

## Provider 结合 resolver 搭建方式

### 一、前言

1. 这种方式的特点如下:

- 1) 既可以使用 `graphql.scalars.ExtendedScalars` 来进行类的扩展, 也可以使用自定义的类扩展;
- 2) 不能使用 `playground` 插件 (至少我没这么使用, 而且项目中实用性比较低, 如果愿意研究的话可以研究下, 或许可以使用)
- 3) 可以利用 `provider` 等结合 `controller` 等来进行前后端交互
- 4) `Graphqls` 必须只有一个根文件, 其他的 `query` 或者 `mutation` 必须继承根文件的类型才可以正常使用
- 5) 总体来说, **实用性很强**, 而且 `resolver` 方式相对于 `fetcher` 更加简单实用, 所以推荐这种方式;

### 二、框架搭建方式及技术升级

- 1) 框架搭建方式见 `demo`, 是个很简单的 `provider_resolver` 框架, 下面将通过图例的方式来介绍各部分的功能



- 2) 技术升级：技术升级参照官网及视频教程，官网的话比较全面，但是是全英文文档，视频的话讲解的比较详细，大概包括以下内容：

视频选集 (1/17)		自动连播
P1	01-课程说明和环境需求	16:49
P2	02-创建Schema和Query	22:56
P3	03-maxQueryDepth配置演示	14:03
P4	04-Playground配置演示	12:07
P5	05-Schema可视化Voyager演示	06:08
P6	06-数据解析器Resolvers	13:43
P7	07-定制异常处理Spring ExceptionHandler	11:00
P8	08-定制异常处理GraphQLExceptionHandler	08:08
P9	09-通过DataFetcherResult返回部分数据	05:01
P10	10-异步数据解析(Async Resolvers)	09:52
P11	11-改变数据(Mutations)	11:46
P12	12-文件上传(File Upload)	12:28
P13	13-DataFetchingEnvironment_Selection...	09:48
P14	14-定制标量(Custom Scalars)	13:03
P15	15-定制日期标量(Date Type Scalars)	09:00
P16	16-输入校验(Input Validation)	05:29
P17	17-请求监听器(Request Listener)	10:16

- 3) 然后就是被教程给出的例子的访问方式，下面给出了以 postman 的访问方式

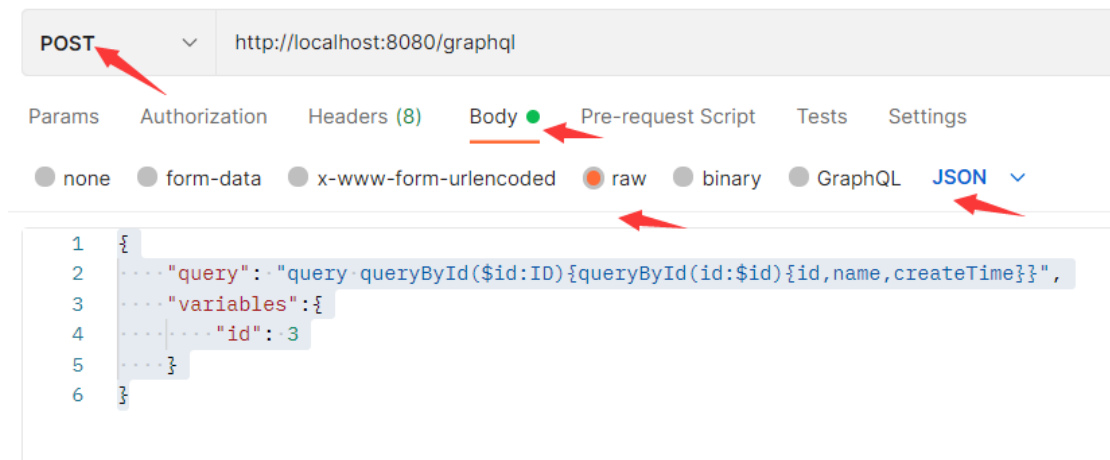
i. 接口一：updateName

```
{
  "query": "mutation updateName($param:Param){updateName(param:$param)
{id,name,createTime}}",
  "variables":{
    "param": {"id":1,"name":"java 项目"}
  }
}
```



ii. 接口二: queryById

```
{
  "query": "query queryById($id:ID){queryById(id:$id){id,name,createTime}}",
  "variables":{
    "id": 3
  }
}
```



iii. 接口三: queryItemList

```
{
  "query": "query queryItemList{queryItemList{itemList{id,name,createTime},total}}",
  "variables":{
    "id": 3
  }
}
```

POST ▼ http://localhost:8080/graphql

Params Authorization Headers (8) **Body** ● Pre-request Script Tests Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL **JSON** ▼

```
1 {
2   ... "query": "query queryItemList{queryItemList{itemList{id,name,createTime},total}}",
3   ... "variables":{
4     ... "id": 3
5   }
6 }
```