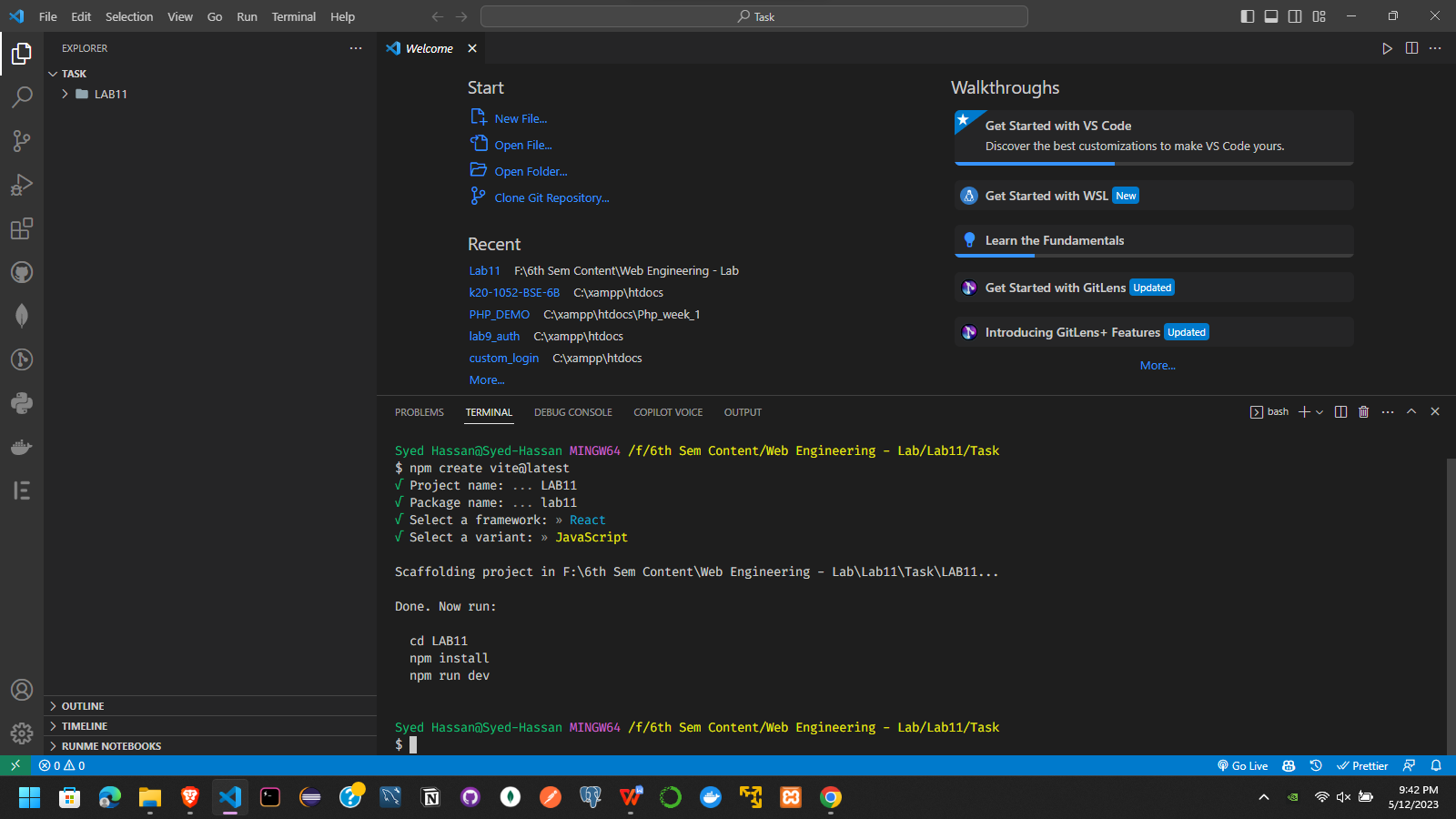
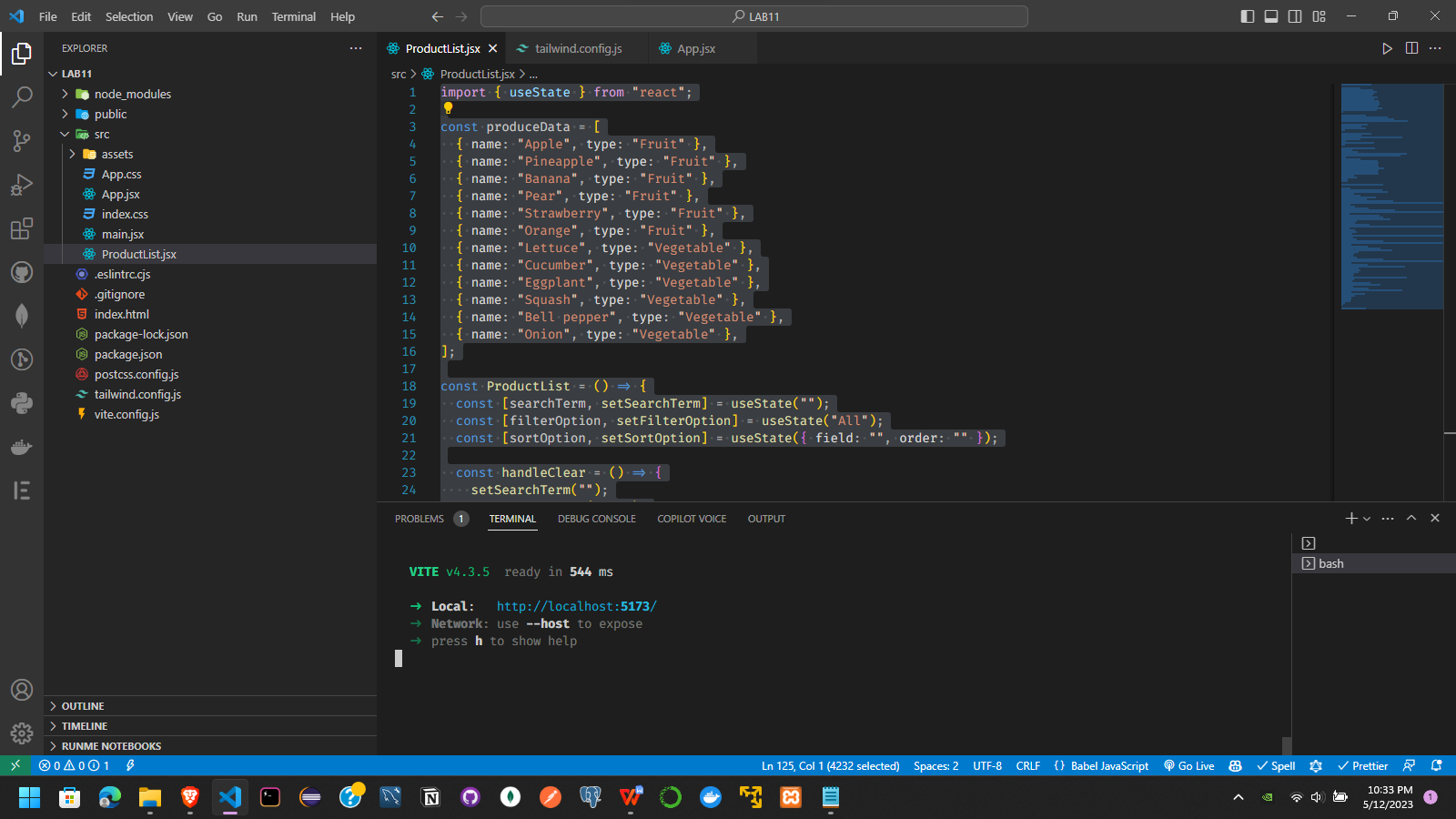
**LAB 11**

**REACT JS**

**K20-1052**





import { useState } from "react";

const produceData = [

  { name: "Apple", type: "Fruit" },

  { name: "Pineapple", type: "Fruit" },

  { name: "Banana", type: "Fruit" },

  { name: "Pear", type: "Fruit" },

  { name: "Strawberry", type: "Fruit" },

  { name: "Orange", type: "Fruit" },

  { name: "Lettuce", type: "Vegetable" },

  { name: "Cucumber", type: "Vegetable" },

  { name: "Eggplant", type: "Vegetable" },

  { name: "Squash", type: "Vegetable" },

  { name: "Bell pepper", type: "Vegetable" },

  { name: "Onion", type: "Vegetable" },

];

const ProductList = () => {

  const [searchTerm, setSearchTerm] = useState("");

  const [filterOption, setFilterOption] = useState("All");

  const [sortOption, setSortOption] = useState({ field: "", order: "" });

  const handleClear = () => {

    setSearchTerm("");

    setFilterOption("All");

  };

  const handleSort = (field) => {

    let order = "asc";

    if (sortOption.field === field && sortOption.order === "asc") {

      order = "desc";

    }

    setSortOption({ field, order });

  };

  const filteredProduce = produceData

    .filter(

      (produce) =>

        produce.name.toLowerCase().includes(searchTerm.toLowerCase()) &&

        (filterOption === "All" || produce.type === filterOption)

    )

    .sort((a, b) => {

      if (sortOption.field === "name") {

        return sortOption.order === "asc"

          ? a.name.localeCompare(b.name)

          : b.name.localeCompare(a.name);

      } else if (sortOption.field === "type") {

        return sortOption.order === "asc"

          ? a.type.localeCompare(b.type)

          : b.type.localeCompare(a.type);

      } else {

        return 0;

      }

    });

  const totalResults = filteredProduce.length;

  return (

    <div className="flex flex-col space-y-2">

      <div className="flex items-center space-x-2">

        <input

          type="text"

          placeholder="Search Produce"

          value={searchTerm}

          onChange={(e) => setSearchTerm(e.target.value)}

          className="border border-gray-400 rounded-md py-2 px-4 w-full md:w-80 focus:outline-none focus:ring-2 focus:ring-blue-500"

        />

        <select

          value={filterOption}

          onChange={(e) => setFilterOption(e.target.value)}

          className="border border-gray-400 rounded-md py-2 px-4 focus:outline-none focus:ring-2 focus:ring-blue-500"

        >

          <option value="All">All</option>

          <option value="Fruit">Fruit</option>

          <option value="Vegetable">Vegetable</option>

        </select>

        <button

          onClick={handleClear}

          className="bg-blue-500 hover:bg-blue-700 text-white font-bold py-2 px-4 rounded focus:outline-none focus:shadow-outline"

        >

          Clear

        </button>

      </div>

      <div className="flex flex-col md:flex-row items-center justify-between space-y-2 md:space-y-0 md:space-x-2">

        <div>

          <button

            onClick={() => handleSort("name")}

            className="bg-gray-200 hover:bg-gray-300 text-gray-800 font-bold py-2 px-4 rounded focus:outline-none focus:shadow-outline"

          >

            Sort by Name{" "}

            {sortOption.field === "name" &&

              (sortOption.order === "asc" ? "▲" : " ▼")}

          </button>

        </div>

        <div>

          <button

            onClick={() => handleSort("type")}

            className="bg-gray-200 hover:bg-gray-300 text-gray-800 font-bold py-2 px-4 rounded focus:outline-none focus:shadow-outline"

          >

            Sort by Type{" "}

            {sortOption.field === "type" && sortOption.order === "asc"

              ? "▲"

              : " ▼"}

          </button>

        </div>

        <div>

          <h2 className="text-xl font-bold">Results: {totalResults}</h2>

        </div>

      </div>

      <ul className="space-y-2">

        {filteredProduce.map((produce) => (

          <li

            key={produce.name}

            className="border border-gray-400 rounded-md py-2 px-4"

          >

            {produce.name}

          </li>

        ))}

      </ul>

    </div>

  );

};

export default ProductList;

