

## GRAPHQL :

### Cas de query :

Sur query on veut un service qui permet de consulter la liste des comptes : retourne une liste des bank account, utiliser une classe nommée BankAccountGraphQLController et injecter bankAccountRepository qui permet d'accéder à la bdd et faire l'injection via auowired ou constructeurs, puis créer methode qui permet de retourner la liste.

la methode : pour la configuration par défaut, on lui donne le même nom (accountList) sans parametres et retourner bankAccountRepository.findAll()

toutes les requettes graphql sont de type POST.

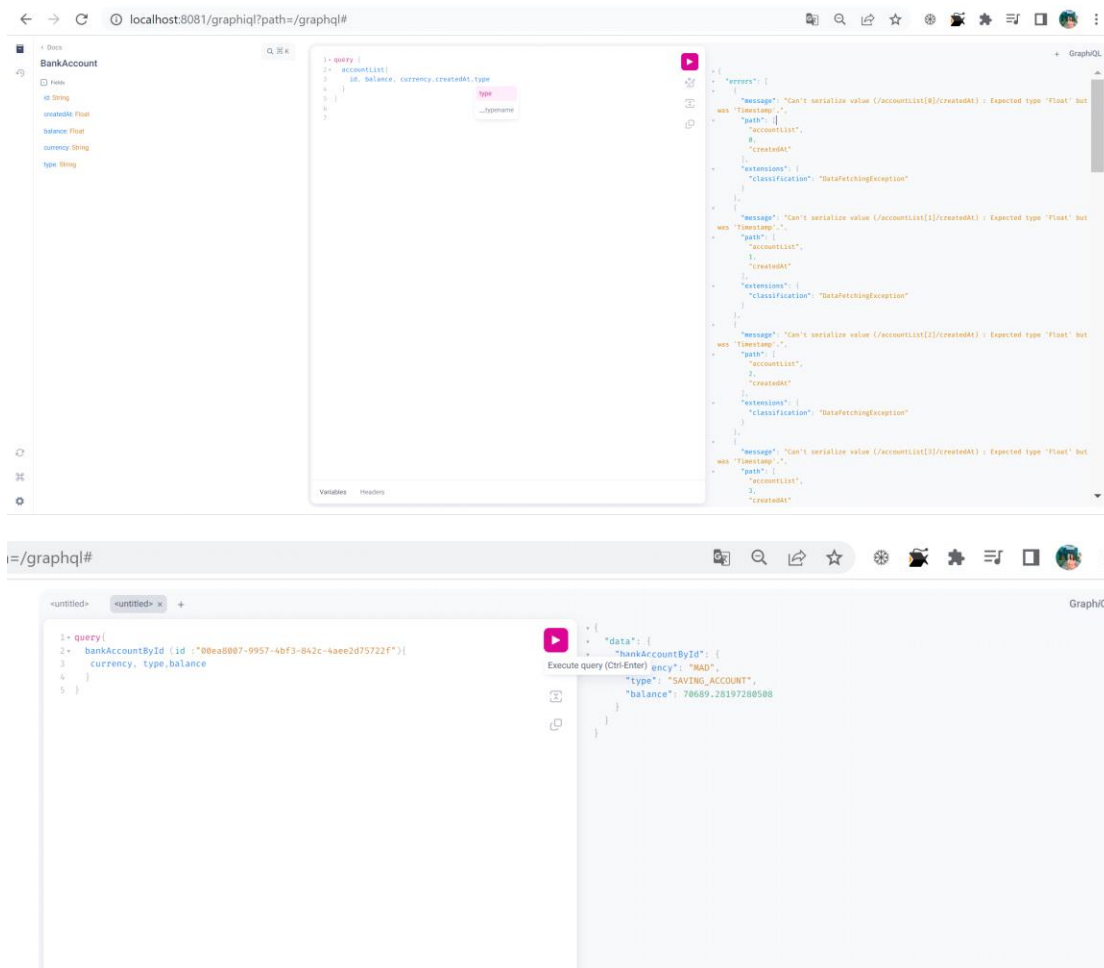
-créer un fichier schema.Graphqls qui contient le nom de liste des comptes à retourner et ses attributs, créer une classe BankAccountGraphQLController qui contient l'annotation queryMapping : spring boot graphql quand ona une requette de type query, automatiquement quand le client demande accountlist, il va executer la methode accountlilst (querymapping annotation)

on fait la recherche par id en ajoutant la methode bankAccountById qui a un id en parametre et retourne les infos sur le compte de l'id sinon, on génère une exception.

```
@Autowired
private BankAccountRepository bankAccountRepository;

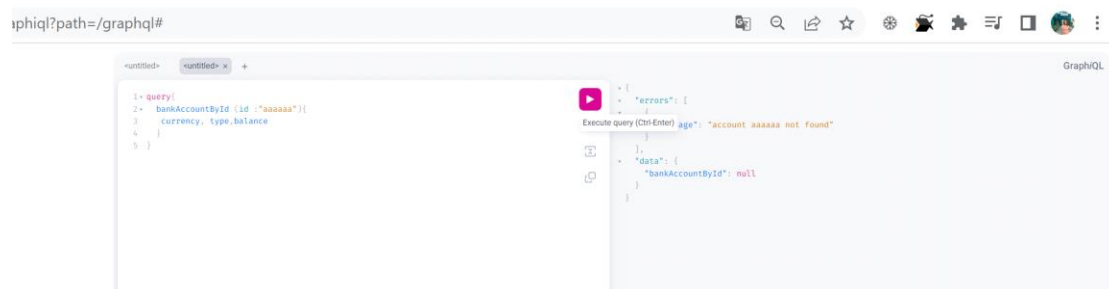
@QueryMapping
public List<BankAccount> accountList(){
    return bankAccountRepository.findAll();
}

@QueryMapping
public BankAccount bankAccountById(@Argument String id){
    return bankAccountRepository.findById(id)
        .orElseThrow(()->new RuntimeException(String.format("account %s not found",id)));
}
```

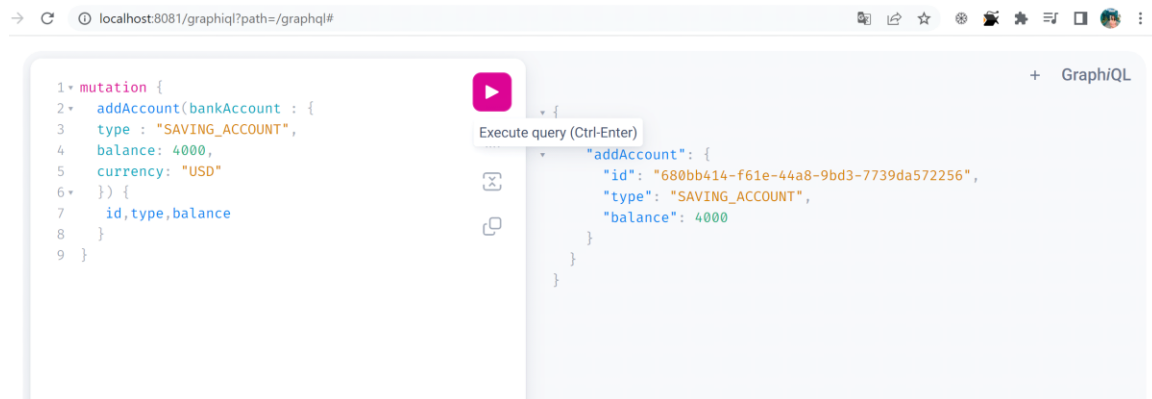


exception :

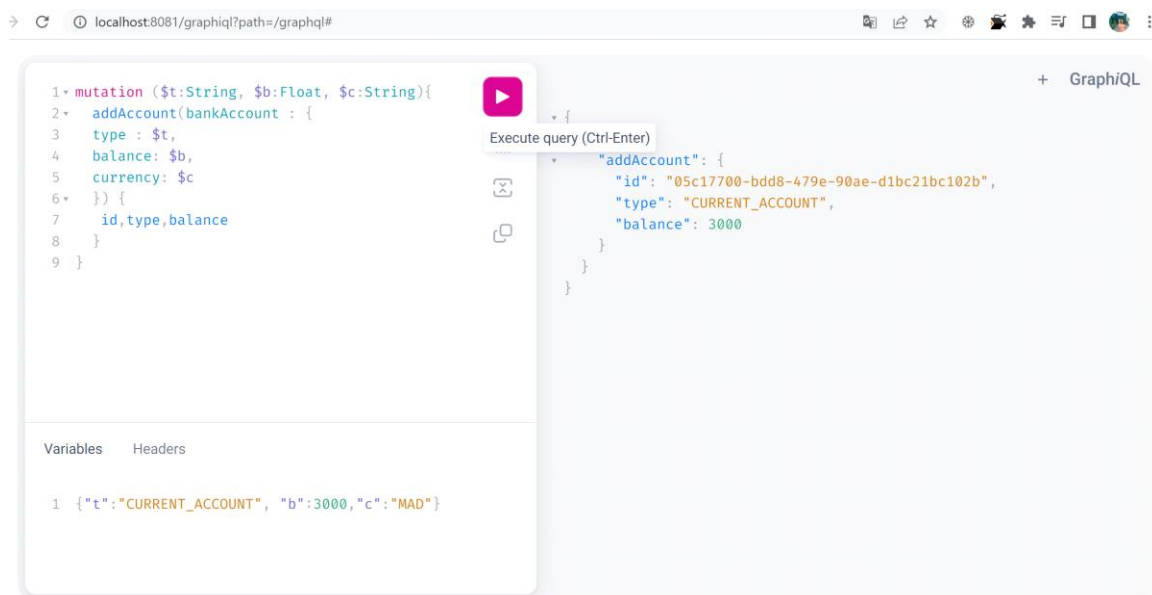
pour récupérer le msg d'exception, faut avoir un handler d'exception, on créer un package dans la gestion d'exception om on créer une classe pour handler d'exception, que en cas d'exception à l'aide de la methode getMessage il affiche sur GraphQL le message d'erreur.



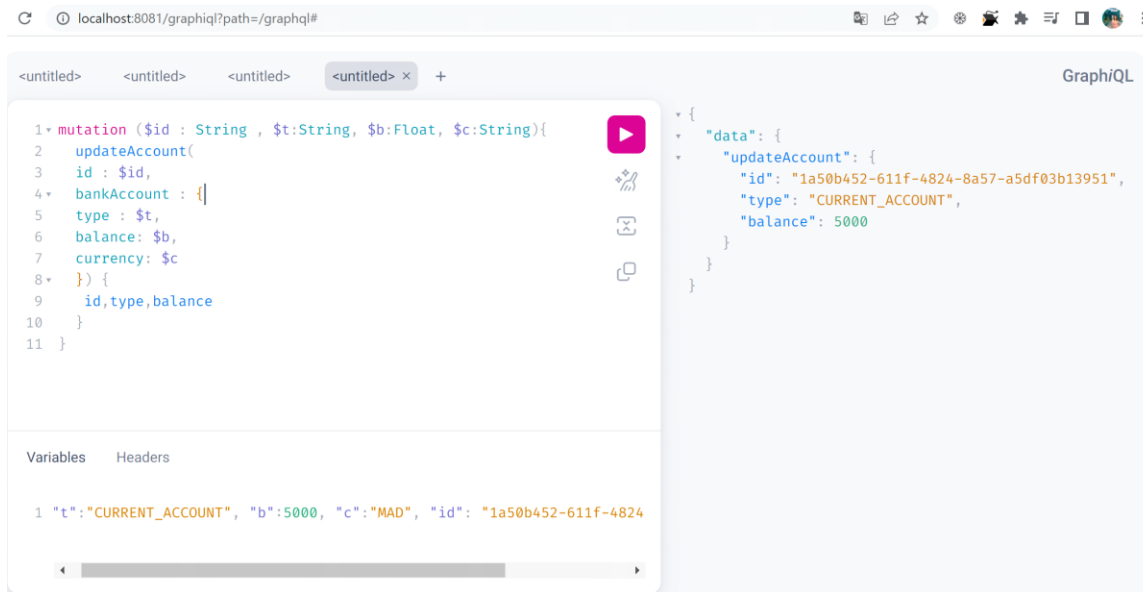
Cas de mutation



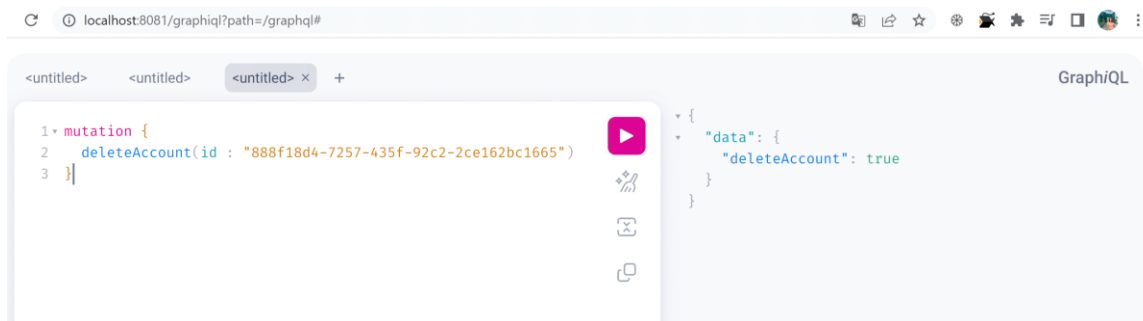
## Mutation par variable



## Update Mutation :



Delete mutation :



Création du customer :

localhost:8081/h2-console/lo...

Auto Max 1000 Auto complete Auto select

jdbc:h2:mem:account-db

BANK\_ACCOUNT

CUSTOMER

ID

NAME

Indexes

INFORMATION\_SCHEMA

Users

H2 2.1.214 (2022-06-13)

Run Run Selected Auto complete Clear SQL statement:

SELECT \* FROM CUSTOMER |

SELECT \* FROM CUSTOMER;

ID	NAME
1	Mohamed
2	Yassine
3	Hanae
4	Imane

(4 rows, 2 ms)

Edit

BankAccount sur la BDD :

localhost:8081/h2-console/login.do?sessionId=54e4eefcfe54228df0babac70bfbefb

Auto commit Max rows: 1000 Auto complete Off Auto select On

jdbc:h2:mem:account-db

BANK\_ACCOUNT

CUSTOMER

ID

NAME

Indexes

INFORMATION\_SCHEMA

Users

H2 2.1.214 (2022-06-13)

Run Run Selected Auto complete Clear SQL statement:

SELECT \* FROM BANK\_ACCOUNT

SELECT \* FROM BANK\_ACCOUNT;

ID	BALANCE	CREATED_AT	CURRENCY	TYPE	CUSTOMER_ID
3f8567a9-39a6-4037-bcd3-df36c93c0f70	77793.49737859333	2022-11-02 22:25:18.403	MAD	SAVING_ACCOUNT	1
e5e2f3e4-4dc2-4fcd-ad8b-fe65295ae5ce	51018.56872735981	2022-11-02 22:25:18.439	MAD	CURRENT_ACCOUNT	1
01508897-0bc0-44f8-b74d-07b65f39bc10	80462.91440956236	2022-11-02 22:25:18.442	MAD	SAVING_ACCOUNT	1
47bf7d1-739a-4e65-8fb-29dccc266107	36572.61855714495	2022-11-02 22:25:18.444	MAD	CURRENT_ACCOUNT	1
244b504b-f5da-42ff-b760-5e5f57174e59	22984.812968868522	2022-11-02 22:25:18.447	MAD	SAVING_ACCOUNT	1
9546d057-ddfc-41ca-8e9b-116d8f1cc6ce	84561.67199602436	2022-11-02 22:25:18.451	MAD	SAVING_ACCOUNT	1
f9e7575e-4324-45e0-84f3-450a8ff9c1c6	73481.49827110142	2022-11-02 22:25:18.454	MAD	CURRENT_ACCOUNT	1
3e4267a0-8b59-48da-89f5-7ac7ac02d4df	64330.81624088672	2022-11-02 22:25:18.459	MAD	CURRENT_ACCOUNT	1
5e804422-2d8c-4b8d-ac9e-2d59e7968601	36107.910450520525	2022-11-02 22:25:18.461	MAD	SAVING_ACCOUNT	1
fc968f47-2063-4587-915e-761ac3b47580	74970.5798583435	2022-11-02 22:25:18.465	MAD	SAVING_ACCOUNT	1
2d890256-fbf9-4a2b-85c9-b67e02a01932	62073.53209149498	2022-11-02 22:25:18.467	MAD	CURRENT_ACCOUNT	2
09c27742-9004-4b21-baed-8ccda46d6b6f	79235.1629868197	2022-11-02 22:25:18.471	MAD	SAVING_ACCOUNT	2
fd888806-6105-45cc-9202-70d7e9853baf	64896.301250705495	2022-11-02 22:25:18.476	MAD	CURRENT_ACCOUNT	2
04487e10-8c47-436a-84c4-6dd7e2ad6a60	49988.720725444444	2022-11-02 22:25:18.478	MAD	CURRENT_ACCOUNT	2
e3f2fca6-1305-4ea2-8c66-826f9431bc04	39178.459529639906	2022-11-02 22:25:18.481	MAD	CURRENT_ACCOUNT	2

Query liste des comptes :

localhost:8081/graphql?path=/graphql#

GraphiQL

```
1 query {
2   accountList {
3     id, balance, customer { name, bankAccount { id } }
4   }
5 }
```

```
{
  "data": {
    "accountList": [
      {
        "id": "6207a579-d6a5-482d-9bc5-e23e79458d25",
        "balance": 12129.857230396996,
        "customer": {
          "name": "Mohamed",
          "bankAccount": null
        }
      },
      {
        "id": "e032d382-b6e6-4b98-bd67-ced04c8155da",
        "balance": 31659.514435676334,
        "customer": {
          "name": "Mohamed",
          "bankAccount": null
        }
      },
      {
        "id": "d5b93713-13d0-4ed1-a356-"
      }
    ]
  }
}
```

Variables Headers

## Query Customers :

localhost:8081/graphql?path=/graphql#

GraphiQL

```
1 query {
2   customers {
3     id, name, bankAccounts { balance, id, type }
4   }
5 }
```

```
{
  "data": {
    "customers": [
      {
        "id": "1",
        "name": "Mohamed",
        "bankAccounts": [
          {
            "balance": 55809.689277633996,
            "id": "51bd0f6d-97b0-431a-98bb-00405c44bd00",
            "type": "SAVING_ACCOUNT"
          },
          {
            "balance": 25435.94417259089,
            "id": "77c223d6-56ef-4055-986a-4812c7001049",
            "type": "SAVING_ACCOUNT"
          },
          {
            "balance": 57662.25653150449,
            "id": "f9e2488d-ddb3-465f-a2be-f69c6ce0efcf",
            "type": "SAVING_ACCOUNT"
          }
        ]
      }
    ]
  }
}
```

Variables Headers

## API BankAccounts :



<untitled>

<untitled> x

<untitled>

<untitled>

+

1 query{  
2 customers {  
3 id, name, bankAccounts {balance}  
4 }  
5 }

▶

🔍

🗑️

📄

Variables

Headers

GraphQL

{  
 "data": {  
 "customers": [  
 {  
 "id": "1",  
 "name": "Mohamed",  
 "bankAccounts": [  
 {  
 "balance": 62877.15029077908  
 },  
 {  
 "balance": 88834.65880828467  
 },  
 {  
 "balance": 20638.255083221473  
 },  
 {  
 "balance": 10249.488447142436  
 },  
 {  
 "balance": 31916.283384910337  
 }  
 ]  
 }  
 ]  
 }  
}