

MODULE 2

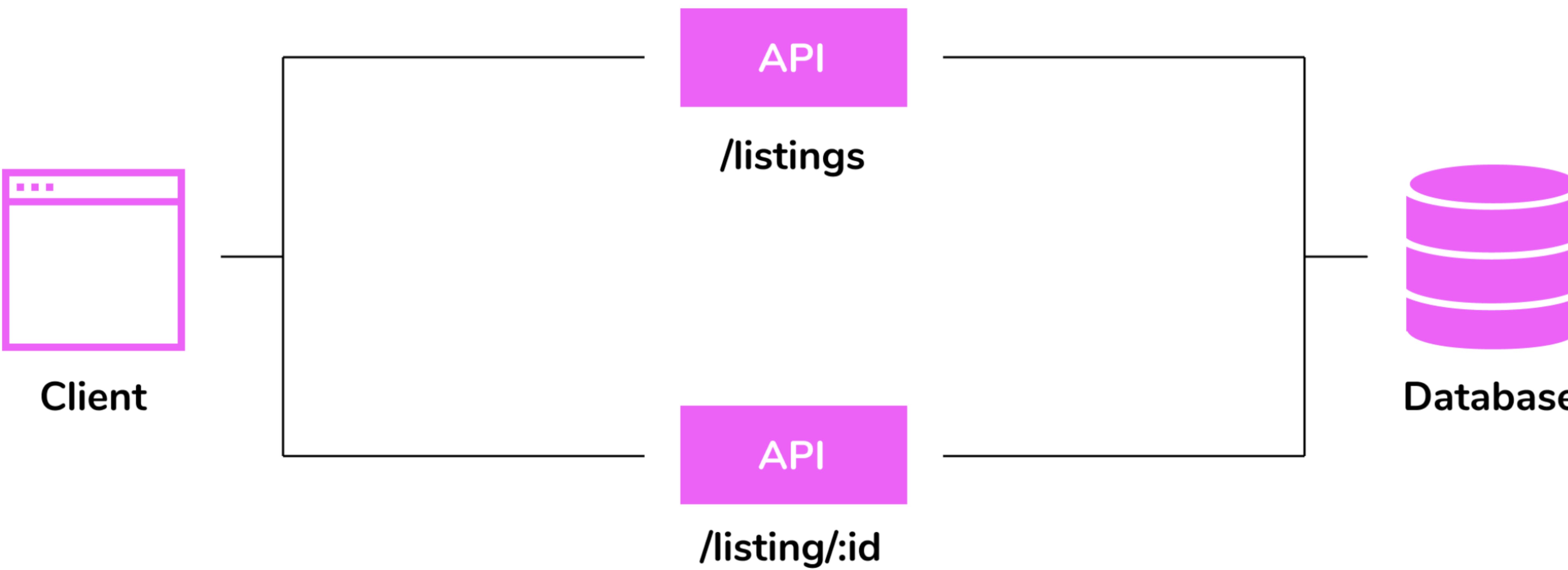
2.1 - WHAT IS GRAPHQL?

A FULLSTACK REACT MASTERCLASS

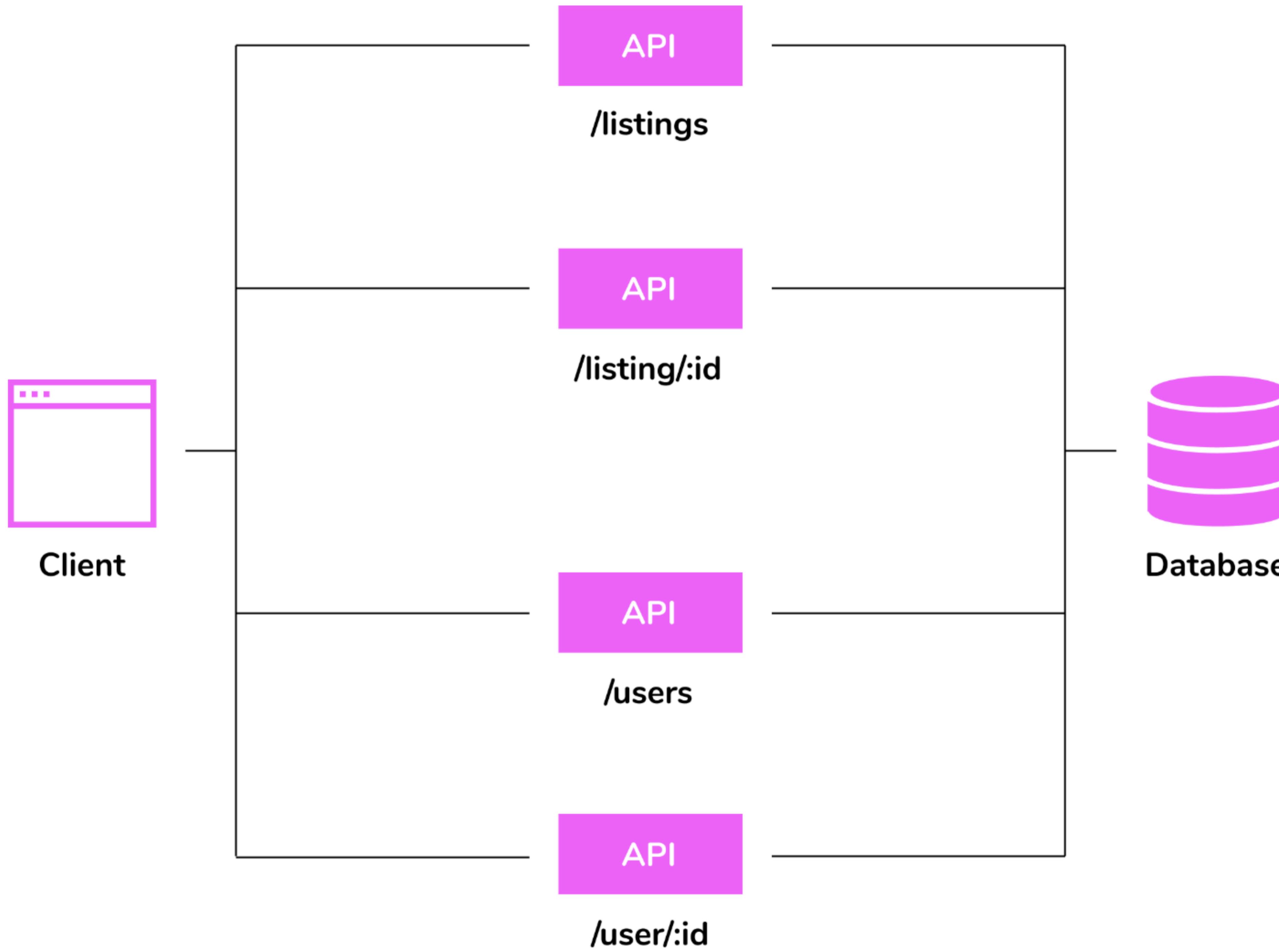
TINYHOUSE



Rest API



Rest API



Fetch user data and listings data for certain user

- `/user/:id` - to fetch certain user data
- `/user/:id/listings` - to fetch listings for certain user

GraphQL is a *query* language for APIs

```
query User($id: ID!) {  
  user(id: $id) {  
    id  
    name  
    listings {  
      id  
      title  
    }  
  }  
}
```



Get expected results

Describe our data

```
type User {  
  id: ID!  
  name: String!  
  listings: [Listing!]!  
}
```

```
{  
  "data": {  
    "user": {  
      "id": "11",  
      "name": "Hassan Djirdeh",  
      "listings": [  
        {  
          "id": "100",  
          "title": "Chic condo in downtown Toronto"  
        }  
      ]  
    }  
  }  
}
```



GraphQL APIs are...

- **Intuitive:** Client specifies *exactly* what data it needs.
- **Performant:** No useless data needs to be transferred.
- **Typed:** Every field in the schema is type-defined.

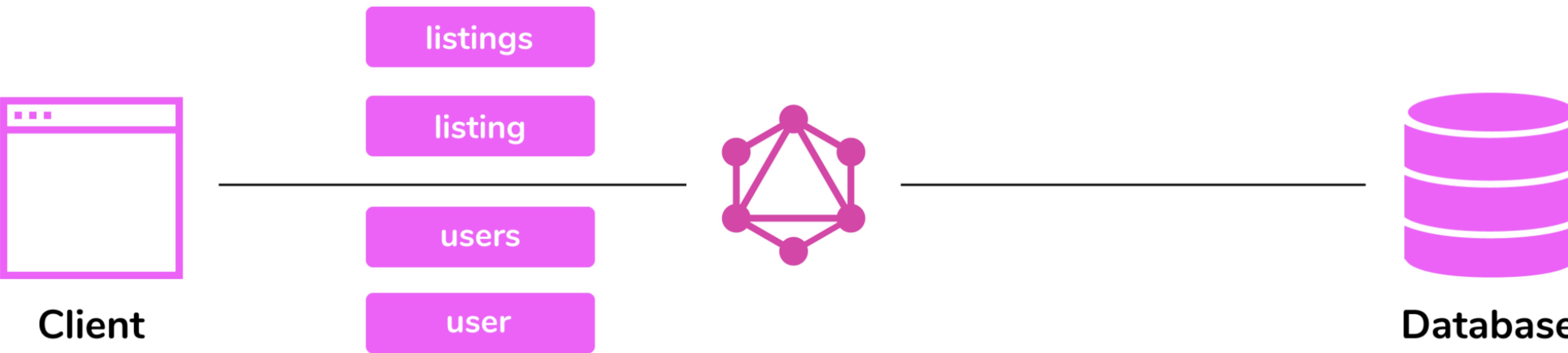


GraphQL APIs are also...

- *self documenting*
- encourage the use of GraphQL [IDEs](#)
- consist of a *single* endpoint



GraphQL API



GraphQL is *not* tied to any specific technology



Server Libraries

- Apollo Server | **Node Frameworks** - [Github](#)
- GraphQL Java | **Java** - [Github](#)
- GraphQL Ruby | **Ruby** - [Github](#)
- [...and a lot more](#)



Client Libraries

- Apollo Client | **React, Vue, etc.** - [Github](#)
- GraphQL Python | **Python** - [Github](#)
- GraphQL.Client | **.NET** - [Github](#)
- [...and a lot more](#)

