





GRAPHIQL

QUERIES (2)

MUTATIONS (0)

CACHE

Search queries

ROOT\_QUERY

Country:1012

Country:1028

Country:1046

Country:105

Country:1066

Country:1083

Country:1105

Country:1123

Country:1141

Country:1160

Country:1179

Country:1197

Country:1219

Country:1239

Country:1259

Country:1281

Country:129

Country:1299

Country:1317

Country:1334

Country:1351

Country:1368

Country:1387

ROOT\_QUERY CACHE ID

ROOT\_QUERY

\_\_typename: Query

▶ Country({}): [] 250 items







GRAPHIQL

QUERIES (2)

MUTATIONS (0)

CACHE

Search queries

ROOT\_QUERY

Country:1012

Country:1028

Country:1046

Country:105

Country:1066

Country:1083

Country:1105

Country:1123

Country:1141

Country:1160

Country:1179

Country:1197

Country:1219

Country:1239

Country:1259

Country:1281

Country:129

Country:1299

Country:1317

Country:1334

Country:1351

Country:1368

Country:1387

ROOT\_QUERY CACHE ID

ROOT\_QUERY

\_\_typename: Query

▶ Country({}): [] 250 items



Search queries

ROOT\_QUERY

Country:1012

Country:1028

Country:1046

Country:105

Country:1066

Country:1083

Country:1105

Country:1123

Country:1141

Country:1160

Country:1179

Country:1197

Country:1219

Country:1239

Country:1259

Country:1281

Country:129

Country:1299

Country:1317

Country:1334

Country:1351

Country:1368

Country:1387

ROOT\_QUERY  
 \_\_typename: Query  
 ► Country({}): [] 250 items

A

GRAPHIQL

QUERIES (2)

MUTATIONS (0)

CACHE

Q

Search queries

ROOT\_QUERY

Country:1012

Country:1028

Country:1046

Country:105

Country:1066

Country:1083

Country:1105

Country:1123

Country:1141

Country:1160

Country:1179

Country:1197

Country:1219

Country:1239

Country:1259

Country:1281

Country:129

Country:1299

Country:1317

Country:1334

Country:1351

Country:1368

Country:1387

ROOT\_QUERY

CACHE ID

ROOT\_QUERY

\_\_typename: Query

▶ Country({}): [] 250 items

# Apollo Client / React / useQuery

```
1 const REPOS = gql`
2   query GetRepos {
3     repos {
4       id
5       name
6       author {
7         id
8         name
9       }
10    }
11  }
12 `;
13
14 function Repos() {
15   const { loading, error, data } = useQuery(REPOS);
16
17   if (loading) return <p>Loading...</p>;
18   if (error) return <p>Error :( </p>;
19
20   return (
21     <ul>
22       {data.repos.map(({ id, name, author }) => (
23         <li key={id}>
24           {name} by {author.name}
25         </li>
26       ))}
27     </ul>
28   );
29 }
```