

**.graphqlrc** X

```
1  overwrite: true
2  schema: schema.graphql
3  documents: ./src/**/*.tsx
4  extensions:
5    codegen:
6      generates:
7        src/:
8          config:
9            dedupeOperationSuffix: true
10           documentMode: graphQLTag
11           nonOptionalTypename: true
12           omitOperationSuffix: true
13           preResolveTypes: true?
14           withHooks: true
15         plugins:
16           - typescript-operations
17           - typescript-react-apollo
18         preset: near-operation-file
19         presetConfig:
20           baseTypesPath: __generated__/types.ts
21           extension: .ts
22           folder: __generated__
23         src/__generated__/types.ts:
24           config:
25             namingConvention: keep
26           plugins:
27             - typescript
```

## CountryList.ts X

```
85  /**
86   * __useCountryListQuery__
87   *
88   * To run a query within a React component, call `useCountryListQuery` and pass it any options that fit your needs.
89   * When your component renders, `useCountryListQuery` returns an object from Apollo Client that contains loading,
90   * error, and data properties
91   * you can use to render your UI.
92   *
93   * @param baseOptions options that will be passed into the query, supported options are listed on: https://www.apollographql.com/docs/react/api/react-hooks/#options;
94   *
95   * @example
96   * const { data, loading, error } = useCountryListQuery({
97   *   variables: {
98   *   },
99   * });
100  */
101  export function useCountryListQuery(
102    baseOptions?: Apollo.QueryHookOptions<
103      CountryListQuery,
104      CountryListQueryVariables
105    >,
106  ) {
107    const options = { ...defaultOptions, ...baseOptions };
108    return Apollo.useQuery<CountryListQuery, CountryListQueryVariables>(
109      CountryListDocument,
110      options,
111    );
112  }
113  export function useCountryListLazyQuery(
114    baseOptions?: Apollo.LazyQueryHookOptions<
115      CountryListQuery,
116      CountryListQueryVariables
117    >,
118  ) {
119    const options = { ...defaultOptions, ...baseOptions };
120    return Apollo.useLazyQuery<CountryListQuery, CountryListQueryVariables>(
121      CountryListDocument,
122      options,
123    );
124  }
```