

```
USEQUETY

A React Apollo's useQuery Hook
```

```
(optional) options to
                                                                                                                                     configure the
                                                                                                                                    behaviour of the
             object returned from query that contains details of query result and additional functions (e.g. refetch)
                                                                                                type of variables the query is to be
                                                                                                                                   component making
                                                                                                                                    the query (e.q.
                                                                                                    executed with
                                                                                                                                  passing in variables)
const { data, loading, error } = useQuery<TData, TVariables>(query,options);
                                                                                  type of data
                                                                                                                        graphal AST
                                           runtime error
                              boolean
                object
                                                                                expected to be
                                             of query
                             indicating
                                                                                                                           query
              containing
                                                                                 returned from
                           whether query
                                                                                                                         document
                                              request
              results of
                             is in flight
                                                                                                                        constructed
                                                                                     query
                query
                                                                                                                      with the agl tag
```

## Run a query when component renders

```
import React from "react";
import gql from "apollo-boost";
                                                                                   import gal tag from apollo-boost
                                                                                   import useQuery hook from react apollo
import { useQuery } from "@apollo/react-hooks";
import
   Superhero as SuperheroData,
                                                                                    import autogenerated type
                                                                                  definitions (data + variables) for
   SuperheroVariables
                                                                                           the query
} from "./__generated__/Superhero/";
const SUPERHERO = gql'
  query Superhero {
                                                                                    create GraphQL query document with the gal tag
     superhero {
         name
export const FunctionComponent = () => {
                                                                                  pass the relevant type definitions and query document to the useQuery hook
  const { data, loading, error } = useQuery
      SuperheroData,
      SuperheroVariables
                                                                                    return the data, loading, and
                                                                                    error results of the query
   >(SUPERHERO);
   if (loading) {
      return
         <div>
                                                                             when loading is true (i.e. query is in flight), component presents a loading message
           <h2>Loading!</h2>
         </div>
  if (error) {
      return
         <div>
                                                                                  when error is true (i.e. query returns an error), component
           <h2>Error!</h2>
                                                                                    presents an error message
         </div>
   return
      <div>
                                                                                  when query is complete and no error exists, component returns
        <h2>{data.superhero.name}</h2>
                                                                                     the intended ui containing
                                                                                   information from query data
      </div>
```