



Introduction to GraphQL

By Ilker Guller



Agenda

- Let's talk about REST first
- What is GraphQL?
- GraphQL Basic Concept
- Demo for showing how to write GraphQL in Node.js
- GraphQL in Details
- Libraries to use GraphQL
- Third-Party Services
- Demo with Third Party Services and RAN toolkit



Let's talk about REST first

- Easy to use and develop
- URI is resource name, HTTP Method is action
- Too many url for most of scenarios
- Many requests on one page
- Performance and usability limits
- Hard to implement customized data models
- Need to use different libraries for documentation, type definitions



Example REST

“Order History” Scenario

- /api/order/history
- /api/order/{OrderID}

“Add Item” Scenario

- /api/restaurant/menu/first-item (item-based link)
- /api/restaurant/ (submit)
- /api/getCart



What is GraphQL?

- New way of API
- It's not DB
- Query language for API
- Single endpoint (example: POST /graphql)
- Using strongly-typed static schema system
- No specific language or data source for using it
- Real-time Data Sync (with *subscription* feature)

Example GraphQL

```

1 {
2   getUser(id: 1) {
3     email
4     orders {
5       orderDetail {
6         priceTotal
7         priceSubtotal
8         createdAt
9         paymentMethod {
10          type
11        }
12        deliveryType {
13          type
14        }
15      }
16    }
17  }
18 }

```

```

{
  "data": {
    "getUser": {
      "email": "ilker.guller@deliveryhero.com",
      "orders": [
        {
          "orderDetail": {
            "priceTotal": 18.02,
            "priceSubtotal": 18.02,
            "createdAt": "Fri Nov 17 2017 17:36:12 GMT+0100 (CET)",
            "paymentMethod": {
              "type": "Cash"
            },
            "deliveryType": {
              "type": "D"
            }
          }
        },
        {
          "orderDetail": {
            "priceTotal": 28.3,
            "priceSubtotal": 28.3,
            "createdAt": "Fri Nov 17 2017 17:36:12 GMT+0100 (CET)",
            "paymentMethod": {
              "type": "Cash"
            },
            "deliveryType": {
              "type": "C"
            }
          }
        }
      ]
    }
  }
}

```

Example GraphQL

```
1 {  
2   getUser(id: 1) {  
3     email  
4     orders {  
5       orderDetail {  
6         priceTotal  
7         priceSubtotal  
8         createdAt  
9         paymentMethod {  
10          type  
11        }  
12        deliveryType {  
13          type  
14        }  
15      }  
16    }  
17  }  
18 }
```

```
{  
  "data": {  
    "getUser": {  
      "email": "ilker.guller@deliveryhero.com",  
      "orders": [  
        {  
          "orderDetail": {  
            "priceTotal": 18.02,  
            "priceSubtotal": 18.02,  
            "createdAt": "Fri Nov 17 2017 17:36:12 GMT+0100 (CET)",  
            "paymentMethod": {  
              "type": "Cash"  
            },  
            "deliveryType": {  
              "type": "D"  
            }  
          }  
        },  
        {  
          "orderDetail": {  
            "priceTotal": 28.3,  
            "priceSubtotal": 28.3,  
            "createdAt": "Fri Nov 17 2017 17:36:12 GMT+0100 (CET)",  
            "paymentMethod": {  
              "type": "Cash"  
            },  
            "deliveryType": {  
              "type": "C"  
            }  
          }  
        }  
      ]  
    }  
  }  
}
```

GraphQL Basic Concept

- **Queries** for read data

```
query {  
  getRestaurant(id: 1) {  
    name  
    address  
    image  
  }  
}
```

- **Mutations** for write & read data

```
mutation {  
  createUser(  
    email: "test@test.com",  
    password: "1111"  
  ) {  
    email  
    createdAt  
  }  
}
```

- **Subscriptions** for read data in real-time



@sly777



@the_bluescreen

Live Demo

for showing how to write GraphQL in Node.js

https://github.com/Sly777/introduction_to_graphql/blob/master/API



GraphQL in Details

- Interfaces
- Directives
- Inline Fragments
- Enumeration types
- Meta fields
- Union types
- etc...

GraphQL Libraries

(<https://github.com/chentsulin/awesome-graphql>)

- ***Apollo Client***: Most popular GraphQL Library for connecting to GraphQL server from your client (Web + IOS + Android)
- ***GraphQL Playground***: Advanced GraphQL IDE on browser
- ***GraphQL Voyager***: Graphical Browser for GraphQL Schemas
- ***RAN Toolkit***: Node.js + React + GraphQL toolkit for New Projects

GraphQL Services

- **GraphCMS:** GraphQL based Headless Content Management System
- **Graphcool:** Your own GraphQL backend in under 5 minutes. Works with every GraphQL client such as Relay and Apollo
- **Reindex:** Instant GraphQL Backend for Your React Apps
- **Scaphold:** GraphQL as a service that includes API integrations such as Stripe and Mailgun



@sly777



@the_bluescreen

Live Demo

with Third Party Services and RAN toolkit

https://github.com/Sly777/introduction_to_graphql/blob/master/client



@sly777



@the_bluescreen



Thanks!

Questions?

