Fetched Products:", fetchedProducts); // Log the fetched products
        setProducts(fetchedProducts);
      } catch (error) {
        console.error("Error fetching products:", error);
```

File explorer sidebar:

∨ src
  ∨ app
    ∨ components
       AddToCartButt...
       BlogSection.tsx
       Discount.tsx
       FeatureProduct....
       Footer.tsx
       Header.tsx
       Hero.tsx
       LatestProducts.t...
       Newslater.tsx
       Offers.tsx
       SearchBar.tsx
       TopBar.tsx
       TopCategories.tsx
       TrendingProduc...
       Unique.tsx
   ∨ contact-us
       page.tsx
   ∨ createaccount
       page.tsx
   > creatorpage
   > faq

> OUTLINE
> TIMELINE