Solution Review: Avoiding Duplicates

Learn to avoid showing duplicated search.

We'll cover the following

Solution:

The source of the five rendered buttons is the `getLastSearches` function. There, we take the array of `urls` and return the last five entries from it. Now we'll change this utility function to return the last six entries instead of five, removing the last one. Afterward, only the five *previous* searches are displayed as buttons.

```
const getLastSearches = urls =>
  urls
    .slice(-6)
    .slice(0, -1)
    .map(extractSearchTerm);
```

src/App.js

If the same search is executed twice or more times in a row, duplicate buttons appear, which is likely not your desired behavior. It would be acceptable to group identical searches into one button if they followed each other. We will solve this problem in the utility function as well. Before separating the array into the five previous searches, group the identical searches:

```
const getLastSearches = urls =>
  urls
    .reduce((result, url, index) => {
      const searchTerm = extractSearchTerm(url);

      if (index === 0) {
        return result.concat(searchTerm);
      }

      const previousSearchTerm = result[result.length - 1];

      if (searchTerm === previousSearchTerm) {
        return result;
      } else {
        return result.concat(searchTerm);
      }
    }, [])
    .slice(-6)
    .slice(0, -1);
```

src/App.js

The reduce function starts with an empty array as its `result`. The first iteration concats the `searchTerm` we extracted from the first `url` into the `result`. Every extracted `searchTerm` is compared to the one before it. If the previous search term is different from the current, concat the `searchTerm` to the result. If the search terms are identical, return the result without adding anything.

Lastly, the SearchForm's input field should be set with the new `searchTerm` if one of the last search buttons is clicked. We can solve this using the state updater function for the specific value used in the SearchForm component.

```
const App = () => {
  ...

  const handleLastSearch = searchTerm => {
    setSearchTerm(searchTerm);

    handleSearch(searchTerm);
  };

  ...
};
```

src/App.js

Last, extract the feature's new rendered content from this section as a standalone component, to keep the App component lightweight:

```
const App = () => {
  ...

  const lastSearches = getLastSearches(urls);

  return (
    <div>
      ...
      <LastSearches
        lastSearches={lastSearches}
        onLastSearch={handleLastSearch}
      />

      ...
    </div>
  );
};
const LastSearches = ({ lastSearches, onLastSearch }) => (

  <>
    {lastSearches.map((searchTerm, index) => (
      <button
        key={searchTerm + index}
        type="button"
        onClick={() => onLastSearch(searchTerm)}
      >
        {searchTerm}
      </button>
    ))}
  </>
);
```

src/App.js

The complete demonstration of the above concepts:

```
  □ □ □□  □   ã□  F   □□  □   □  )□     □   9□  5□  @@  □   °□  n□  □PNG
□
   IHDR   □   □□□   (-□S   äPLTE"""""""""""""""""2PX=r□)7;*:>H□¤-BGE□□8do5Xb6[eK□®K□¯1MU9gs3S
□
   IHDR   □   □□□   ×©ÍÊ   □ePLTE"""""""""""""""""""""""2RZN¢¹J□«3R[J□¬)59YÁÞ0KS4W`Q«ÄL□²%+-0JR
?^q÷ñíÛ□ï.},□ìsæÝ_TttÔ¾ □1#□□/(ì□-[□□□è`□è`Ì□ÚïÅðZ□d5□□□□?ÎebZ¿Þ□i.Ûæ□□□iqÎ□+1°□}Â□5ù  ïçd
□
   IHDR       □□   D¤□Æ  □APLTE   """"""""""""""""""""""""""2RZVºÖ_ÔôU·Ñ=r□$()'25]ÎíC□□0LS<o}X
□
   IHDR   @   @□□   □·□ì  □:PLTE   """""""""""""""""""""""""""""""""""""""""""""""""""""""""""
¢ßqÇ8Ù□´□mKË±mÆ¶mÛü·yi!è□ÎªYÏuë ÀÏ_Àï?i÷□ý+ò□□ÄA□|□ù{□□´?¿□_En□).□JËD¤<□
©¬¢Z\Ts©R*□(□  ¯©□J□□□□u□X/□4J□9□¡5·DEµ4kÇ4□&i¥V4Ú□¡®Ð□□¯□vsf:àg,□¢èBC»î$¶□ºÍùî□□á□@□ô□I_
-ê>Û□º«¢XÕ¢î}ß¨ëÛÑ;□ÃöN´□ØvÅý□Î¸ÿ1 □ë×ÄO@&v/Äþ_□ö\ô□Ç\í.□□¼+0□□;□□□!□fÊ□¦´Ó%Â JY·O□Â□'/Å]_□
```

This feature wasn’t an easy one. Lots of fundamental React but also JavaScript knowledge was needed to accomplish it. If you had no problems implementing it yourself or to follow the instructions, you are very well set. If you had one or the other issue, don’t worry too much about it. Maybe you even figured out another way to solve this task and it may have turned out simpler than the one I showed here.

Exercises:

- Confirm the changes from the last section.