**- In “SNEAKER\_ECOMMERCE\_WEBSITE\_PROJECT/backend/routes/productRoute.js” file, update the following code:**

import express from 'express';

import Product from '../models/productModel';

import { isAuth, isAdmin } from '../util';

const router = express.Router();

router.get("/", async (req, res) => {

    const products = await Product.find({});

    res.send(products);

});

router.get("/:id", async (req, res) => {

    const product = await Product.findOne({\_id: req.params.id});

    if(product) {

        res.send(product);

    } else {

        res.status(404).send({ message: "Product Not Found" });

    }

});

// Creating Product

router.post("/", isAuth, isAdmin, async (req, res) => {

    const product = new Product({

        name: req.body.name,

        price: req.body.price,

        image: req.body.image,

        brand: req.body.brand,

        category: req.body.category,

        countInStock: req.body.countInStock,

        description: req.body.description,

        rating: req.body.rating,

        numReviews: req.body.numReviews,

    });

    const newProduct = await product.save();

    if(newProduct) {

        return res.status(201).send({ message: 'New Product Created', data: newProduct });

    }

    return res.status(500).send({ message: 'Error in Creating Product.' });

})

// Editing Product

router.put("/:id", isAuth, isAdmin, async (req, res) => {

    const productId = req.params.id;

    const product = await Product.findById(productId);

    if(product){

        product.name = req.body.name;

        product.price = req.body.price;

        product.image = req.body.image;

        product.brand = req.body.brand;

        product.category = req.body.category;

        product.countInStock = req.body.countInStock;

        product.description = req.body.description;

        const updatedProduct = await product.save();

        if(updatedProduct) {

            return res.status(200).send({ message: 'Product Updated', data: updatedProduct });

        }

    }

    return res.status(500).send({ message: 'Error in Updating Product.' });

})

// Deleting Product

router.delete("/:id", isAuth, isAdmin, async (req, res) => {

    const deletedProduct = await Product.findById(req.params.id);

    if(deletedProduct) {

        await deletedProduct.remove();

        res.send({message: "Product Deleted"});

    } else {

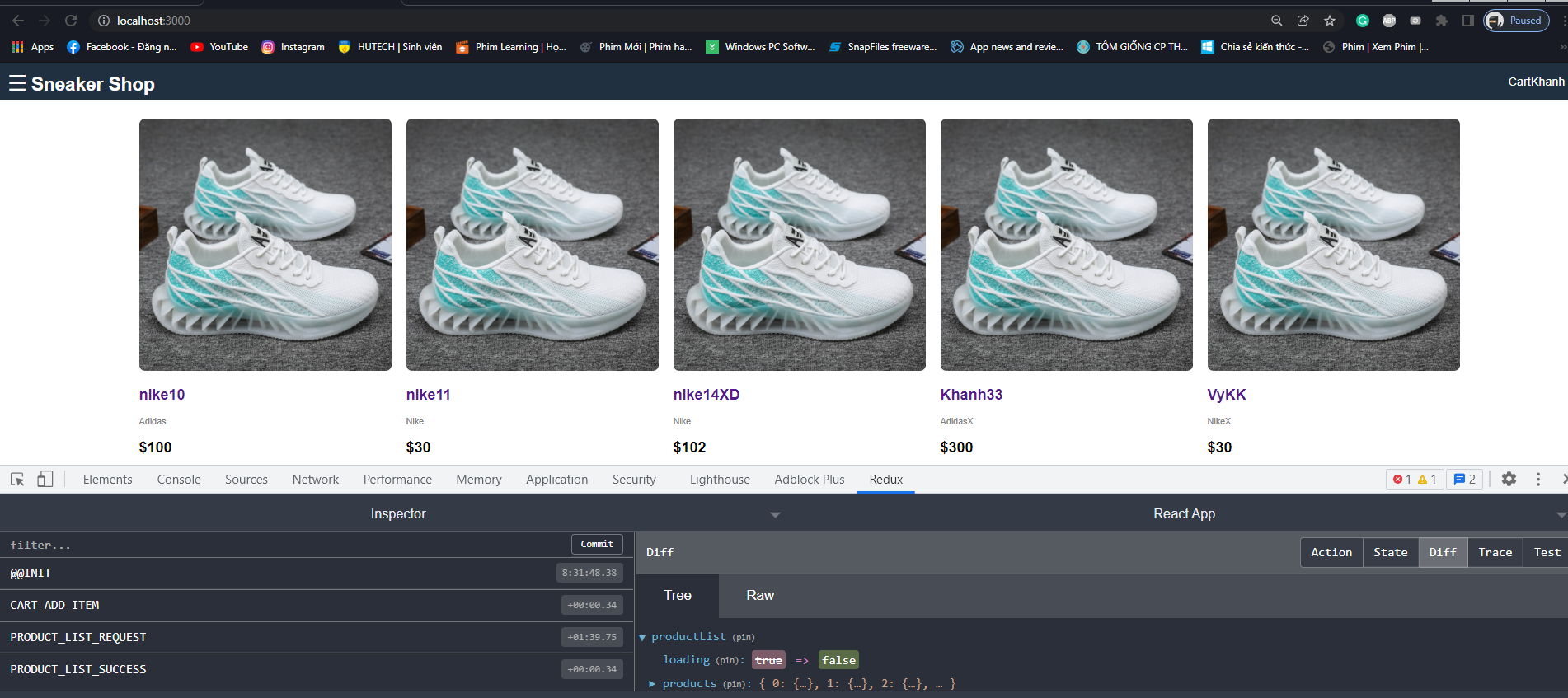
        res.send("Error in Deletion.");

    }

});

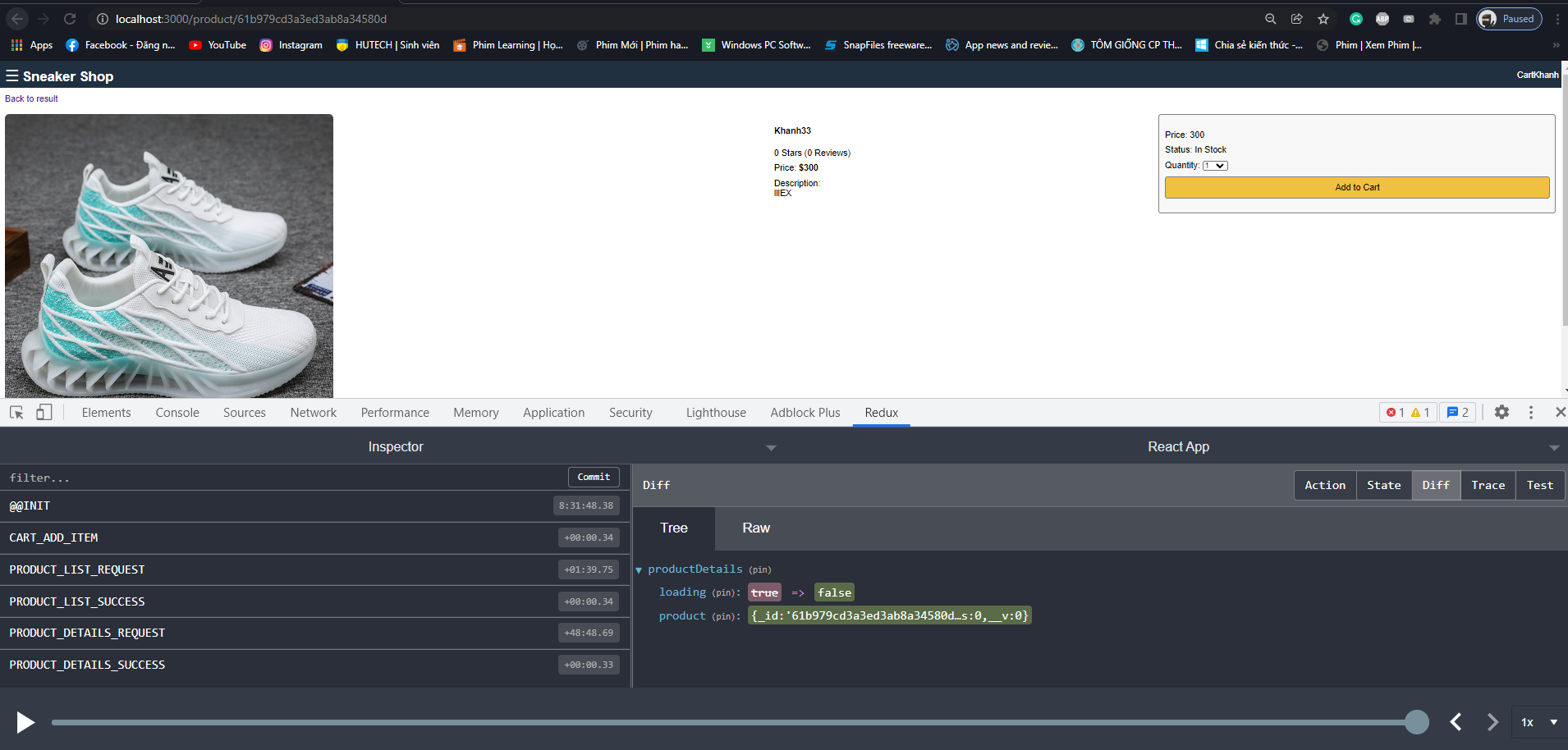
export default router;

**- Result: to checking ... 4:40:50 (Signed)**



Click on any product

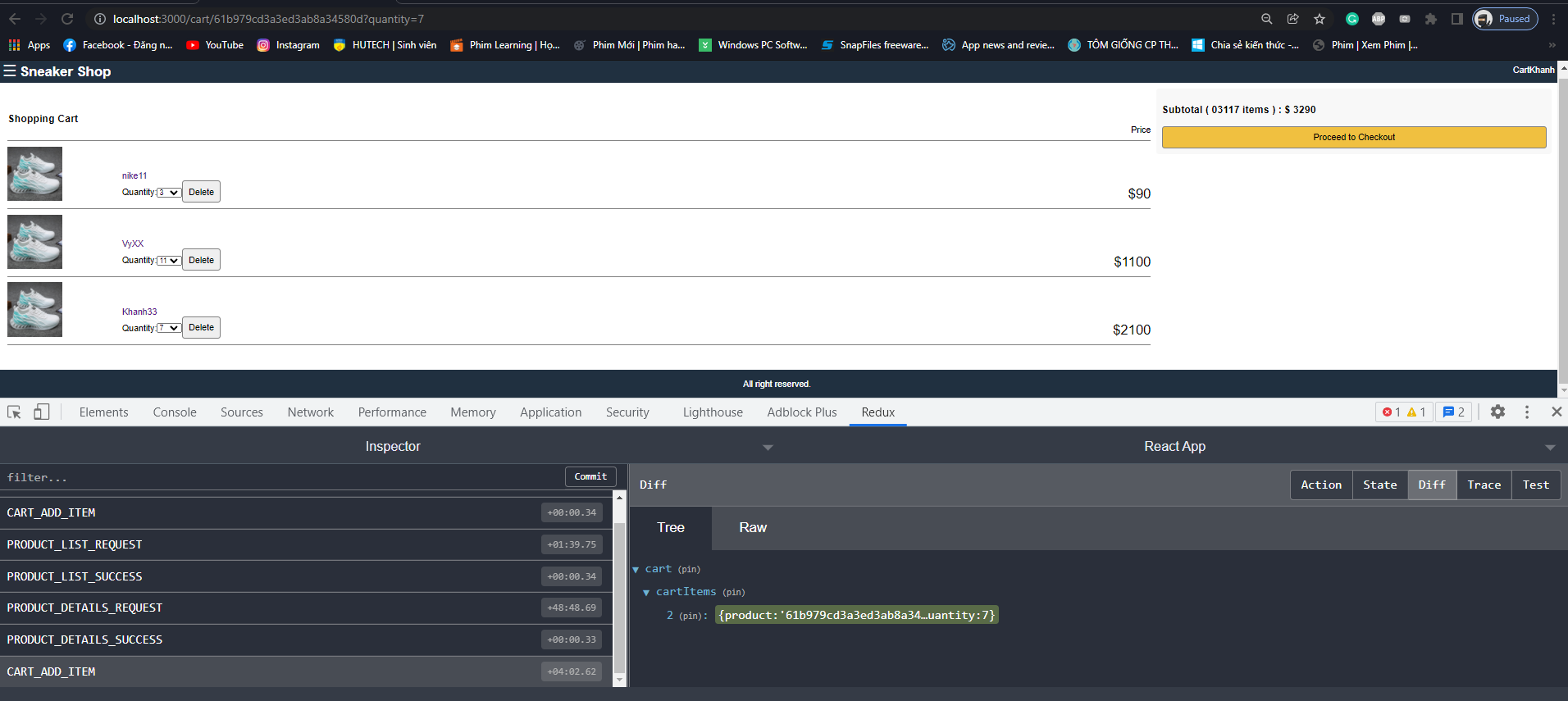
**Image 1: Step 1**



This is Product’s details

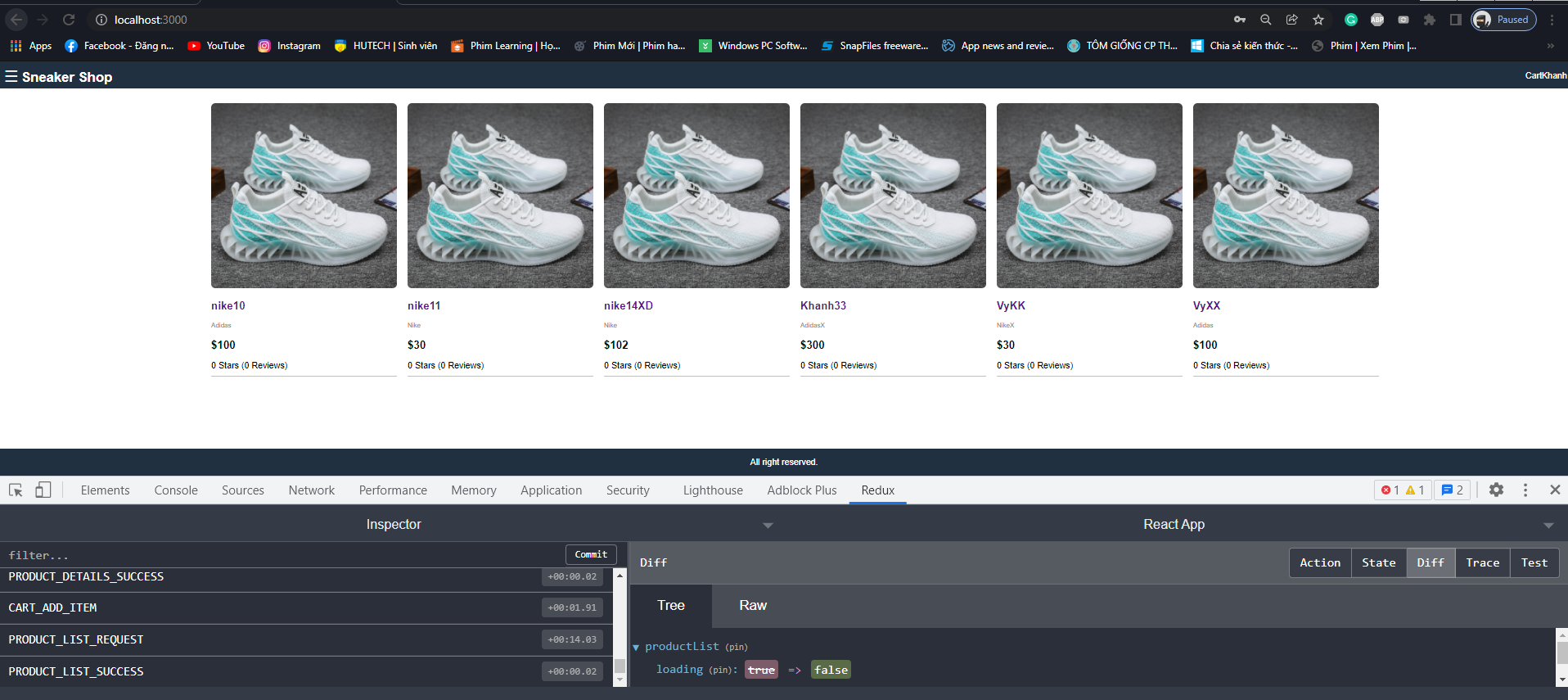
Select quantity and click “Add to Cart” button

**Image 2: Step 2**



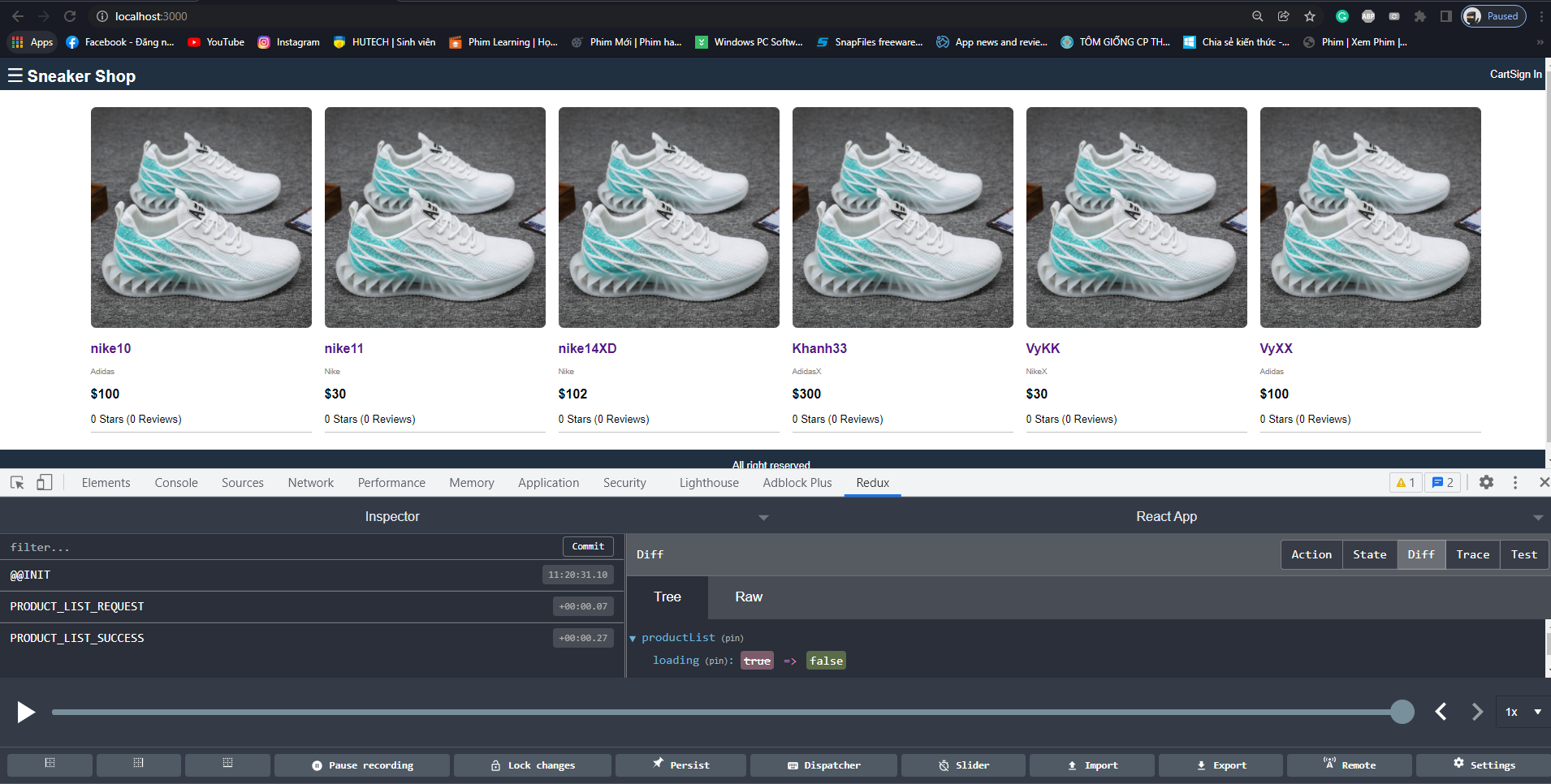
Press “Process to Checkout” button

**Image 3: Step 3**



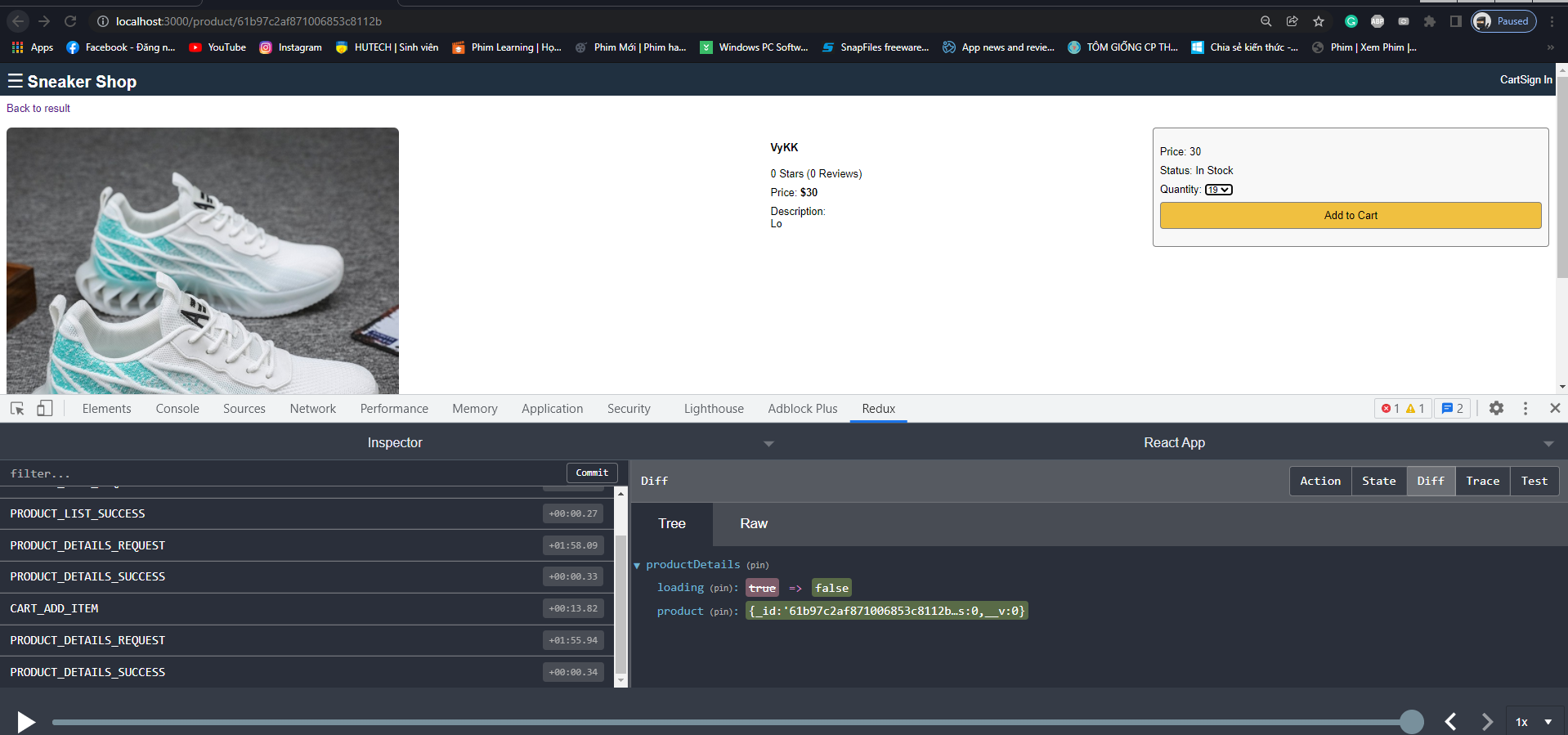
**Image 4: Result**

**- Result: to checking ... 4:40:50 (Not Signed in yet)**



Click on any product

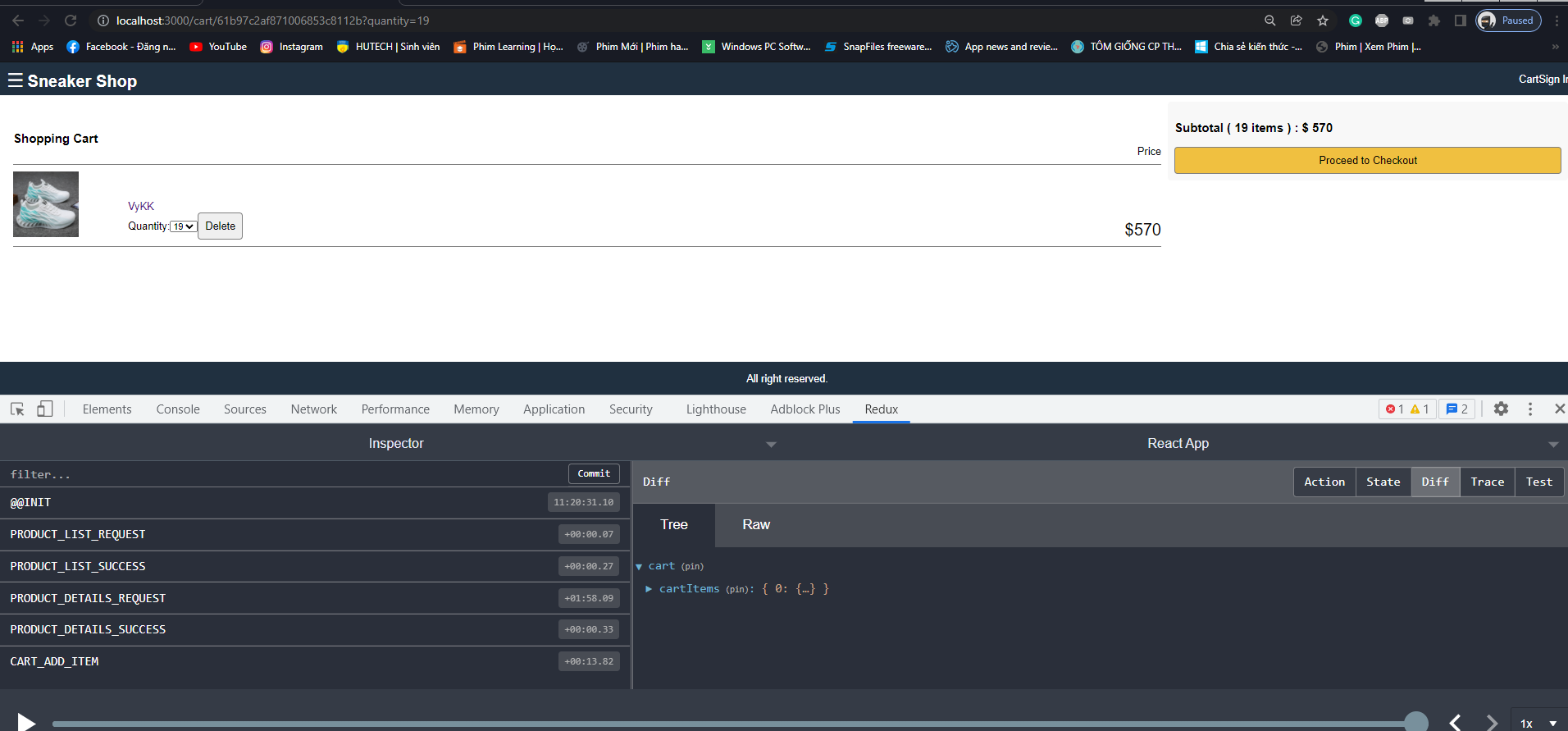
**Image 1: Step 1**



This is Product’s details

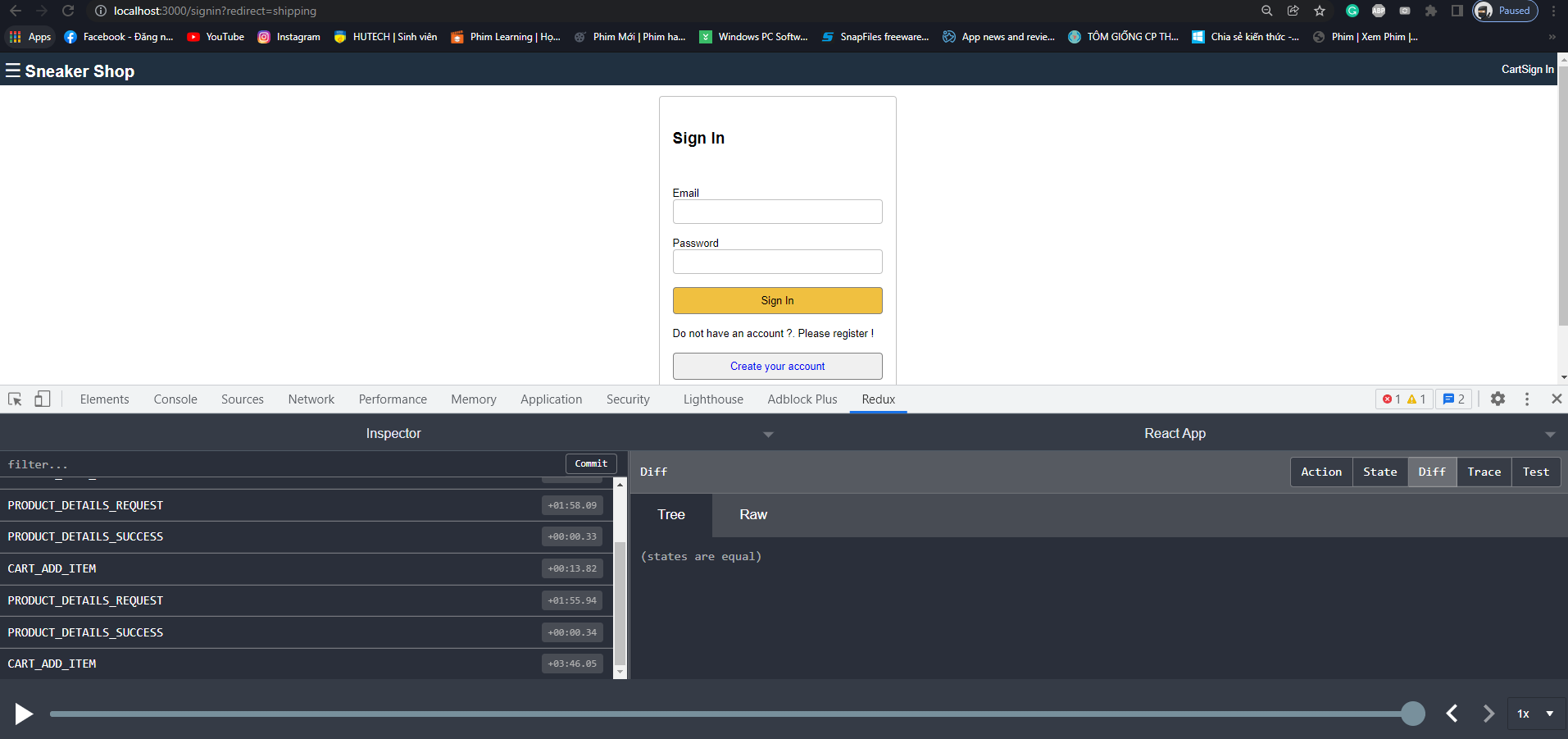
Select quantity and click “Add to Cart” button

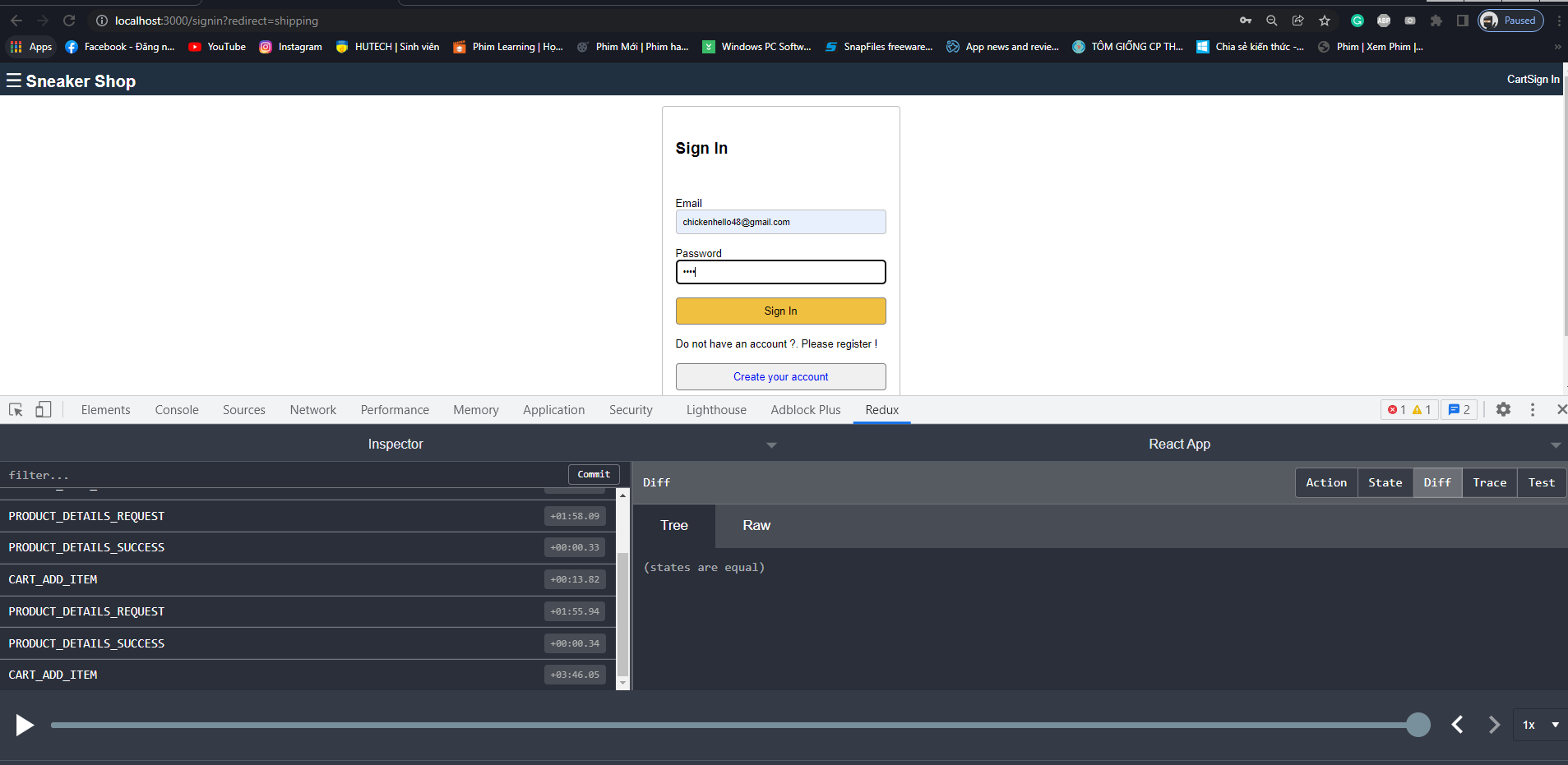
**Image 2: Step 2**



Press “Process to Checkout” button

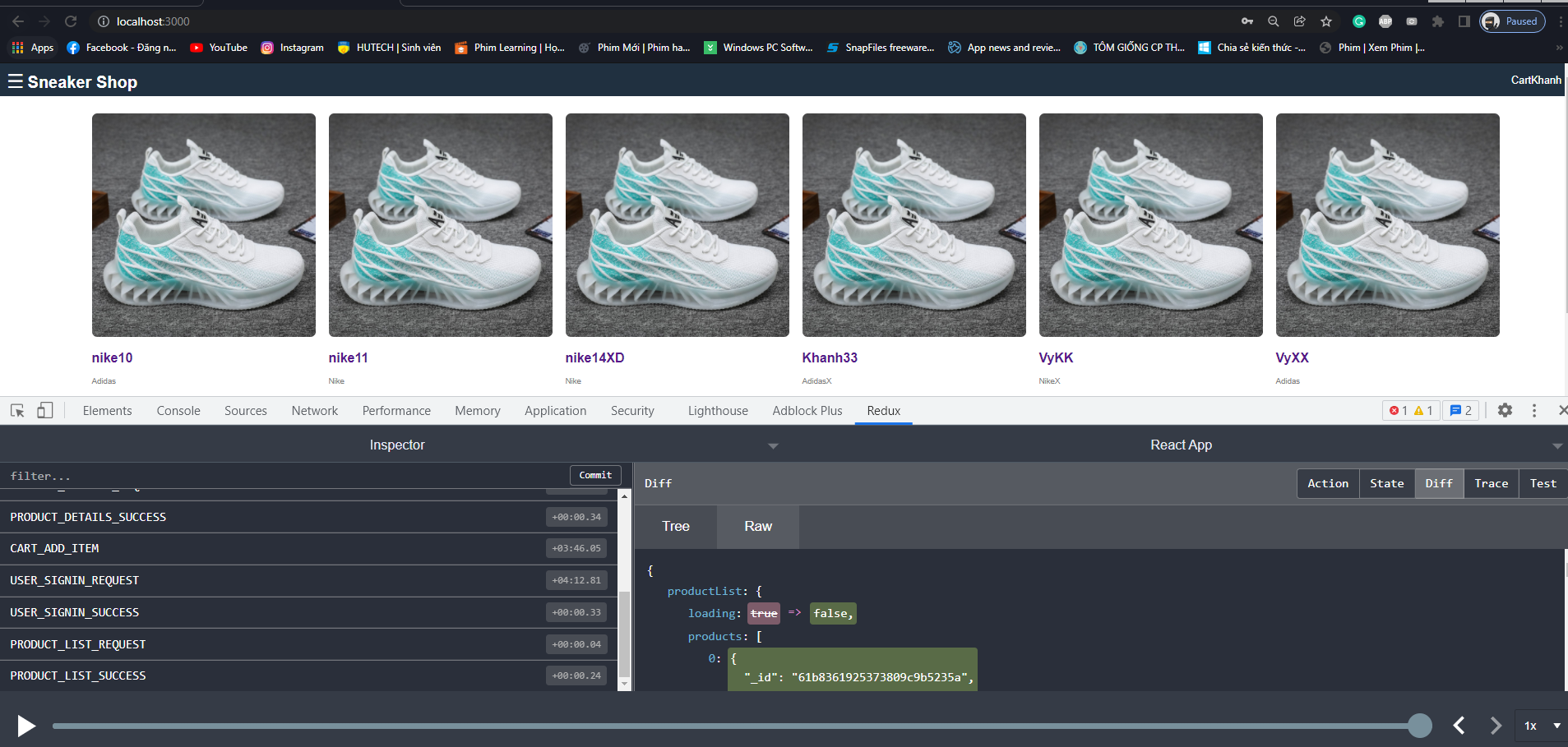
**Image 3: Step 3**





Filled in the “Sign In” form and press “Sign In” button

**Image 4,5: Step 4**



**Image 6: Result**

**- Create “SNEAKER\_ECOMMERCE\_WEBSITE\_PROJECT/frontend/src/actions/userActions.js” file, add the following code:**

**(Error 3 ... 3:55:26) Nếu làm như trong video mà tại lúc test đăng ký, khi nhập hết các field và nhấn nút đăng ký mà nó bắn ra lỗi 500.**

**(Solution (Error 3)):**

**- Result: to checking ... 3:55:26**

2