```
import React, { useState } from "react";
// import our data from the Javascript file.
// JSON object representing Spotify streams to August 2023
import { spotify2023 } from "./spotify2023";

// Writing our own functions for use in our app
// Writing a comment about a specific number.
function writeNumberAsWords(n) {
  let nAsStr = n.toString(10); // our number is in base 10
  let digits = nAsStr.length; // number of digits
  let firstDigit = nAsStr.charAt(0);

  let message = "nothing";
  if (digits === 9) {
    message = "(At least " + firstDigit + " hundred million!)";
  } else if (digits === 10) {
    message = "(At least " + firstDigit + " billion!)";
  } else {
    message = "(Lots of streams)";
  }

  return message;
}

// Parent component - App
function App() {
  // lets keep the searchTerm as a state varaiable
  // searchTerm is given an initial value of empty string
  const [searchTerm, setSearchTerm] = useState("");

  // Here is our textbox handler function.
  // This handles the event that is fired when
  // the search form (text box) changes
  function onSearchFormChange(event) {
    // An event is generated by Javascript.
    // We use the hook setSearchTerm to safely assign
    // the current value in the textbox to searchTerm
    setSearchTerm(event.target.value);
  }

  return (
    <>
      <h1>Parent component CS385 Spotify Search</h1>
      <p>Your current search term is [{searchTerm}]</p>
      <form>
        <h3>Type your search here: </h3>
        <input onChange={onSearchFormChange} type="text" />
      </form>
      <hr />
      <ResultsComponent
        searchTermFromParent={searchTerm}
        spotifyArrayFromParent={spotify2023}
      />
    </>
  );
}

// This is the child component. It is used to display the results
// of the searches conducted. The parent must provide the searchTerm
// and access to the array of JSON objects.

function ResultsComponent(props) {
  // Within this component or function we create our
  // filter function. It will be needed to conduct
  // the search of the array of JSON objects.

  function spotifyFilterFunction(searchTerm) {
    return function (spotifyObject) {
      // convert everything to lower case for string matching
      let artist = spotifyObject.artist.toLowerCase();
      let track = spotifyObject.track.toLowerCase();
      return (
        searchTerm !== "" &&
        (track.includes(searchTerm.toLowerCase()) ||
          artist.includes(searchTerm.toLowerCase()))
      );
    };
  }
  // We can use the filter function to tell us how many search results
  // we have. We find the length of the filtered array

  let numberResults = props.spotifyArrayFromParent.filter(
    spotifyFilterFunction(props.searchTermFromParent)
  ).length;

  return (
    <>
      <h1>Child Component: Search Results</h1>
      <h2>There are {numberResults} search results </h2>
      {numberResults === 0 && <p>No results</p>}
      {numberResults > 0 && numberResults < 10 && <p>Some results, not many</p>}
      {numberResults > 10 && <p>Lots of results</p>}
      {props.spotifyArrayFromParent
        .filter(spotifyFilterFunction(props.searchTermFromParent))
        .map((a, index) => (
          <p key={index}>
            <b>{a.artist}</b>, <i>{a.track}</i> Streams: {a.streams}{" "}
            {writeNumberAsWords(a.streams)}
          </p>
        ))}
    </>
  );
```

```
} // end of child component for results.
export default App;
```