

# React Query

---

## Installation

---

```
npm install @tanstack/react-query
```

## Fetchers

---

```
import axios from "axios";

export const fetchUsers = async () => {
  console.log("Fetching users");
  const response = await axios.get("http://localhost:3000/users");
  const users = response.data;
  console.log("Users :", users);
  return users;
}
```

## Utilisation

---

```
import { fetchUsers } from "...";

export function WithReactQuery() {
  console.log("Render");
  const { isError, isSuccess, isLoading, data, error } = useQuery(
    ["users"],
    fetchUsers,
    { staleTime: 60000 }
  );

  // Not necessary of course...
  useEffect(() => {
    console.log("Component mounted")
  }, []);

  if (isLoading) {
    console.log("Loading...");
    return <div>Loading...</div>
  }

  if (isError) {
    console.log("Error...");
  }
}
```

```

        return <div>Error...</div>
    }

    return (
        <div>
            { data &&
                data.map((user) => <User key={user.id} user={user}
/>)
            }
        </div>
    );
}

```

## useMutation

---

```

const queryClient = useQueryClient();

const usersMutation = useMutation(
    (newUser) => {
        console.log("Run: ", newUser);
        return axios.post("http://localhost:3000/users", newUser);
    },
    {
        onSuccess: () => {
            queryClient.invalidateQueries(["users"]);
            queryClient.refetchQueries(["users"], { force: true });
        },
        onError: () => {...}
    }
);

const onCreateUser = (event) => {
    event.preventDefault();
    usersMutation.mutate({ name: "Quentin", age: 29 });
}

if (usersMutation.isLoading) {
    return <div>Loading...</div>
}

```