• Name: Afreen Ahmed

• Enrollment Number: Mern@003

• Assignment: User Directory with useEffect and Component Reusability

---

1. Program Statement:

This React program fetches random user data from an external API (randomuser.me) and displays each user's information using a reusable UserCard component. It utilizes React hooks like useState and useEffect for state management and side effects. The user cards are displayed in a responsive grid layout, showing profile pictures, names, emails, and locations.

---

2. Algorithm:

1. Start
2. Create a functional component UseDirectory
3. Initialize a state variable users with an empty array
4. Use useEffect to fetch data from the API when the component mounts
5. Store the fetched user data in the users state
6. Create a reusable UserCard component to display individual user data
7. Use .map() to render a grid of UserCard components
8. Export both components
9. End

---

3. Pseudocode:

Component: UseDirectory

- Initialize users = []
- On component mount (useEffect),
    - Fetch user data from randomuser.me
    - Parse JSON and store results in users
- Render:
    - Heading "Profiles"
    - Grid layout of UserCard for each user

Component: UserCard

- Accept users as props
- Render:
    - User image

- Full name
- Email
- City and country

---

4. Program Code (React):

UseDirectory.js

jsx

CopyEdit

```
import React, { useEffect, useState } from 'react';
import UserCard from './UserCard';

const UseDirectory = () => {
  const [users, setusers] = useState([]);

  useEffect(() => {
    fetch("https://randomuser.me/api/?results=12")
      .then((res) => res.json())
      .then((data) => setusers(data.results));
  }, []);

  return (
    <div className="p-6">
      <h1 className="text-3xl font-bold text-center mb-6">Profiles</h1>
      <div className="grid grid-cols-1 sm:grid-cols-2 md:grid-cols-3 lg:grid-cols-4 gap-6">
        {users.map((user, index) => (
          <UserCard key={index} users={user} />
        ))}
      </div>
    </div>
  );
};

export default UseDirectory;
```

UserCard.js

jsx

CopyEdit

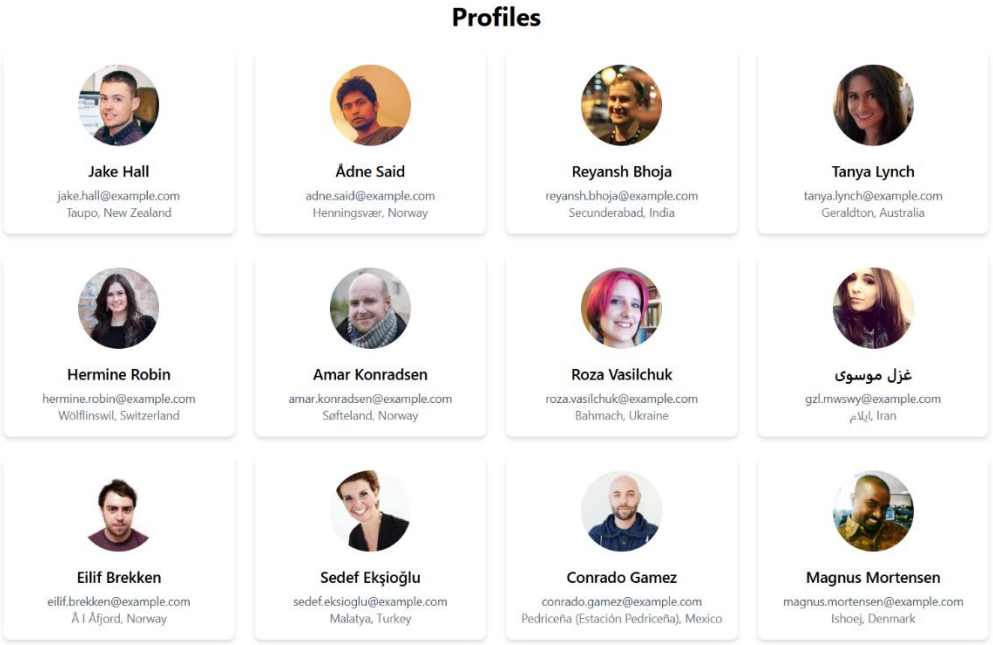
```
import React from 'react';

const UserCard = ({ users }) => {
  return (
    <div className="bg-white shadow-md rounded-lg p-4 hover:shadow-lg transition duration-300 text-center">
      <img
        src={users.picture.large}
        alt={`${users.name.first} ${users.name.last}`}
        className="w-24 h-24 rounded-full mx-auto mb-4"
      />
      <h2 className="text-lg font-semibold">
        {users.name.first} {users.name.last}
      </h2>
      <h3 className="text-sm text-gray-600 mt-1">{users.email}</h3>
      <h3 className="text-sm text-gray-500">
        {users.location.city}, {users.location.country}
      </h3>
    </div>
  );
};

export default UserCard;
```

---

6. Screenshot of Output:



---

7. Observation / Reflection:

This activity helped reinforce the usage of React hooks (useState and useEffect) to manage asynchronous data fetching. It also demonstrated how to build reusable components like UserCard, making code more modular and maintainable. The project improved understanding of rendering lists using .map(), handling props, and styling components for responsiveness using CSS (e.g., Tailwind). It mimics real-world UI behavior seen in modern directory or dashboard apps.