

My First Query

In this lesson, we actually get to start seeing how GraphQL works first hand by writing our first query.

WE'LL COVER THE FOLLOWING



- GraphQL Query with GitHub's GraphQL API
- GraphQL Objects

Before we start building full-fledged GraphQL applications, let's explore GraphQL with the tools we have installed in the previous chapter. You can either use GraphiQL or the GitHub's GraphQL Explorer. In the following lessons, you will learn about GraphQL's fundamentals by executing your first GraphQL queries and mutations. We will even further explore features such as pagination in the context of GitHub's GraphQL API.

GraphQL Query with GitHub's GraphQL API

Let's start interacting with the GitHub API using queries and mutations without React. This will allow you to use your GraphiQL application or GitHub's GraphQL Explorer to make GraphQL query requests to GitHub's API. Both tools should be authorized to make requests using a personal access token as explained in the previous chapter. On the left-hand side of your GraphiQL application, you can fill in GraphQL queries and mutations.



Add the following query to request data about yourself.

Environment Variables ^

Key:	Value:
REACT_APP_GITHUB...	Not Specified...
GITHUB_PERSONAL...	Not Specified...

```
{
  viewer {
    name
    url
  }
}
```

The `viewer` object can be used to request data about the currently authorized user. Since you are authorized by your personal access token, it should show data of your own account.

GraphQL Objects

The `viewer` is an object in GraphQL terms. Objects hold data about an entity and this data is accessed using a so-called `field` in GraphQL. Fields are used to ask for the specific properties of objects. For instance, the `viewer` object exposes a wide range of fields and in our query, it is using the fields `name` and `url`. In its most basic form, a query is just objects and fields, and objects are fields in themselves.

Once you run the query in GraphiQL, you should see output similar to the one below, where your name and URL are in the place of mine:

Environment Variables ^

Key:	Value:
REACT_APP_GITHUB...	Not Specified...
GITHUB_PERSONAL...	Not Specified...

```
{
  "data": {
    "viewer": {
      "name": "Robin Wieruch",
      "url": "https://github.com/rwieruch"
    }
  }
}
```

Congratulations, we have performed our first query to access fields from our own user data

own user data.

In the next lesson, we will see how to request data from a source other than yourself.