1. GraphQL in Practice
2. Location to put graphql schema files

Graphical user interface, text, application

Description automatically generated

1. Create a graphql query file :query.graphqls. This file defines all the queries, available in graphql server . Here we are creating query for bankAccount

Graphical user interface, text, application

Description automatically generated

1. Now since BankAccount is not there it is shown as red. Lets go ahead and create Bankaccount type. In bankAccount.graphqls all the banking related queries will be there.
2. Bankaccoutn will have id, name, currency associated with it.

Graphical user interface, text, application

Description automatically generated

1. We will create another type Currency of type Enum , which lists all the currencies

Graphical user interface, application

Description automatically generated

1. Create our resolver file . Resolver is the file which serves the requests from the bankaccount
2. We Create BankAccountResolver and mark with @Component. This class will be implementing GraphQLQueryResolver and QueryResolver. The difference is : if it is a root query – GraphQLQueryResolver is needed. IF it is nested elements we use GraphQLResolver.
3. Inside BankAccountResolver, wei will add details to accept the BankAccount details and also we will create BankAccount POJO

Text

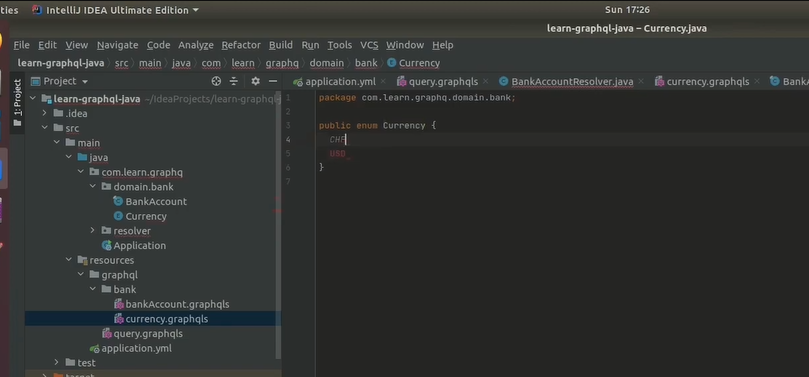
Description automatically generated

* Bank Account Class:

Text

Description automatically generated

* Currency Class:Currency and bankAccount should match the graphqls elements. Note that BankAccount class has @Builder annotation

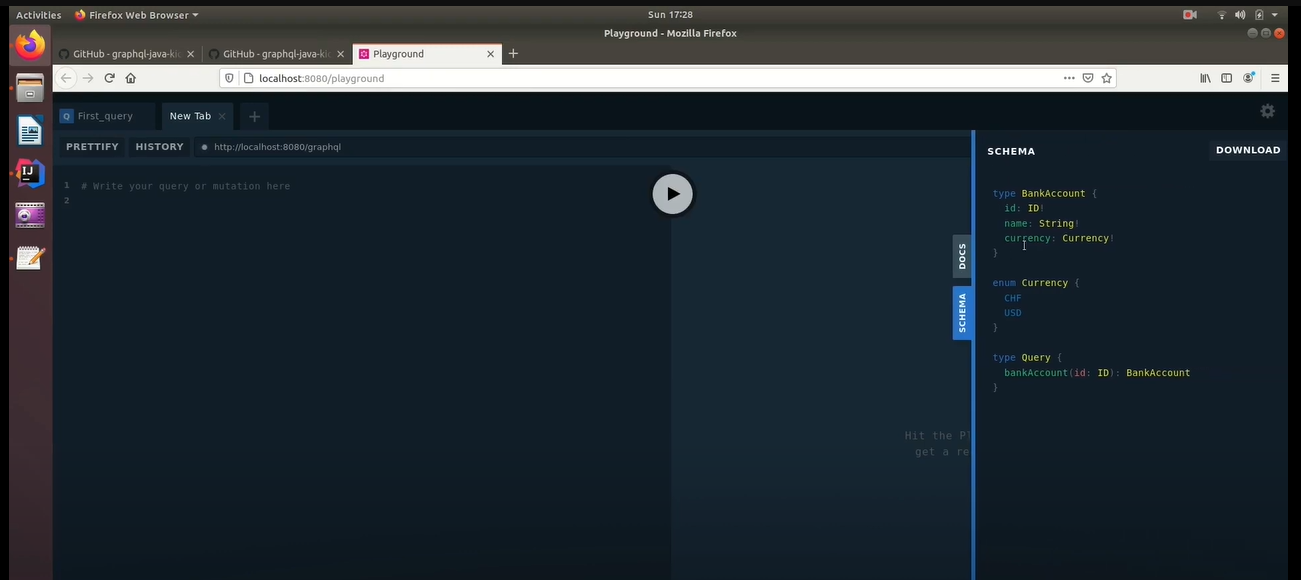


* Now we have the resolver which should resolve BankAccount
* Graphql.playground dependency helps : GraphQL Playground is **a graphical, interactive, in-browser GraphQL IDE**, created by Prisma and based on GraphiQL. In development, Apollo Server enables GraphQL Playground on the same URL as the GraphQL server itself (e.g.

Text

Description automatically generated

* Now if we run the application . we can see the schema in playground



* Docs of playground

Graphical user interface, application

Description automatically generated

* Now if we post the query: using UUID, we get the below response

Graphical user interface, text, application

Description automatically generated

* Now if we want only currency we can request for that also:

Graphical user interface, application

Description automatically generated

1. Max Depth Limit:

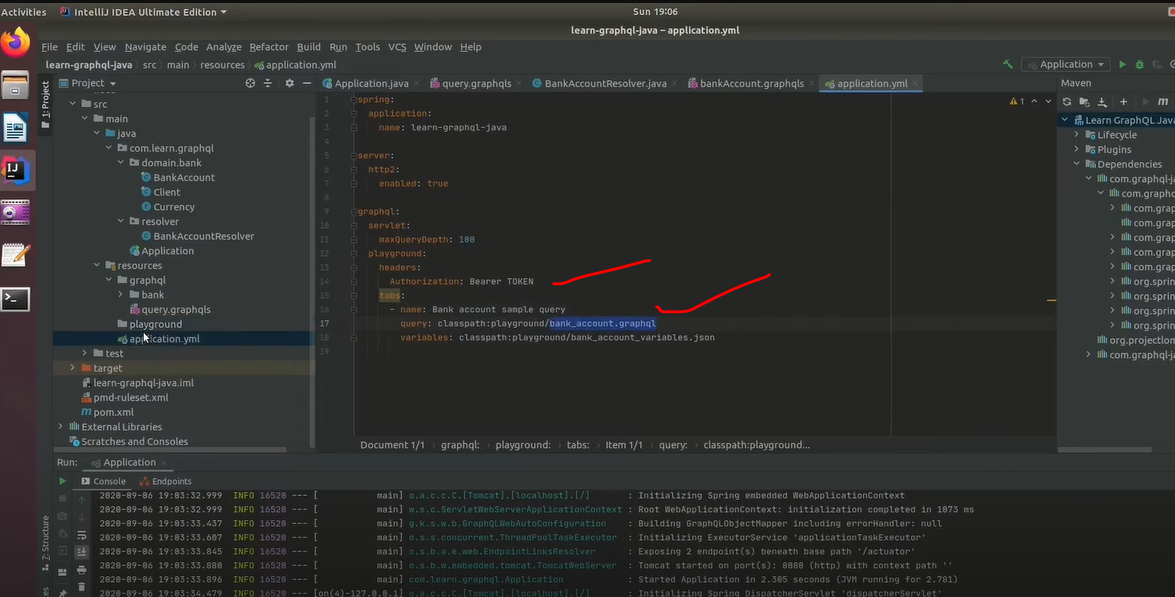
maxQueryDepth: 5

* It is the maximum depth you can have, when you are making a query.
* Even if you don’t have recursion better to set maxQueryDepth

Graphical user interface, text, application, Teams

Description automatically generated

1. **Specifying Auth:**



1. Method with argument and its variables:

Graphical user interface, text

Description automatically generated

Variables: Variables are in key value pairs

Text

Description automatically generated

* In the application:

Graphical user interface, application

Description automatically generated

1. **Voyager: Schema visualizer**

A picture containing text

Description automatically generated

Disabling Voyager:

Text

Description automatically generated