Министерство образования Республики Беларусь

Учреждение образования

“Брестский государственный университет”

Кафедра ИИТ

Лабораторная работа №4

По дисциплине: “Веб-технологии ”

Тема: “Изучение JS-фрейморка”

Выполнил:

Студент 2-го курса

Группы ПО-7

Комиссаров А.Е.

Проверил:

Михняев А.Л.

Брест, 2022

**Цель:** Изучить основы JS-фреймворка React JS.

Использованные технологии: React, HTML, CSS, Bootstrap, JavaScript.

Данное веб-приложение представляет собой мини-корзину, которая считает кол-во типов добавленных предметов, которые так же можно добавлять по несколько единиц нажимая “+” и убирать нажимая “-“. Так же есть возможность удаления всех одинаковых предметов нажатием кнопки удаления, а когда корзина будет полностью пуста есть возможность вернуть всё в начальное состояние нажав кнопку обновления страницы либо встроенную кнопку перезагрузки.

**Код программы:**

**App.js**

import React, { Component } from "react";

import NavBar from "./components/navbar";

import Counters from "./components/counters";

class App extends Component {

  state = {

    counters: [

      { id: 1, value: 0 },

      { id: 2, value: 0 },

      { id: 3, value: 0 },

      { id: 4, value: 0 },

    ],

  };

  handleIncrement = (counter) => {

    const counters = [...this.state.counters];

    const index = counters.indexOf(counter);

    counters[index] = { ...counters[index] };

    counters[index].value++;

    this.setState({ counters });

  };

  handleDecrement = (counter) => {

    const counters = [...this.state.counters];

    const index = counters.indexOf(counter);

    counters[index] = { ...counters[index] };

    counters[index].value--;

    this.setState({ counters });

  };

  handleReset = () => {

    const counters = this.state.counters.map((c) => {

      c.value = 0;

      return c;

    });

    this.setState({ counters });

  };

  handleDelete = (counterId) => {

    const counters = this.state.counters.filter((c) => c.id !== counterId);

    this.setState({ counters });

  };

  handleRestart = () => {

    window.location.reload();

  };

  render() {

    return (

      <div className="main\_\_wrap">

        <main className="container">

          <div className="card\_\_box">

            <NavBar

              totalCounters={

                this.state.counters.filter((c) => c.value > 0).length

              }

            />

            <Counters

              counters={this.state.counters}

              onReset={this.handleReset}

              onIncrement={this.handleIncrement}

              onDecrement={this.handleDecrement}

              onDelete={this.handleDelete}

              onRestart={this.handleRestart}

            />

          </div>

        </main>

      </div>

    );

  }

}

export default App;

**Counter.jsx**

import React, { Component } from "react";

class Counter extends Component {

  render() {

    return (

      <div>

        <div className="row">

          <div className="">

            <span style={{ fontSize: 24 }} className={this.getBadgeClasses()}>

              {this.formatCount()}

            </span>

          </div>

          <div className="">

            <button

              className="btn btn-secondary"

              onClick={() => this.props.onIncrement(this.props.counter)}

            >

              <i className="fa fa-plus-circle" aria-hidden="true" />

            </button>

            <button

              className="btn btn-info m-2"

              onClick={() => this.props.onDecrement(this.props.counter)}

              disabled={this.props.counter.value === 0 ? "disabled" : ""}

            >

              <i className="fa fa-minus-circle" aria-hidden="true" />

            </button>

            <button

              className="btn btn-danger"

              onClick={() => this.props.onDelete(this.props.counter.id)}

            >

              <i className="fa fa-trash-o" aria-hidden="true" />

            </button>

          </div>

        </div>

      </div>

    );

  }

  getBadgeClasses = () => {

    let classes = "badge m-2 badge-";

    classes += this.props.counter.value === 0 ? "warning" : "primary";

    return classes;

  };

  formatCount = () => {

    const { value } = this.props.counter;

    return value === 0 ? "Zero" : value;

  };

}export default Counter;

**Counters.jsx**

import React, { Component } from "react";

import Counter from "./counter";

class Counters extends Component {

  render() {

    const { onReset, onIncrement, onDelete, onDecrement, counters, onRestart } =

      this.props;

    return (

      <div>

        <div className="row">

          <div className=''>

            <button

              className="btn btn-success m-2"

              onClick={onReset}

              disabled={counters.length === 0 ? "disabled" : ""}

            >

              <i className="fa fa-refresh" aria-hidden="true" />

            </button>

            <button

              className="btn btn-primary m-2"

              onClick={onRestart}

              disabled={counters.length !== 0 ? "disabled" : ""}

            >

              <i className="fa fa-recycle" aria-hidden="true" />

            </button>

          </div>

        </div>

        {counters.map((counter) => (

          <Counter

            key={counter.id}

            counter={counter}

            onIncrement={onIncrement}

            onDecrement={onDecrement}

            onDelete={onDelete}

          />

        ))}

      </div>

    );

  }

}

export default Counters;

**Navbar.jsx**

import React from "react";

// Stateless Functional Component

const NavBar = ({ totalCounters }) => {

  return (

    <nav className="navbar navbar-light">

      <div className="navbar-brand">

        <i className="fa fa-shopping-cart fa-lg m-2" aria-hidden="true" />

        <span

          className="badge badge-pill badge-info m-2"

          style={{ width: 50, fontSize: "24px" }}

        >

          {totalCounters}

        </span>

        Items

      </div>

    </nav>

  );

};

export default NavBar;

**Index.css**

body {

  margin: 0;

  padding: 0;

  font-family: -apple-system, BlinkMacSystemFont, "Segoe UI", "Roboto", "Oxygen",

    "Ubuntu", "Cantarell", "Fira Sans", "Droid Sans", "Helvetica Neue",

    sans-serif;

  -webkit-font-smoothing: antialiased;

  -moz-osx-font-smoothing: grayscale;

  user-select: none;

}

code {

  font-family: source-code-pro, Menlo, Monaco, Consolas, "Courier New",

    monospace;

}

.main\_\_wrap {

  height: 100vh;

  width: 100vw;

  display: flex;

  padding: 50px 0px;

  justify-content: center;

}

.card\_\_box {

  padding: 50px 10px;

  border: 10px solid #7fd8e6;

  border-radius: 12px;

  background-color: #ddf2f5;

}

.row {

  justify-content: center !important;

}

.navbar {

  justify-content: center !important;

  background: transparent !important;

}

.btn {

  cursor: pointer !important;

}

.badge {

  width: 70px !important;

}

**Index.js**

import React from "react";

import ReactDOM from "react-dom";

import "./index.css";

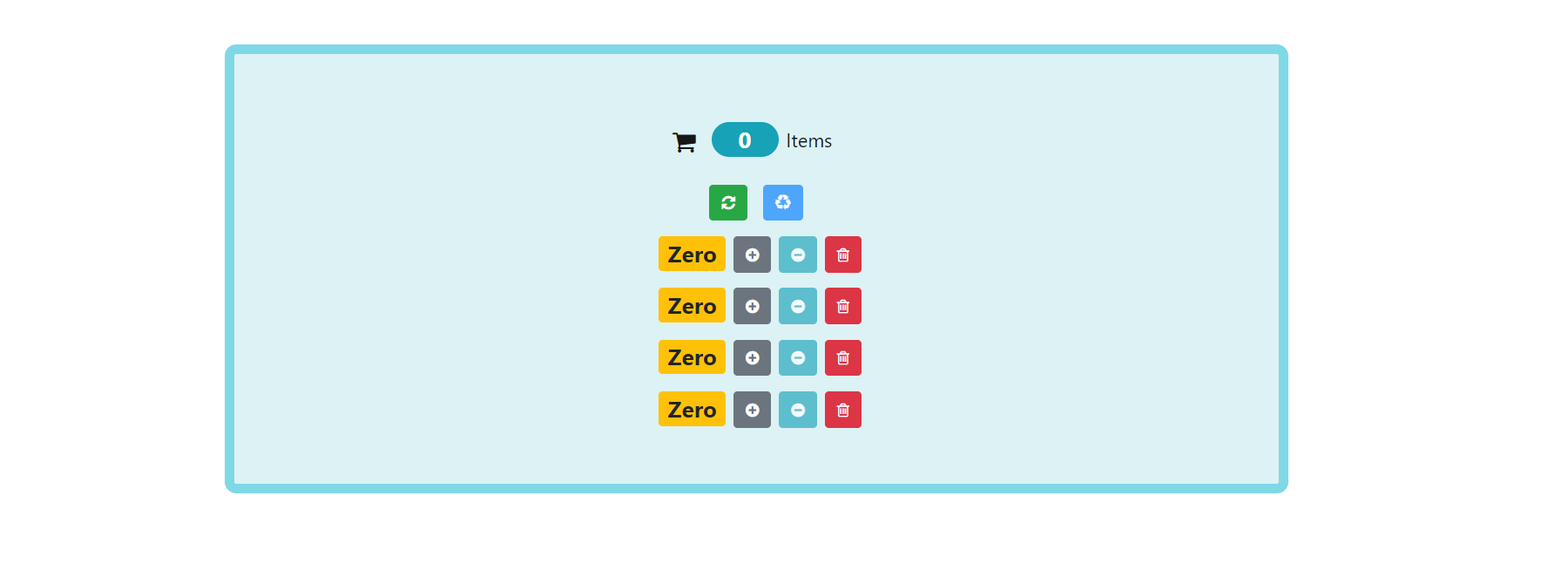
import App from "./App";

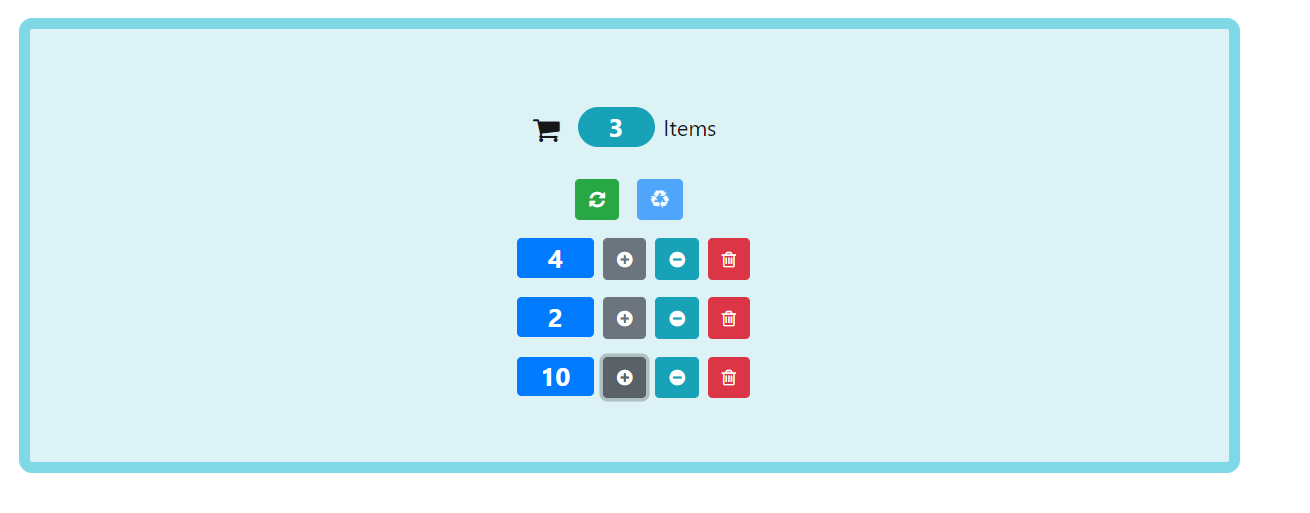
import "bootstrap/dist/css/bootstrap.css";

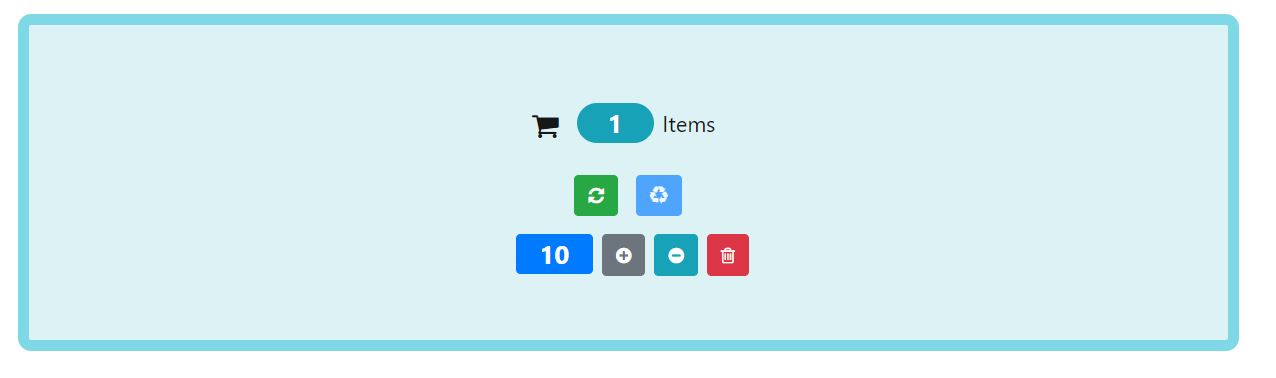
import "font-awesome/css/font-awesome.css";

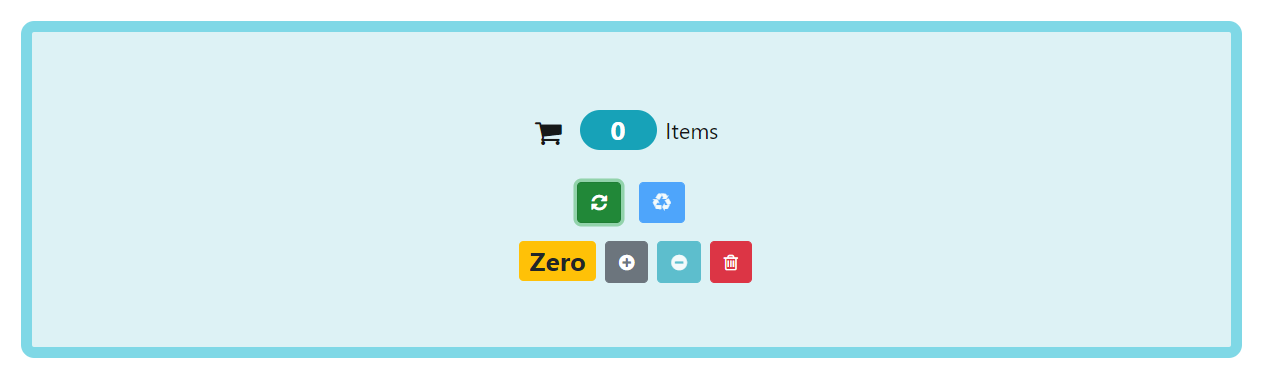
ReactDOM.render(<App />, document.getElementById("root"));

**Результат работы программы:**

****

****

****

****

**Вывод:** В ходе лабораторной работыосвоил базовые принципы работы с фреймворком React JS.