

✓ Terminé

On souhaite créer une requête de mutation, qui va enregistrer un nouveau client. Cette requête prendra en paramètre le *nom* mais pas l'identifiant, qui sera attribué par Postgres : cet identifiant doit être retourné par la requête.

1. la requête d'insertion est alors :

```
INSERT INTO client(nom) VALUES('toto') RETURNING ID;
```

2. dans le schéma GraphQL, commencer par créer un type Mutation pour définir la requête, qui retournera un booléen dans un premier temps, pour tester :

```
type Mutation{
  insertClient(nom:String!): Boolean
}
```

3. insérer un résolveur pour tester :

```
Mutation:{
  insertClient: (parent, args) => true;
},
```

4. tester avec la requête suivante :

```
mutation {
  insertClient(nom: "tata")
}
```

5. terminer le travail en modifiant le schéma pour que la mutation renvoie l'identifiant, idem pour le resolver qui doit faire l'insertion dans Postgres et renvoyer l'identifiant.

6. tester avec l'interface web une mutation d'insertion.

- en utilisant l'interface pour contruire la mutation, vous obtenez la mutation suivante\ :

```
mutation Mutation($nom: String!) {
  insertClient(nom: $nom)
}
```

- elle doit être complétée par le renseignement du nom dans la section *variables* de la mutation\ :

```
{
  "nom": "tonton"
}
```

- en cliquant sur les trois petits points à droite de la requête, vous pouvez obtenir une version *inline* de la mutation\ :

```
mutation Mutation {
  insertClient(nom: "tonton")
}
```

- notez cependant que ce n'est pas une bonne idée d'utiliser une version *inline*. C'est préférable d'avoir une mutation générique accompagnée d'une section de variables. En effet, d'une mutation à l'autre, seule la section des variables sera modifiée.

Modifié le: mardi 12 septembre 2023, 18:10

◀ TP 2.3 - GraphQL avec PostGres

Choisir un élément

Aller à...

TP 2.5 - Test d'une API avec Postman ►

?

[mentions légales](#), [vie privée](#), [charte utilisation](#), [unicaen](#), [cemu](#), [moodle](#)

