|  |
| --- |
| **МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ** |
| **федеральное государственное бюджетное образовательное учреждение высшего образования «Российский экономический университет имени Г.В. Плеханова»**  **Московский приборостроительный техникум** |

**ОТЧЕТ**

по учебной практике

УП 01.01 «Учебная практика»

*индекс по УП и наименование практики*

Профессионального модуля ПМ.01 «Разработка модулей программного обеспечения для компьютерных систем»

*индекс по УП и наименование профессионального модуля*

Специальность 09.02.07 «Информационные системы и программирование»

*код и наименование специальности*

Студент Сорокина Вероника Захаровна

*(фамилия, имя, отчество)*

Группа П-1-23

Руководитель по практической подготовке от техникума

Юшина Дарья Денисовна

*(фамилия, имя, отчество)*

«\_\_\_\_» \_\_\_\_\_\_\_\_\_ 2025 года

СОДЕРЖАНИЕ

[Лабораторная работа №1 «Карта товара» 3](#_Toc208222474)

[Лабораторная работа №2 «Создание корзины» 6](#_Toc208222475)

## Лабораторная работа №1 «Карта товара»

Цель работы: разработать React-приложение с компонентом карточки товара, реализующее функционал добавления товара в корзину с помощью счётчика.

1. На занятиях были изучены и созданы приложение с кнопкой, счётчиком и отправкой данных.

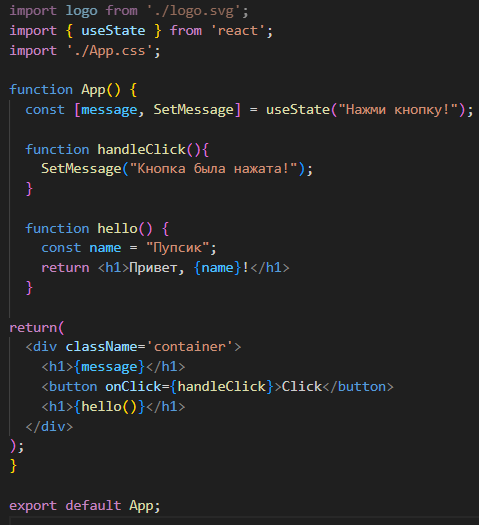


Рисунок 1 – Приложение с нажатием кнопки.

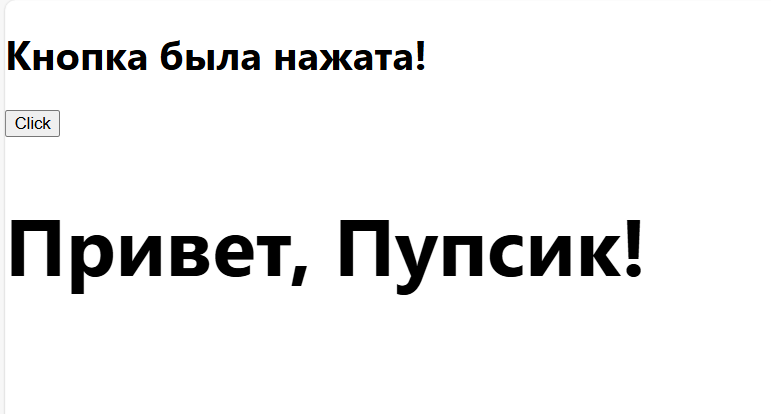


Рисунок 2 – Приложение с нажатием кнопки 2.

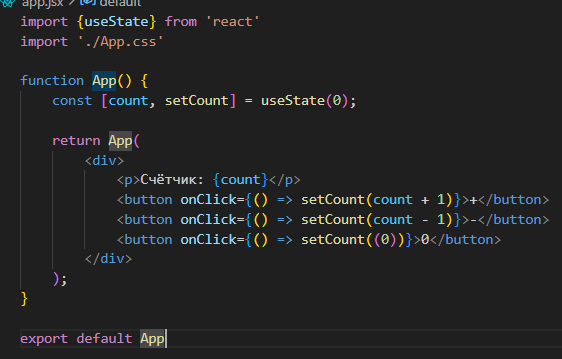


Рисунок 3 – Приложение с счётчиком.

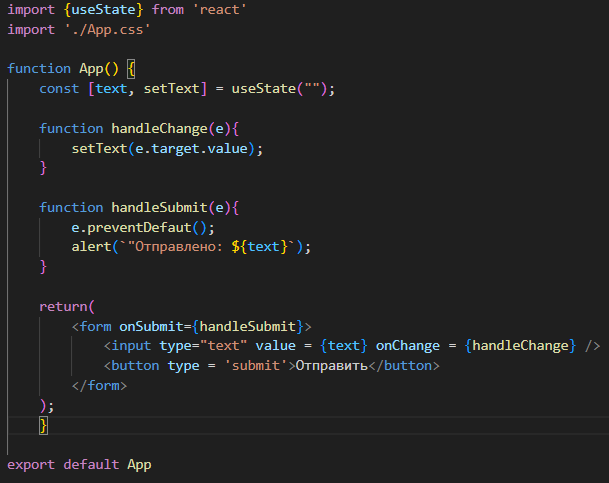


Рисунок 4 – Приложение с отправлением данных.

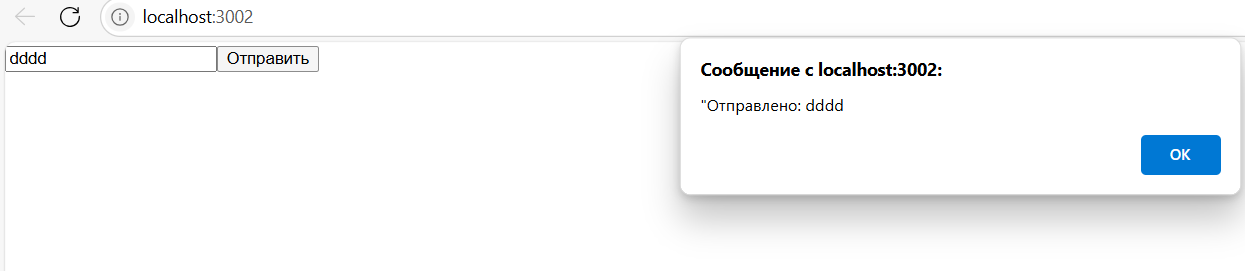


Рисунок 5 – Приложение с отправлением данных 2.

1. Для выполнения работы необходимо объединить знания, полученные на занятиях. Компонент App, используется для отображения карточек товаров и наполнения их собственными данными.

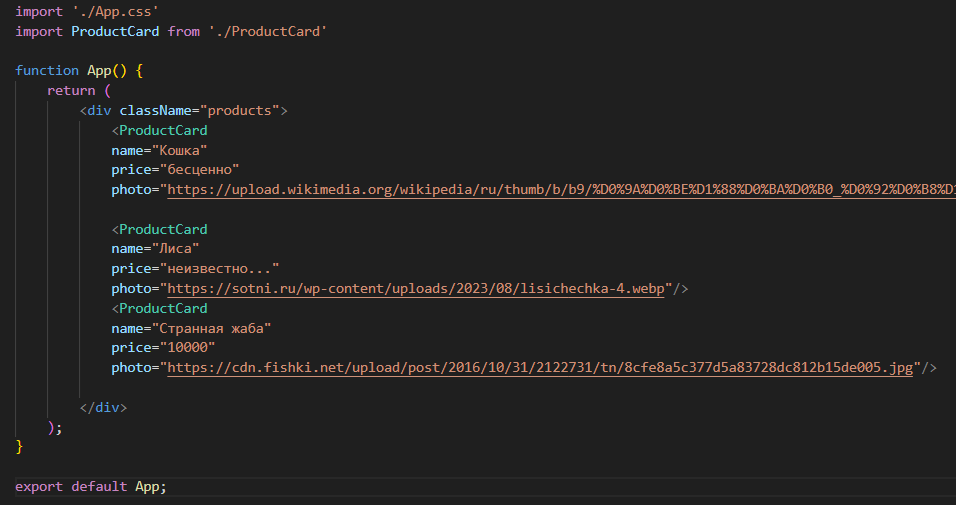


Рисунок 6 *–* App*.*

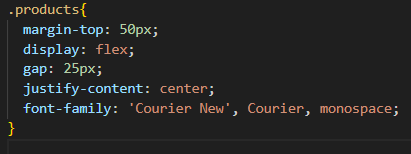


Рисунок 7 – App.css.

Компонент CountProduct необходим для реализации счётчика. Функция handleClick реагирует на нажатие кнопки и прибавляет единицу к счётчику.

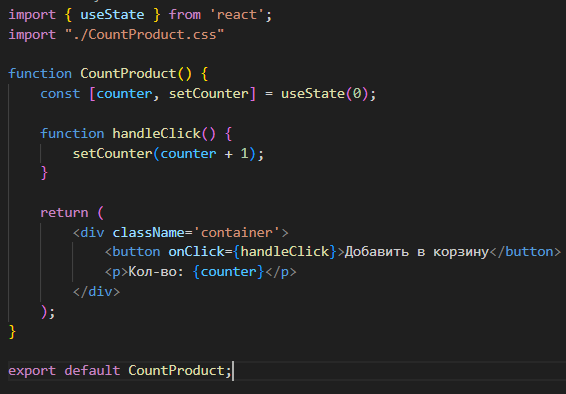


Рисунок 8 – CountProduct.

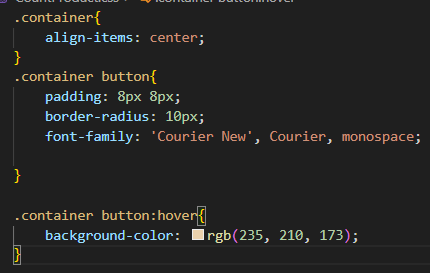


Рисунок 9 – CountProduct.css.

ProductCard содержит в себе универсальный шаблон карточки. Все карточки имеют название, цену, фото и счётчик с кнопкой добавления в корзину.

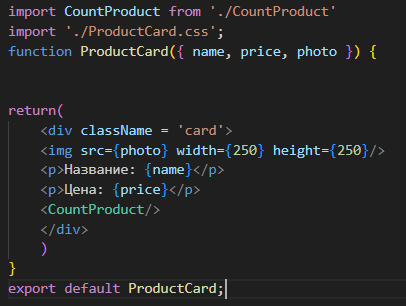


Рисунок 10 – ProductCard.

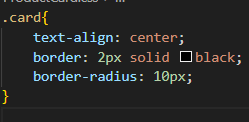


Рисунок 11 - ProductCard.css.

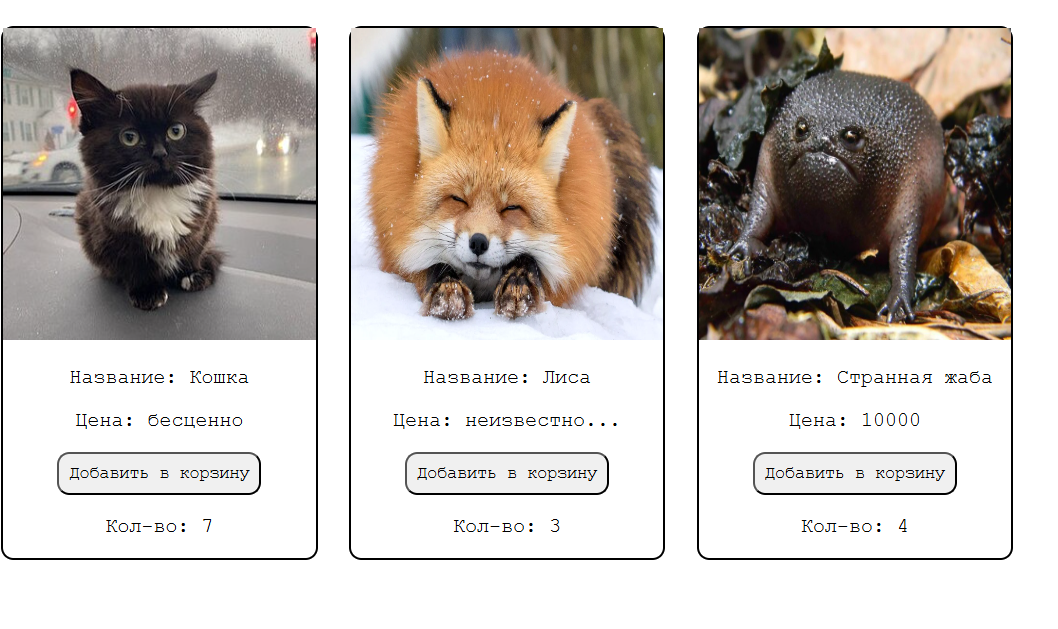


Рисунок 12 – Результат работы приложения.

Вывод: в ходе работы было создано приложение на React, состоящее из нескольких карт товара и возможности добавления товара в корзину через счётчик.

## Лабораторная работа №2 «Создание корзины»

Цель работы: изучить навигацию и context Api в React. Реализовать функционал корзины товаров при помощи context api (добавление, удаление, подсчёт общей суммы).

1. Первая часть. На занятиях были изучены библиотека «react-router-dom» для создания навигации и context api для передачи данных между компонентами. В первой части необходимо было создать context для добавления и подсчёта товаров в корзине.

В App.jsx указаны основные пути навигации routes, а также context для темы приложения, подсчёта товаров и меню навигации.



Рисунок 13 – App.jsx.

ThemeContext создаёт context для смены темы(светлая/тёмная), с помощью функции toggleTheme устанавливается нужная тема.

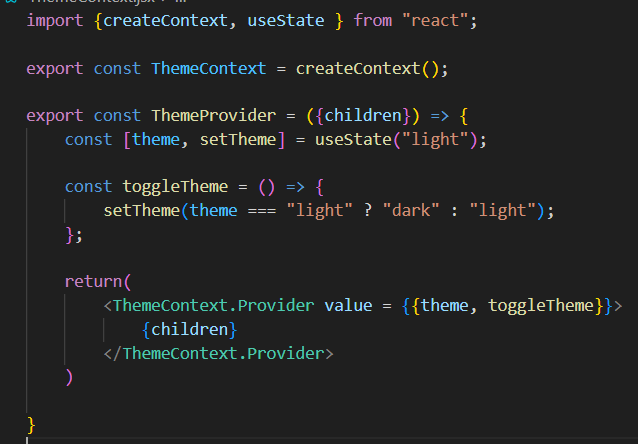


Рисунок 14 – ThemeContext.

В Menu реализованы меню навигации и стили для него, для смены темы используется кнопка с функцией toggleTheme, созданная ранее в ThemeContext.

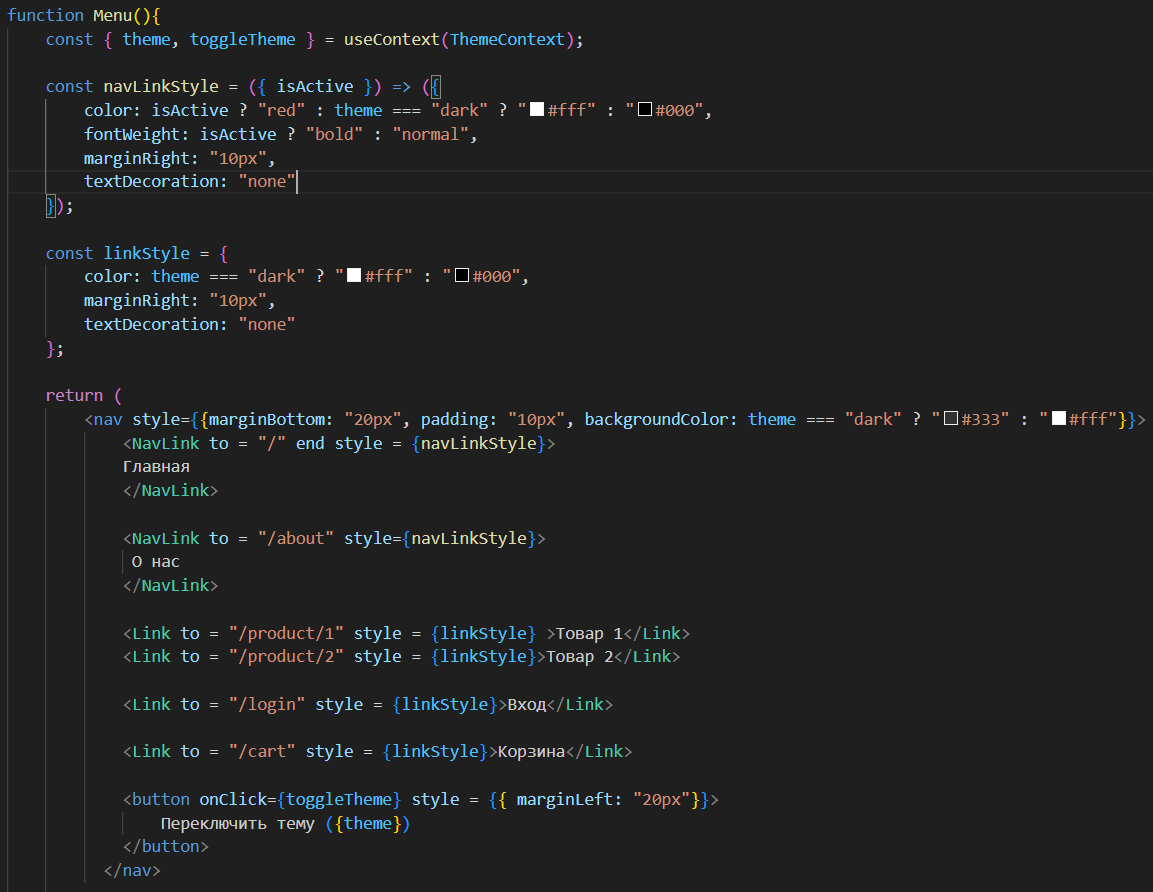


Рисунок 15 – Menu.jsx.

В CartContext реализован счётчик, которые будет подсчитывать добавленные в корзину товары.



Рисунок 16 – CartContext.

Product реализует шаблон карточки товара. Товар содержит id, кнопку для добавления и счётчик.

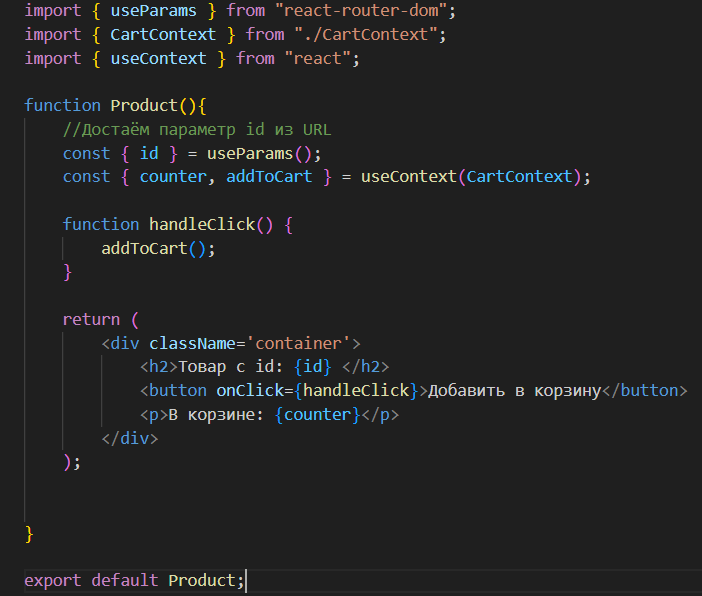


Рисунок 17 – Product.

Cart отображает общее число всех добавленных товаров.

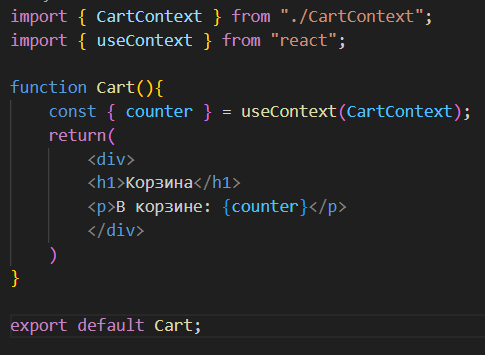


Рисунок 18 – Cart.jsx.

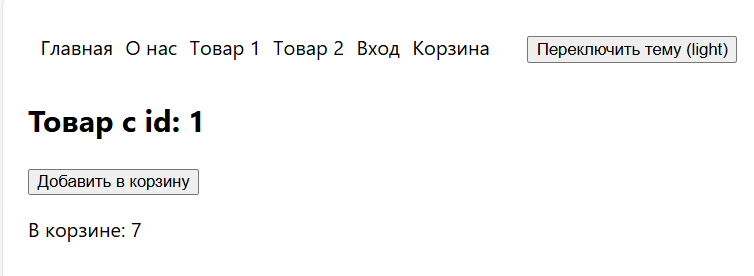


Рисунок 19 – Результат работы программы.

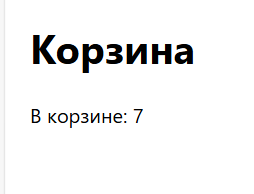


Рисунок 20 – Результат работы программы 2.

1. Вторая часть. Во вторая части работы необходимо создать улучшенную корзину с разделением товаров, удалением и подсчётом суммы.



Рисунок 21 – App.jsx.

Теперь при добавлении товара в корзину будет появляется кнопка «Перейти в корзину». Каждый товар имеет имя, цену и свой ID.



Рисунок 22 – ProductCard.jsx.

В Home заполняются карточки для отображения на главной странице.

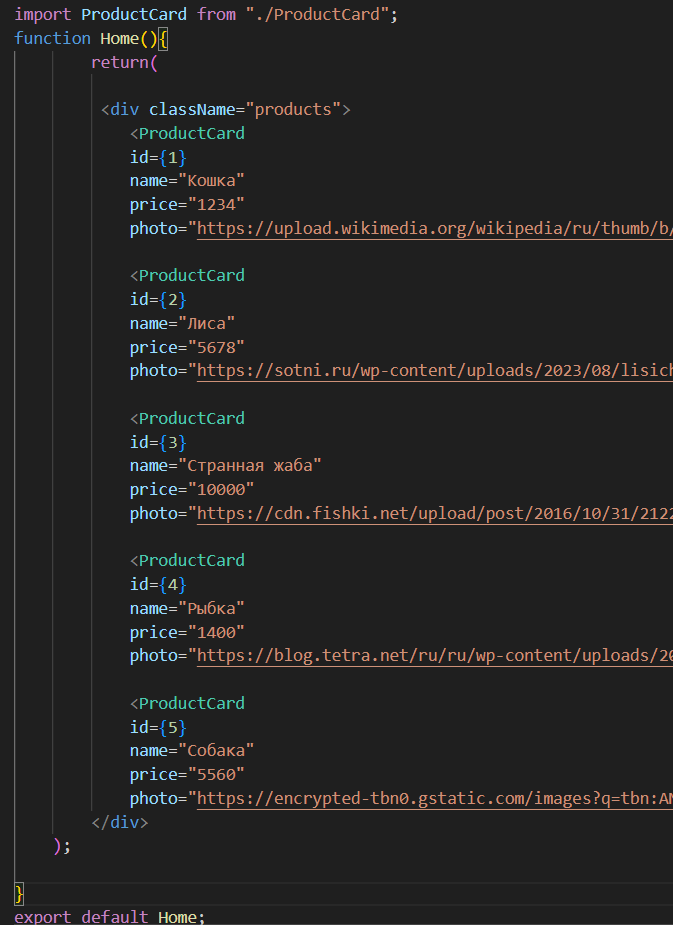


Рисунок 23 – Home.jsx.

При нажатии на «Логин» в навигационном меню, будет показана кнопка «Войти». При нажатии кнопки произойдёт переход на dashboard.



Рисунок 24 – Login.jsx.

В GlobalStyles заданы стили для приложения.

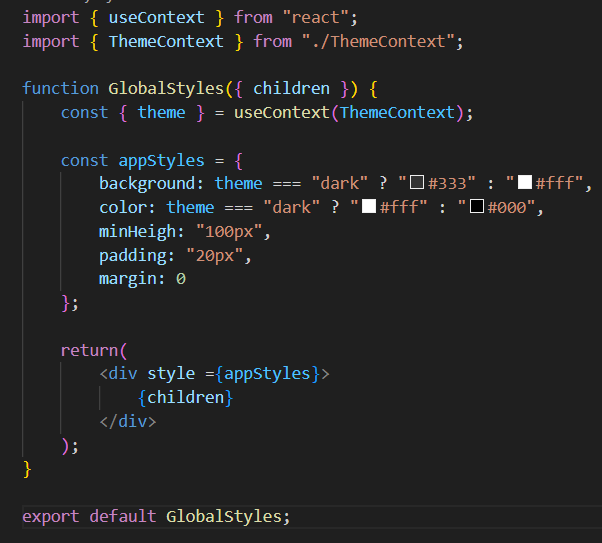


Рисунок 25 – GlobalStyles.jsx.

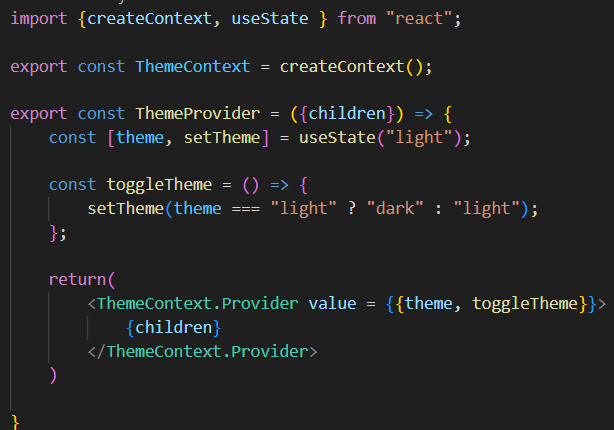


Рисунок 26 – ThemeContext.jsx.

В Menu теперь используется только navLink, так как он позволяет отображать состояние активности ссылки.

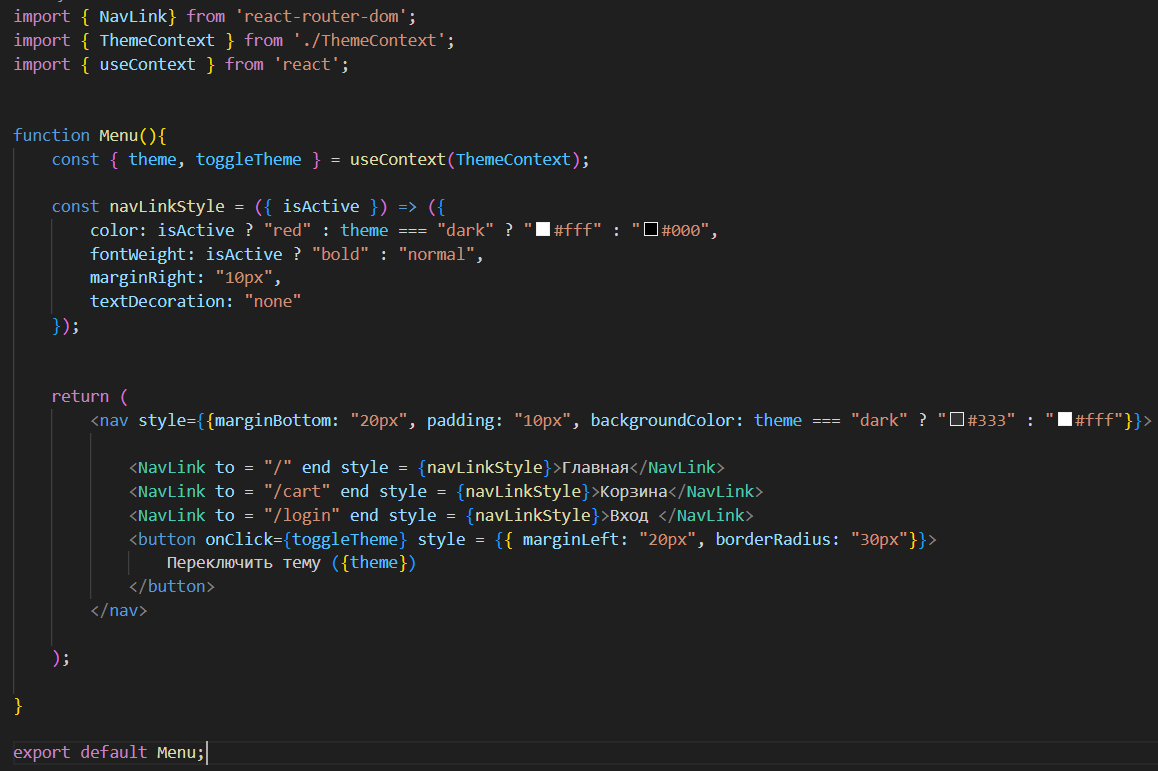


Рисунок 27 – Menu.jsx.

В CartContext реализованы функции для корзины. AddToCart позволяет добавлять новый товар в конец массива items и увеличивает счётчик. RemoveToCart при помощи filter исключает ненужное id из массива и создаёт копию массива. CountPrice при помощи reduce считает итоговую цену всех товаров, складывая всё в одну переменную.

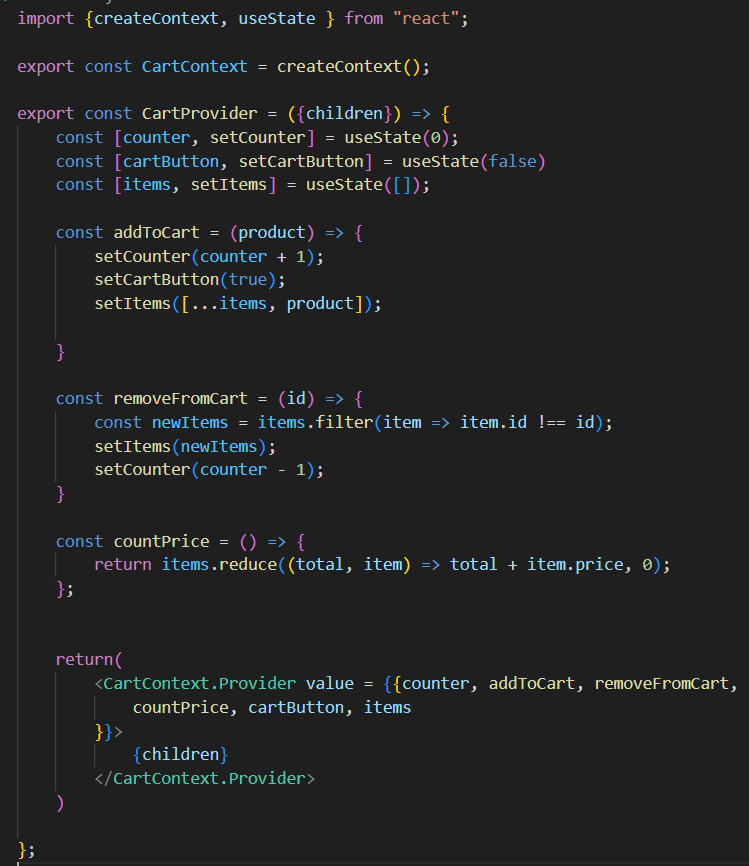


Рисунок 28 – CartContext.jsx.

При помощи map в корзине появляются новые товары по определённому шаблону – имя и цена, а также реализована кнопка удаления из корзины. CountPrice считает и отображает итоговую цену.



Рисунок 29 – Cart.jsx.

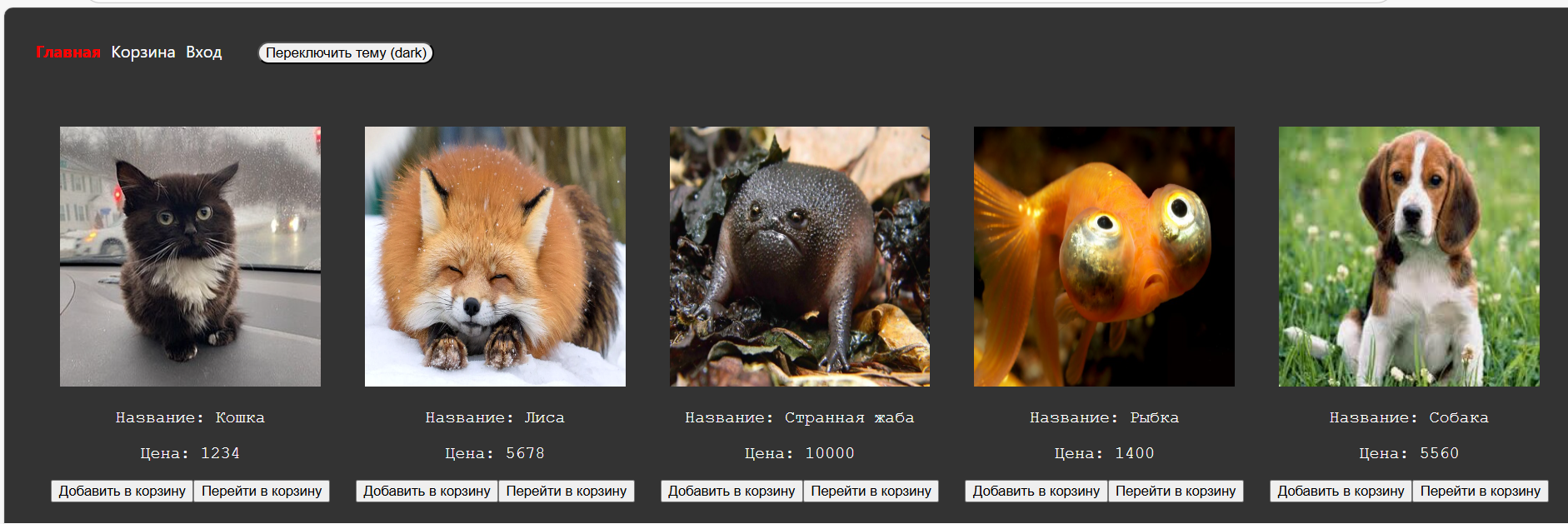


Рисунок 30 – Результат работы программы.



Рисунок 31 – Результат работы программы.

Вывод: в ходе работы была освоена работа с навигацией и context Api в React, а также создана корзина товаров, позволяющая добавлять, удалять товары при помощи context Api.

## Лабораторная работа №3 «Создание базы данных»

Цель работы: создать базу данных для хранения данных о пользователях, ролях, услугах, записях и платежах для магазина животных. Заполнить базу данных тестовыми данными.

База данных содержит 5 таблиц – roles, users, services, appointments и payments. Таблица roles содержит название роли (сотрудник/клиент). Таблица users содержит логин, пароль, ФИО, пол, адрес, номер телефона и внешний ключ, который ссылается на таблицу roles. Для пола создана отдельная проверка на «мужской» и «женский». Таблица services содержит название услуги и цену. Таблица appointments содержит внешний ключ, который ссылается на services, дату и время записи, 2 внешних ключа на users (сотрудник и клиент). Для даты создана проверка, что запись не будет в прошлом.

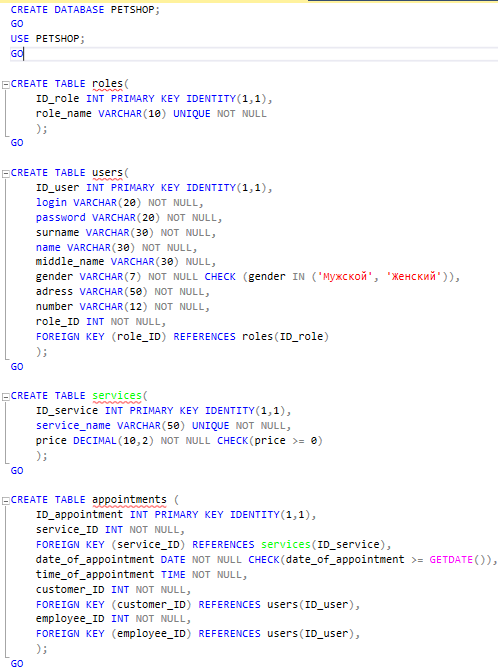


Рисунок 32 – Создание базы данных и таблиц.

Таблица payments включает в себя цену оплаты, дату и время оплаты (если не указать вручную, то она внесётся автоматически и будет равна настоящей), внешний ключ на appointments и users. При помощи Insert таблицы заполняются данными.

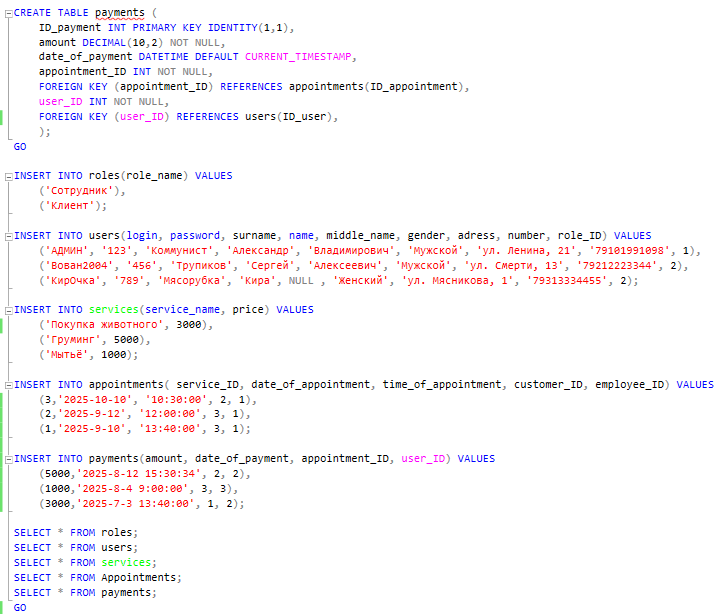


Рисунок 33 – Создание и заполнение таблиц.

Select выводит заполненные таблицы для просмотра.

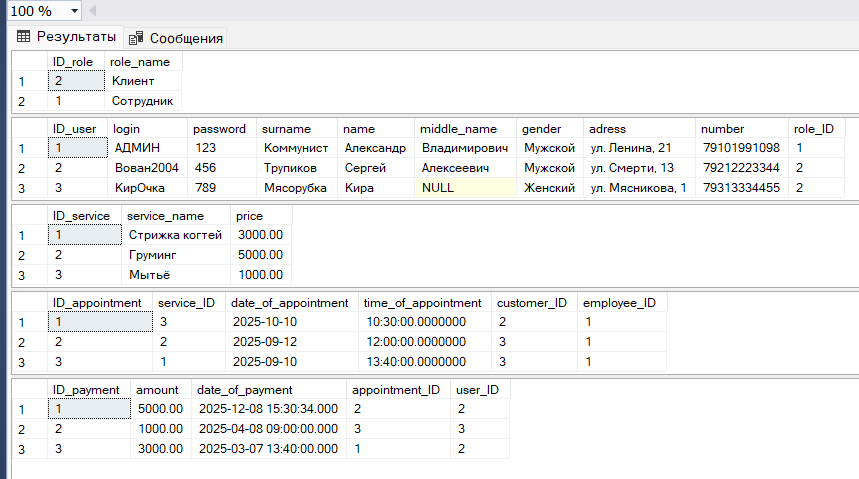


Рисунок 34 – Вывод таблиц.

Вывод: в ходе работы была разработана, а также заполнена пробной информацией база данных для хранения данных о пользователях и работе магазина животных.