Steps and features implemented :

four state variables are created

1. For storing the user searched names through input field
2. Fetching the API searchable user api and display that as suggestion below the input field
3. For storing the user selected names (user selected objects)
4. For storing the already selected user name unique id, for that Set is used,

(already selected user name stored in set and which all user names are not there in set those only displayed below as suggestions, this feature achieved using !set.has(id) and else displayed <></>

* Fetched the user Selection Api and stored in one variable and mapped that as scrollable list
* After selecting the user name from scrollable list those are stored in one more state variable and those are mapped using map function and Pills are created
* For deleting pills filter used
* After deleting pills Set should also updated
* After adding and deleting auto focus is implemented using useRef hook with focus() method

Other features

1. Entering “Backspace” key should delete the last pills (last index of selected array)
2. Pressing “ArrowUp” and “ArrowDown” should go up and down through suggestion list
3. By pressing Enter key in input field should select the active userName

(arrow up ,down and enter functionality not working properly)

const handleKeyDown = (e) => {

    if (

      e.key === "Backspace" &&

      e.target.value === "" &&

      selectedOptions.length > 0

    ) {

      // selectionOptions.pop(); =====> this method also works

      setSuggestion([]);

      const lastIndex = selectedOptions[selectedOptions.length - 1];

      handleRemove(lastIndex);

    } else if (e.key === "ArrowDown" && suggestions.length > 0) {

      e.preventDefault();

      setSuggestionUpDown((preIndex) =>

        preIndex < suggestions.length - 1 ? preIndex + 1 : preIndex

      );

    } else if (e.key === "ArrowUp" && suggestions.length > 0) {

      e.preventDefault();

      setSuggestionUpDown((preIndex) =>

        preIndex > 0 ? preIndex - 1 : preIndex

      );

    } else if (

      e.key === "Enter" &&

      suggestionUpDown >= 0 &&

      suggestionUpDown < suggestions.length

    ) {

      handleSelectOption(suggestions[suggestionUpDown]);

    }

  };





