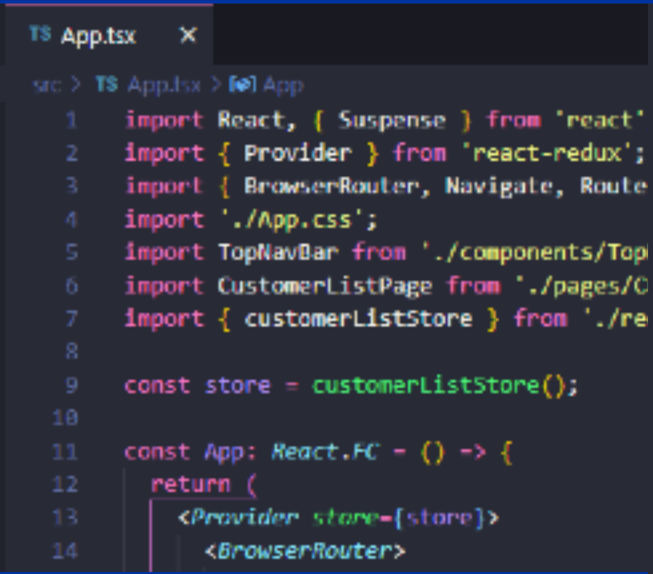
Redux

- Redux is the industry standard and recommended way of managing state but only when prop-drilling because a problem
- You must wrap your app inside a <Provider> which must have a propery of store <Provider store={store}>
- The value of store will be a reference to the **createStore()** that will then reference your **reducers**
- **Reducers** are functions that take the current **state** and action as **arguments**

© 2021 Cognizant

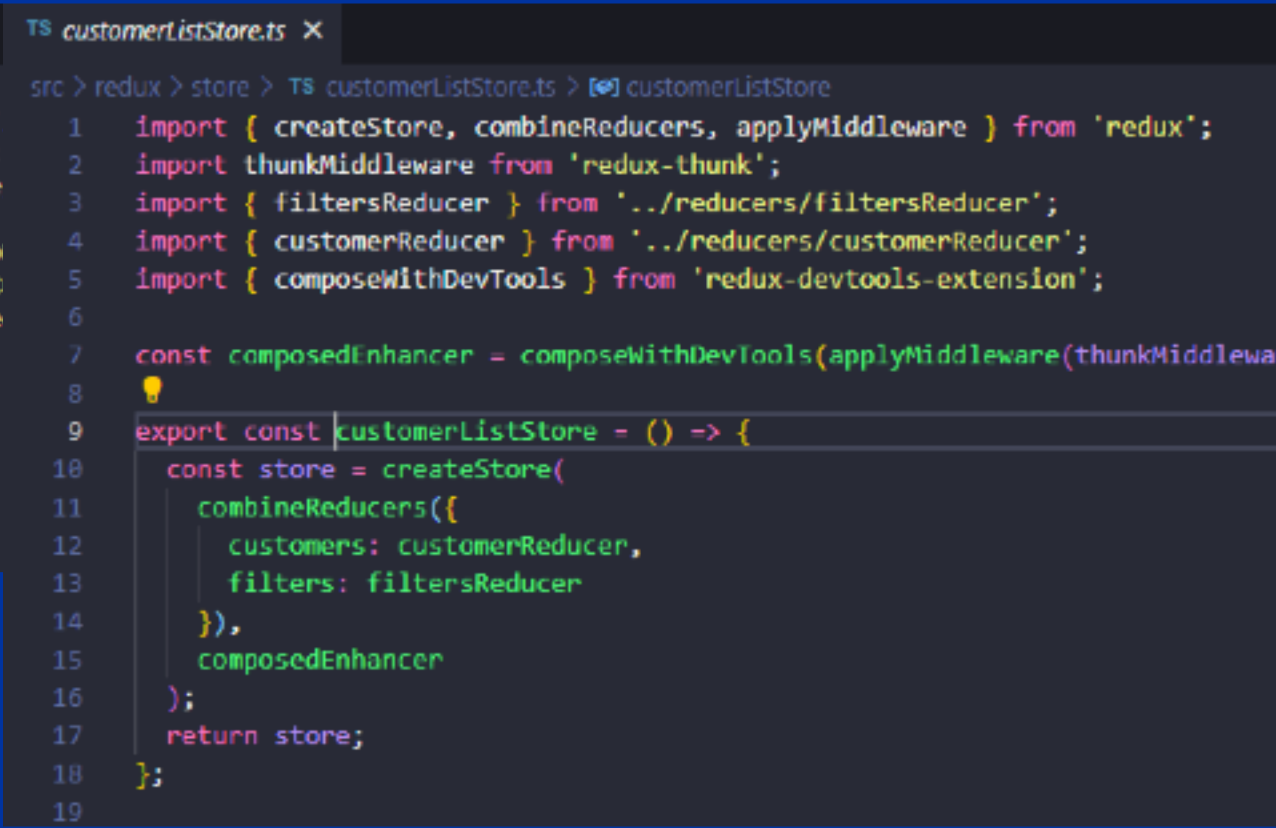
Redux code:



```
import React, { Suspense } from 'react'
import { Provider } from 'react-redux';
import { BrowserRouter, Navigate, Route
import './App.css';
import TopNavBar from './components/Top
import CustomerListPage from './pages/C
import { customerListStore } from './re

const store = customerListStore();

const App: React.FC = () => {
  return (
    <Provider store={store}>
      <BrowserRouter>
```


TS customerListStore.ts ×
src > redux > store > TS customerListStore.ts > customerListStore

```
import { createStore, combineReducers, applyMiddleware } from 'redux';
import thunkMiddleware from 'redux-thunk';
import { filtersReducer } from '../reducers/filtersReducer';
import { customerReducer } from '../reducers/customerReducer';
import { composeWithDevTools } from 'redux-devtools-extension';

const composedEnhancer = composeWithDevTools(applyMiddleware(thunkMiddlewa

export const customerListStore = () => {
  const store = createStore(
    combineReducers({
      customers: customerReducer,
      filters: filtersReducer
    }),
    composedEnhancer
  );
  return store;
};
```

© 2021 Cognizant