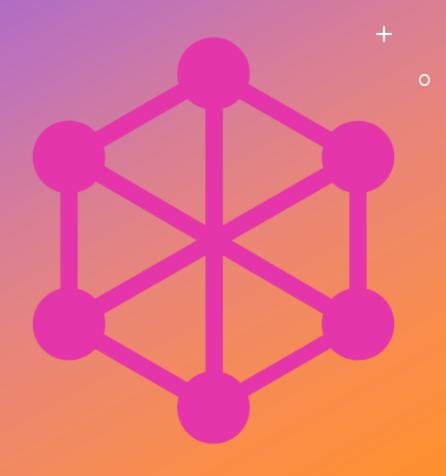
MAKING QUERIES FROM REACT

In this GraphQL tutorial I'll explain how we can make a query to our GraphQL server from one of our React components.



```
BookList.js
```

```
import React from "react";
import { useQuery, gql } from "@apollo/client";
const getBooksQuery = gql`
       books {
          name
          id
function BookList() {
   const {loading, error, data} = useQuery(getBooksQuery)
   if (data) {
       console.log("data: ", data);
   if (loading) return Loading...;
   if (error) return Error :(;
   return (
       Book name: 
       );
export default BookList;
```

Cors Setup in the 'server' part

Open new PowerShell

In the Terminal write nest command:

• npm i cors

```
app.js (server)
const express = require('express');
const { graphqlHTTP } = require('express-graphql');
const schema = require('./schema/schema');
const mongoose = require('mongoose');
const cors = require('cors')
                                                                                 Insert
                                                                                 code
const app = express();
app.use(cors());
mongoose.connect("mongodb+srv://marizza:demo8888@cluster0.9h62k.mongodb.net/gql")
const con = mongoose.connection
con.on('open', () => {
    console.log('connected...')
})
// bind express with graphql
app.use('/graphql', graphqlHTTP({
    schema,
    graphiql: true
}));
app.listen(4000, () => {
    console.log('now listening for requests on port 4000');
});
```

In the Terminal (server part) write next command:

nodemon app

In the Terminal (client part) write next command:

npm start

