```jsx
import React, { useState } from 'react';

import './App.css';

import List from './List';


function App() {
  const [isAuthenticated, setIsAuthenticated] = useState(false);


  const login = () => {
    setIsAuthenticated(true);
  };


  return (
    <div className="App">
      {!isAuthenticated ? (
        <div className="login">
          <h2>Login</h2>
          <input type="text" placeholder="Username" />
          <input type="password" placeholder="Password" />
          <button onClick={login}>Login</button>
        </div>
      ) : (
        <List />
      )}
    </div>
  );
```

```
}


export default App;

import React, { useState, useEffect } from 'react';

import './List.css';

import ListItem from './ListItem';


const List = () => {

  const [contacts, setContacts] = useState([]);

  const [visibleContacts, setVisibleContacts] = useState([]);

  const [isLoading, setIsLoading] = useState(false);


  useEffect(() => {

    fetch('https://randomuser.me/api/?results=500')

      .then((res) => res.json())

      .then((data) => {

        setContacts(data.results);

        setVisibleContacts(data.results.slice(0, 20));

      });

  }, []);


  const handleScroll = () => {

    const scrollTop = document.documentElement.scrollTop;

    const windowHeight = window.innerHeight;

    const fullHeight = document.documentElement.scrollHeight;
```

```javascript
    if (scrollTop + windowHeight === fullHeight && !isLoading) {

      setIsLoading(true);

      setTimeout(() => {

        setVisibleContacts((prevVisibleContacts) => [

          ...prevVisibleContacts,

          ...contacts.slice(

            prevVisibleContacts.length,

            prevVisibleContacts.length + 20

          ),

        ]);

        setIsLoading(false);

      }, 1000);

    }

  };


  useEffect(() => {

    window.addEventListener('scroll', handleScroll);


    return () => window.removeEventListener('scroll', handleScroll);

  }, []);


  const handleLogout = () => {

    window.location.reload();

  };
```

```jsx
  return (
    <div>
      <button className="logout" onClick={handleLogout}>
        Logout
      </button>
      <div className="list">
        {visibleContacts.map((contact, index) => (
          <ListItem key={index} contact={contact} />
        ))}
        {isLoading && <div className="loading">Loading...</div>}
      </div>
    </div>
  );
};


export default List;
import React from 'react';


const ListItem = ({ contact }) => {
  return (
    <div className="list-item">
      <img src={contact.picture.large} alt={contact.name.first} />
      <div>
        <h2>{contact.name.first + ' ' + contact
```