数据接入介绍

目 录

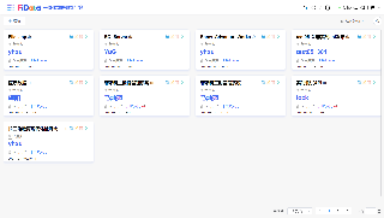
- 1. 概述
- 2. 界面概览
- 3. 数据接入
  - 3.1 创建非实时应用
    - 3.1.1 创建MySQL、SQL Server、Oracle驱动类型的应用
    - 3.1.2 创建ftp驱动类型的应用
    - 3.1.3 创建API驱动类型的应用
    - 3.1.4 使用非实时API
  - 3.2 创建实时应用
    - 3.2.1 创建RestfulAPI驱动类型的应用
- 4. 查看应用日志

1. 概述

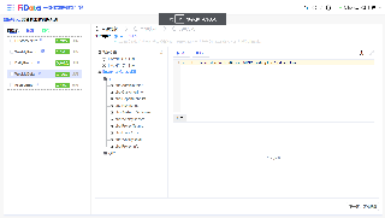
数据接入功能提供可视化的界面，方便用户进行数据的接入，支持SQL Server、Oracle、MySQL、Excel、CSV等多种数据源，数据支持以全量、增量、业务主键覆盖等方式将数据同步到本平台中

2. 界面概览

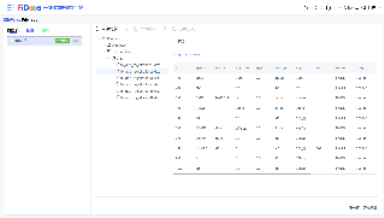
入口



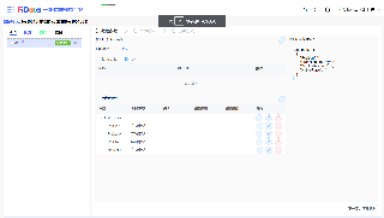
物理表



FTP页面

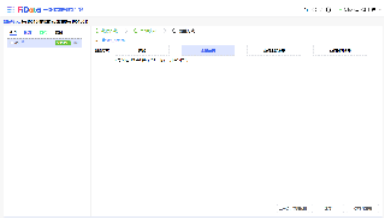


API页面



字段映射

覆盖方式

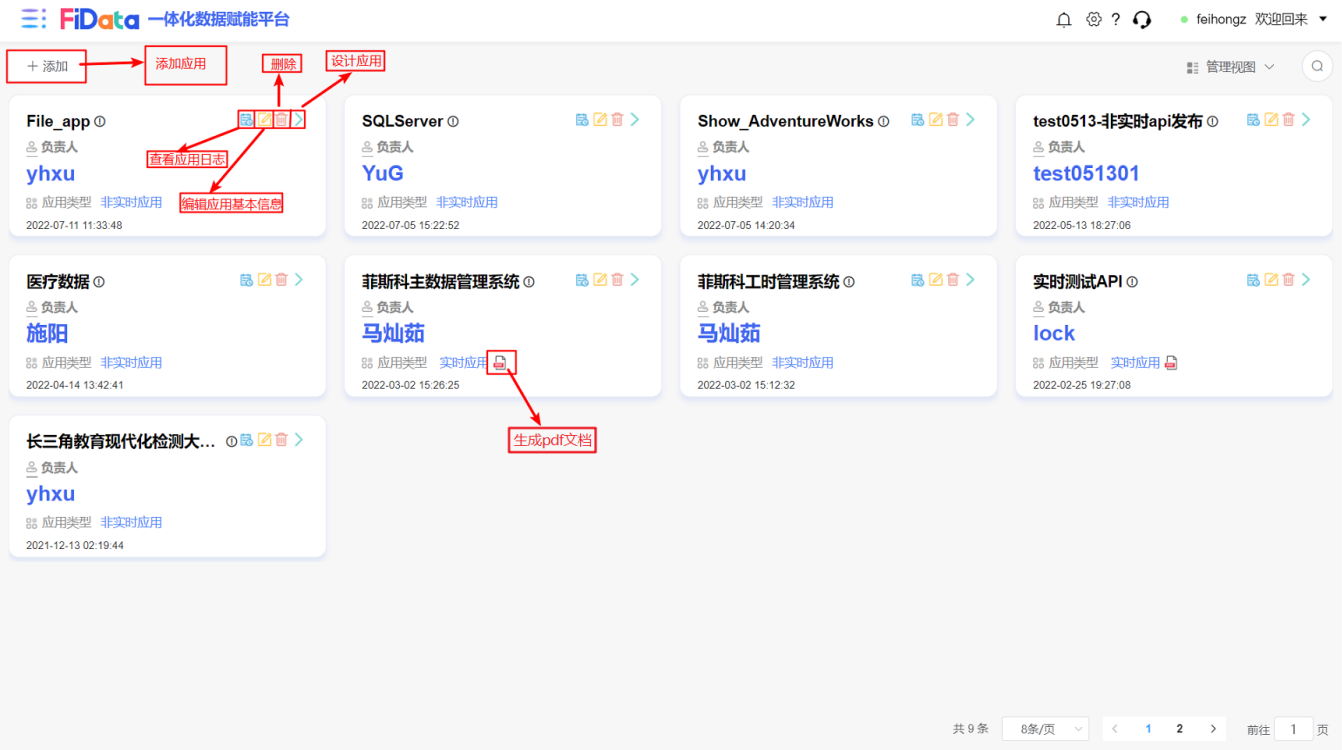


3. 数据接入

3.1 创建非实时应用

3.1.1 创建MySQL、SQL Server、Oracle驱动类型的应用

（1）创建应用： 点击添加，输入应用基本信息、联系人、驱动类型、服务器配置后，点击测试连接，连接通过，即可创建应用

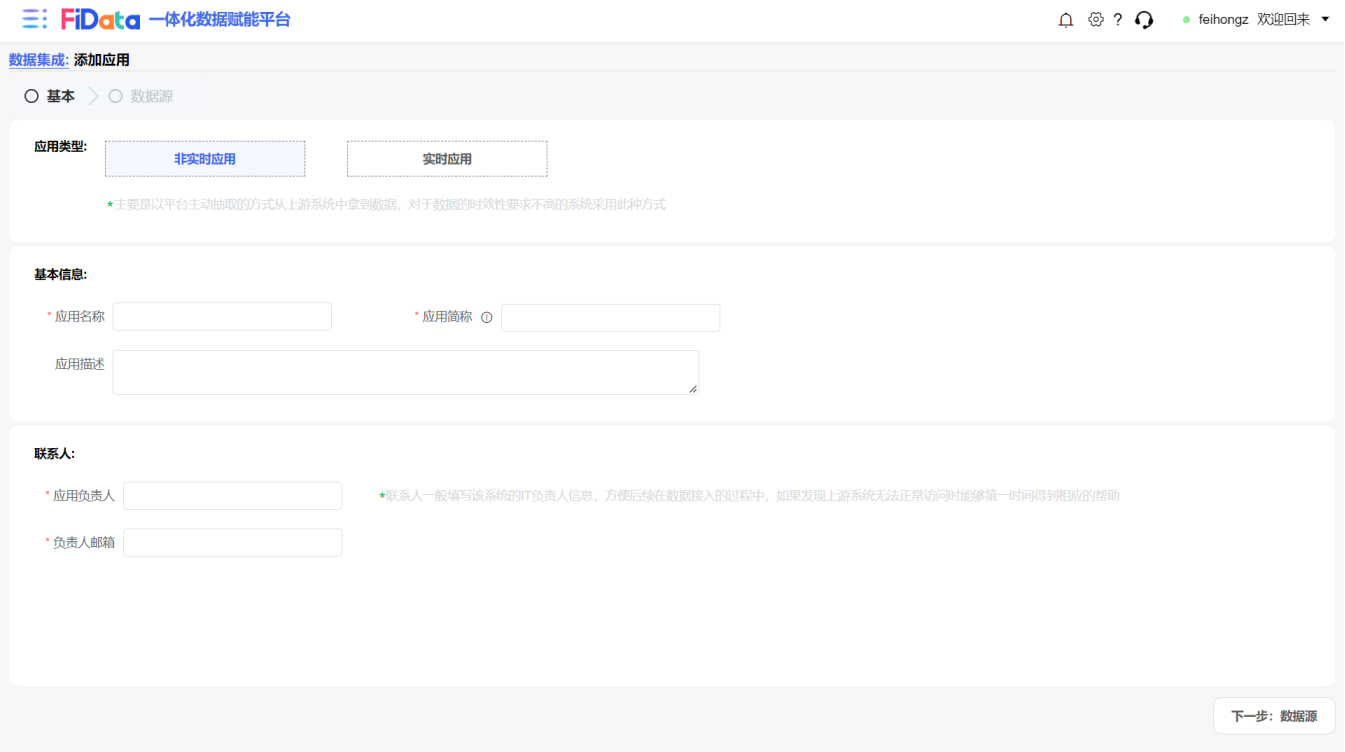


应用名称：中英文名称皆可

应用简称：一般是英文名称，类似数据库的架构名称

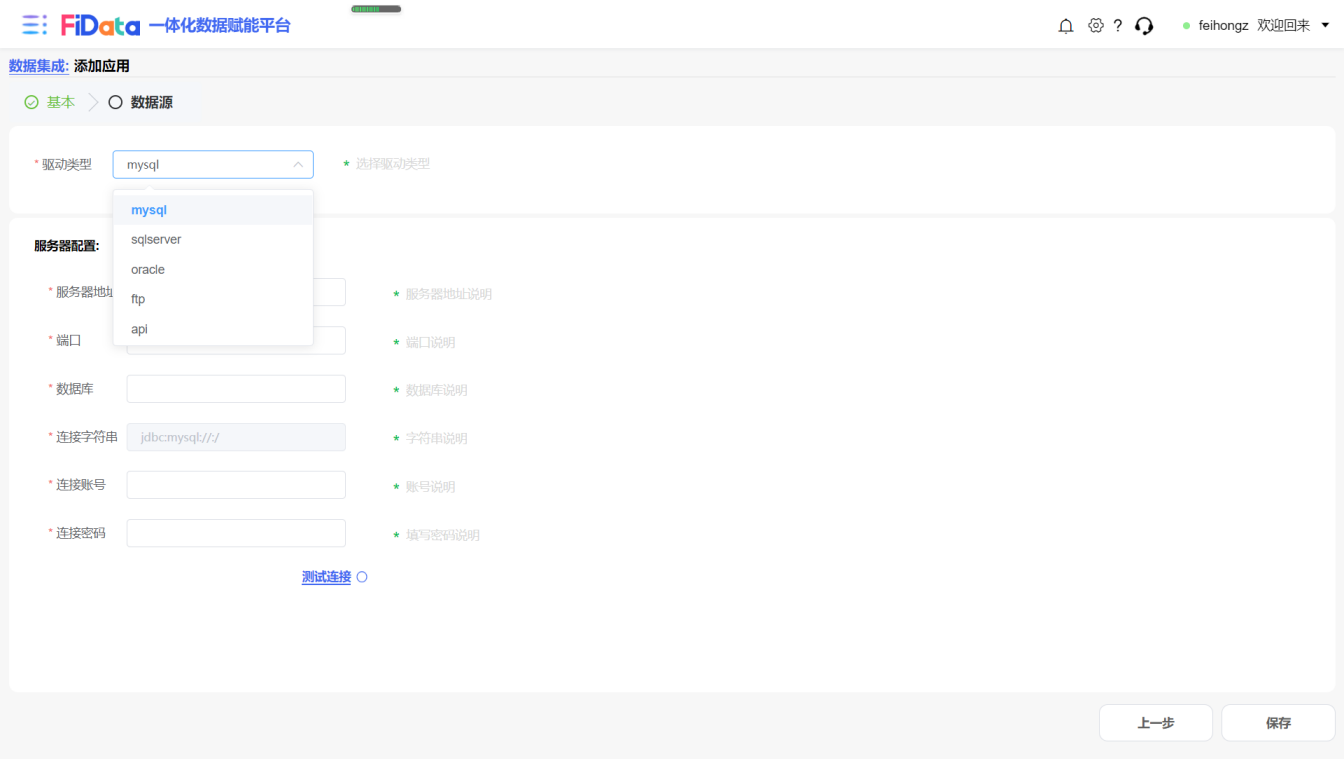
应用负责人：一般是该系统的IT负责人，方便后续在数据接入的过程中，如果发现上游系统无法正常访问时，能够第一时间得到相应帮助

负责人邮箱：同应用负责人作用一样



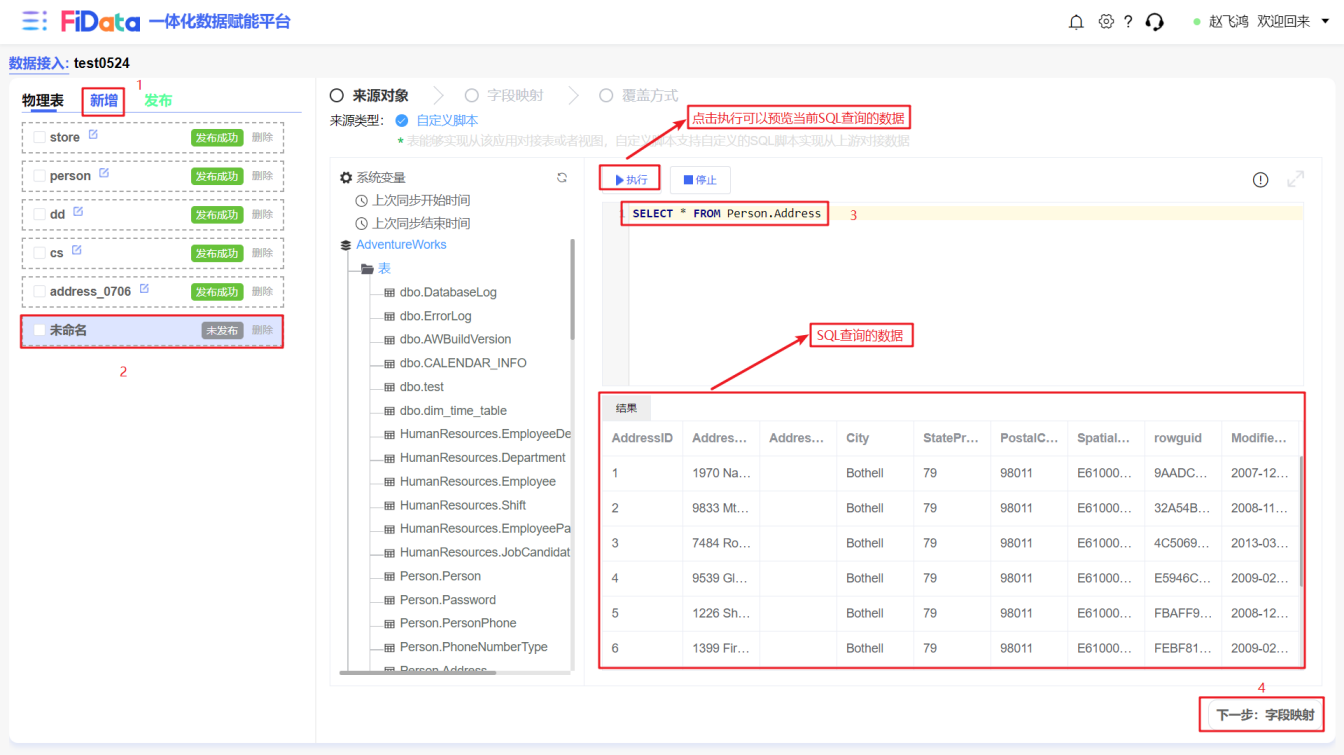
驱动类型：本平台支持的几种上游系统，如mysql、sqlserver、oracle、ftp文件系统、非实时API调用、RestfulAPI实时推送

服务器配置：本平台同步数据时，用于连接上游系统的服务器基本信息



（2）创建物理表

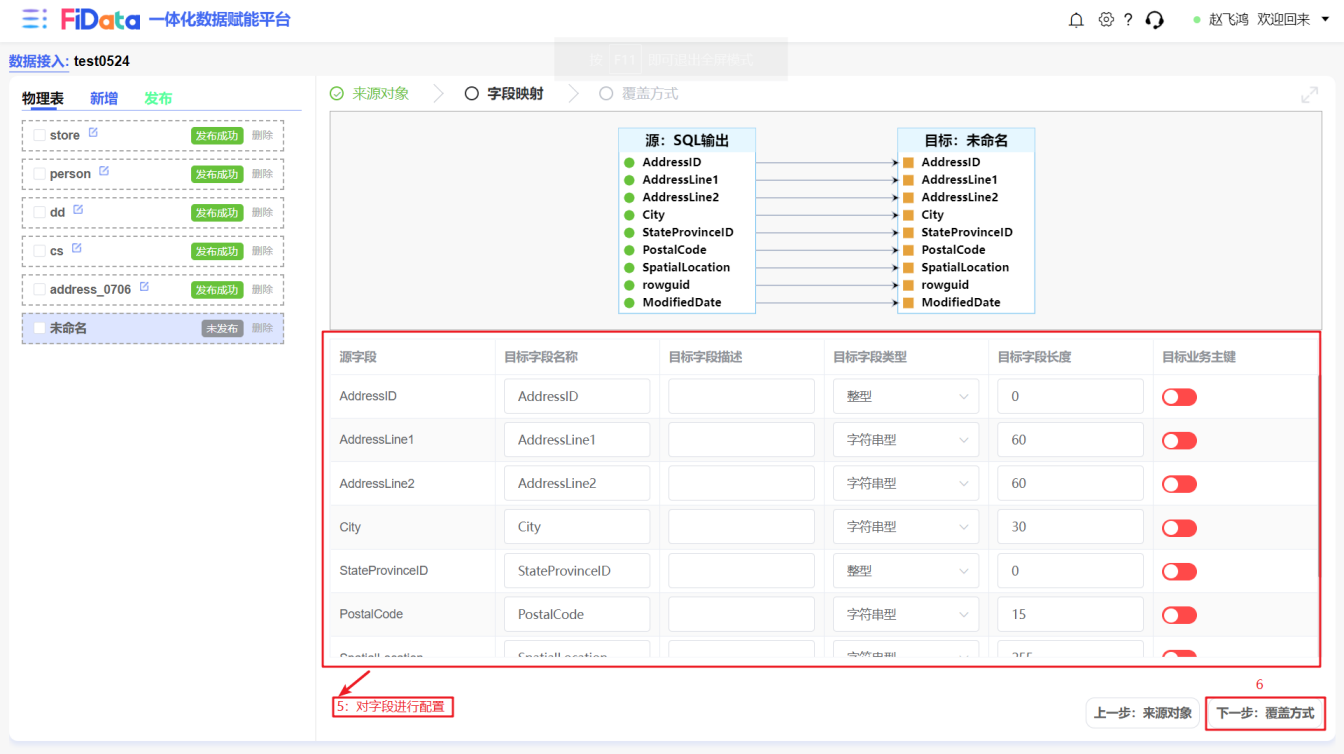
1.新增一张物理表 → 2.选择新增的未命名表 → 3.在页面的SQL编辑器中输入业务SQL→ 4.进入字段映射配置页面；



5.对字段进行配置→ 6.进入覆盖方式页面

数据类型：

- 文本：
    - 存放文本类型数据
    - 设置长度
- 整型：
    - 存放数值类型，无小数类型
- 时间戳：
    - 存放时间类型数据
- 浮点数：
    - 存放小数类型数据
    - 设置精度
- 字符串：
    - 存放字符串类型数据
    - 设置长度



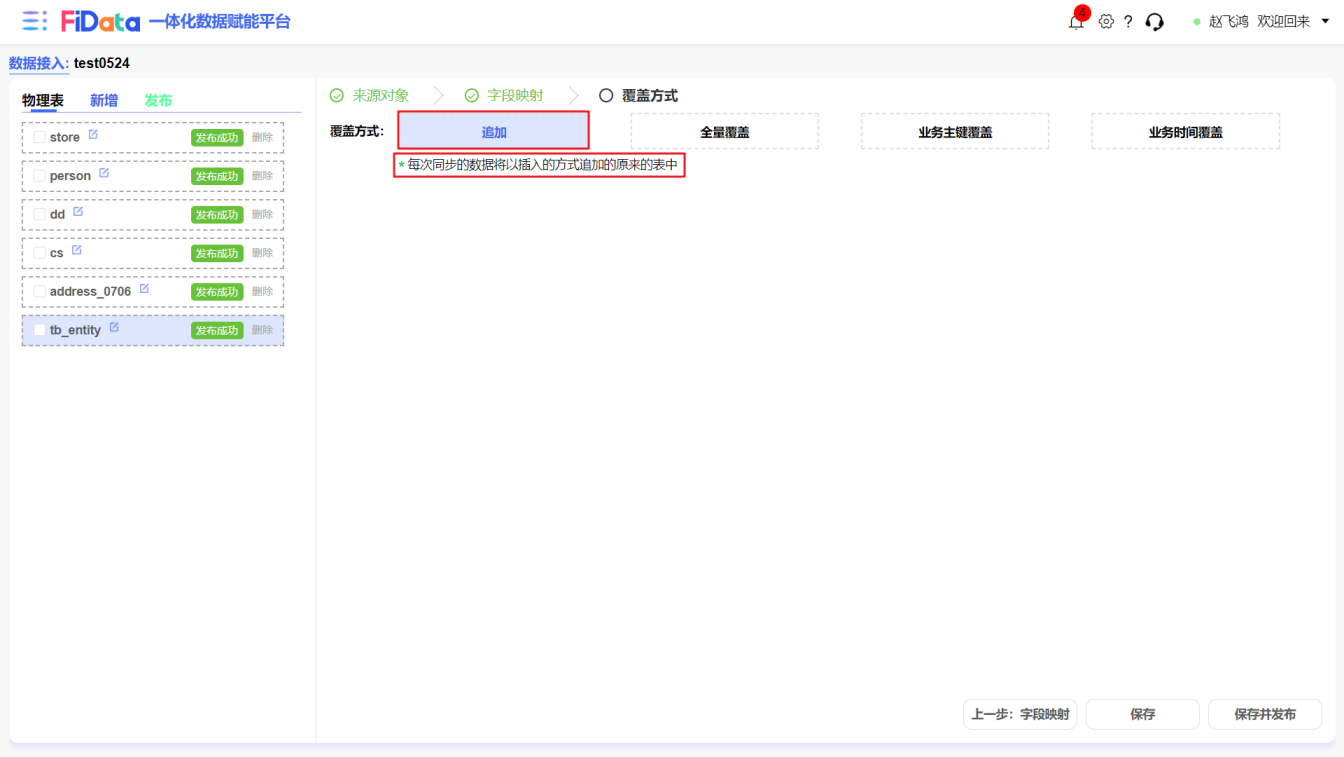
7.选择同步数据的覆盖方式→ 8.保存&发布物理表

9.填写表基本信息（表名、表描述）

10.发布

同步方式：提供追加、全量覆盖、业务主键覆盖和业务时间覆盖等四种同步方式

- 追加：每次同步的数据将以插入的方式追加的原来的表中



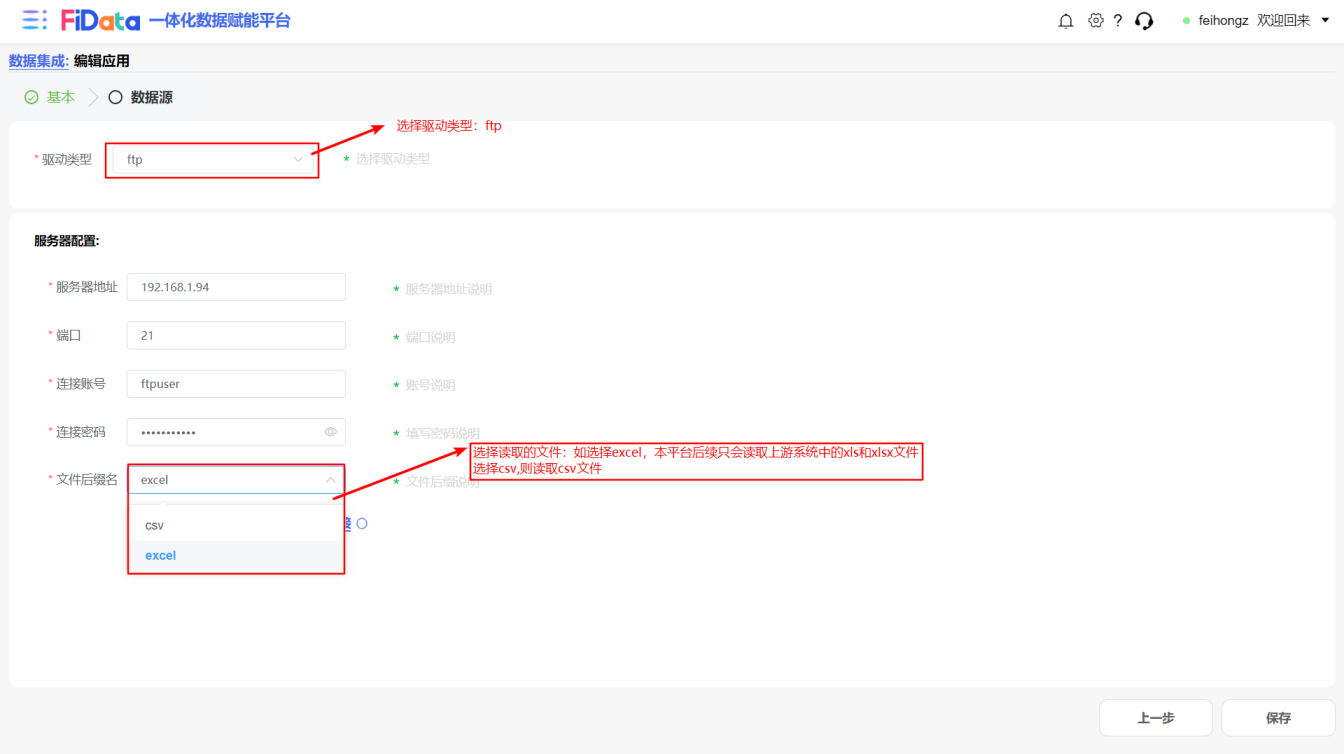
- 全量：每次同步时系统将清空原来的表，并重新插入

- 业务主键覆盖：系统将根据当前设定的业务主键进行数据覆盖

- 业务时间覆盖：系统将根据以下选择的条件先从表中删除满足条件的数据，然后重新从插入满足条件的数据

3.1.2 创建ftp驱动类型的应用

（1）选择ftp驱动类型，填写ftp服务器基本信息，测试连接，连接通过即可保存应用



（2） 创建物理表

1.新增一张物理表

2.选择新增的未命名表

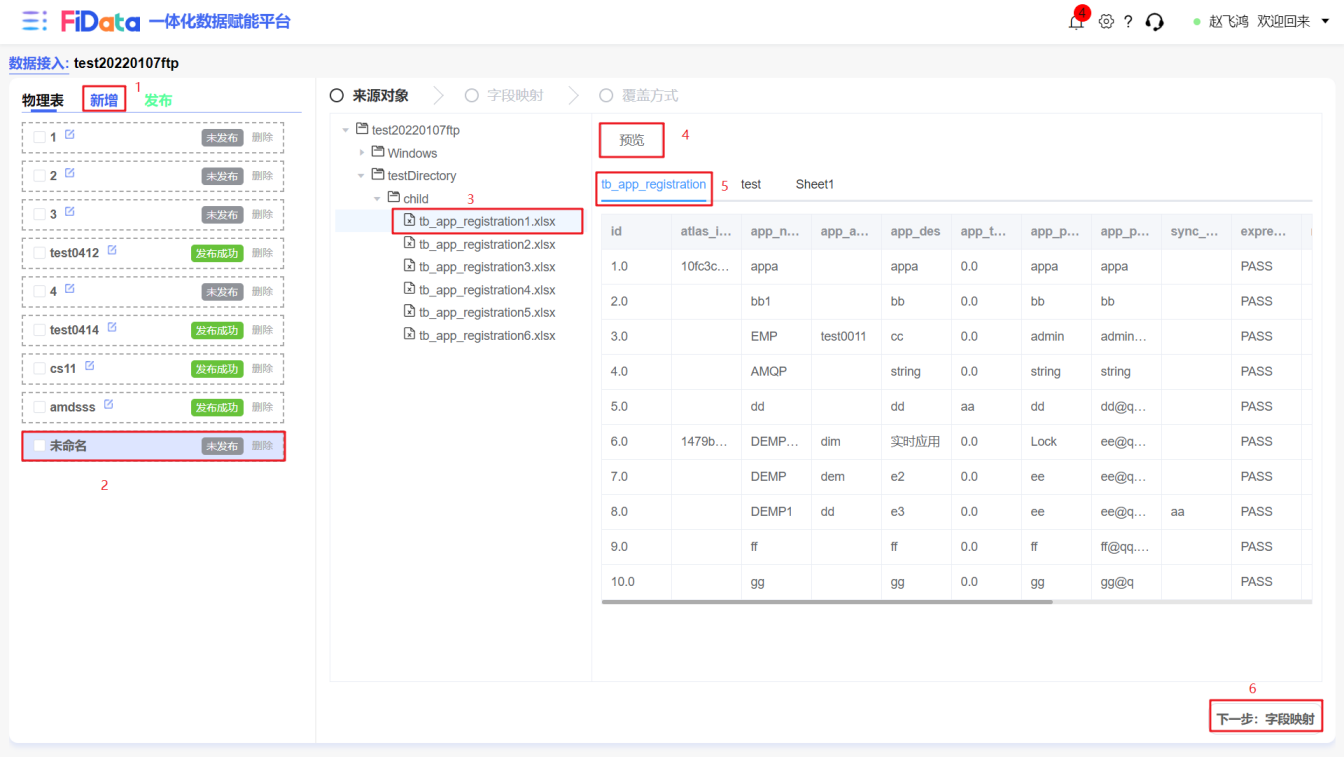
3.选择本次需要同步的excel/csv文件

4.预览文件内容

5.选择本次同步数据的sheet页

6.进入字段映射页面

其他操作流程同mysql驱动类型创建物理表一致



3.1.3 创建API驱动类型的应用

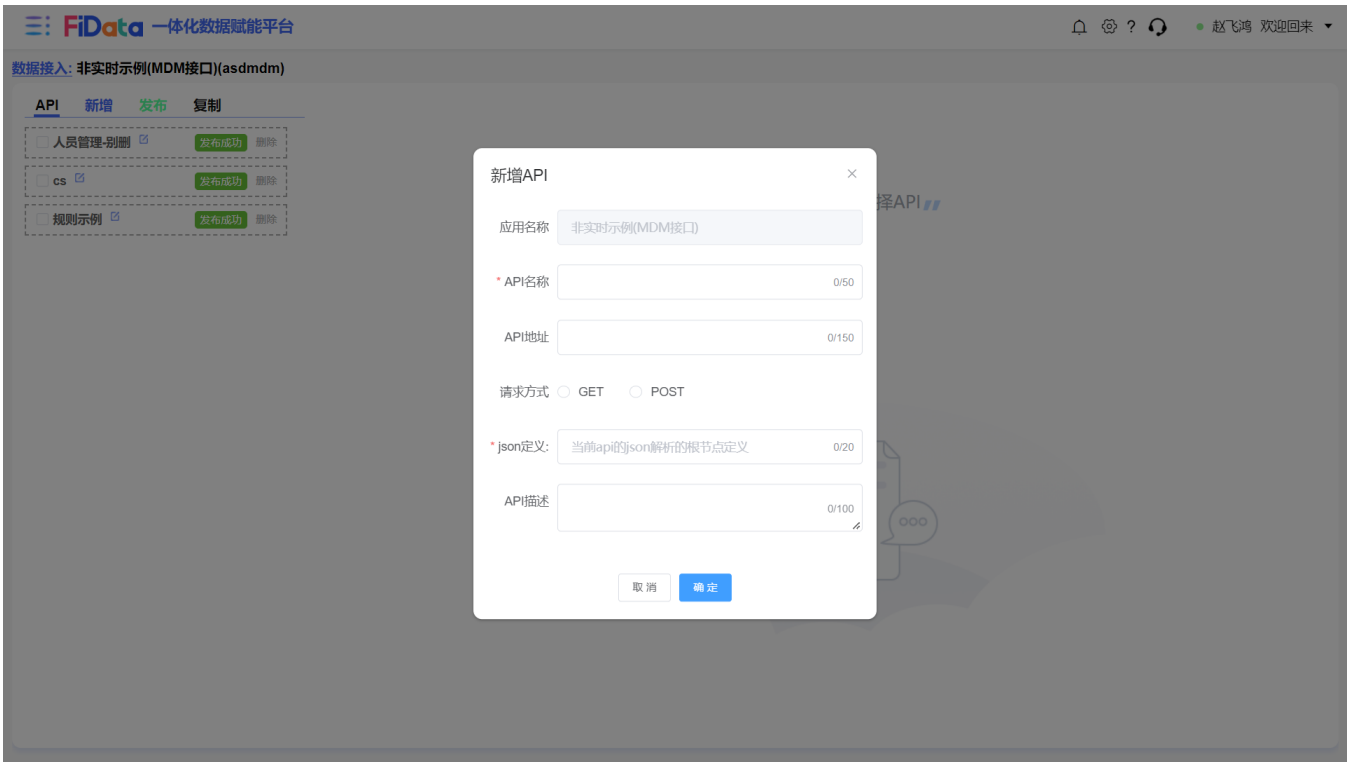
(1) 创建API：点击页面“新增”，输入API基本信息，即可创建API

API名称：中英文名称皆可。

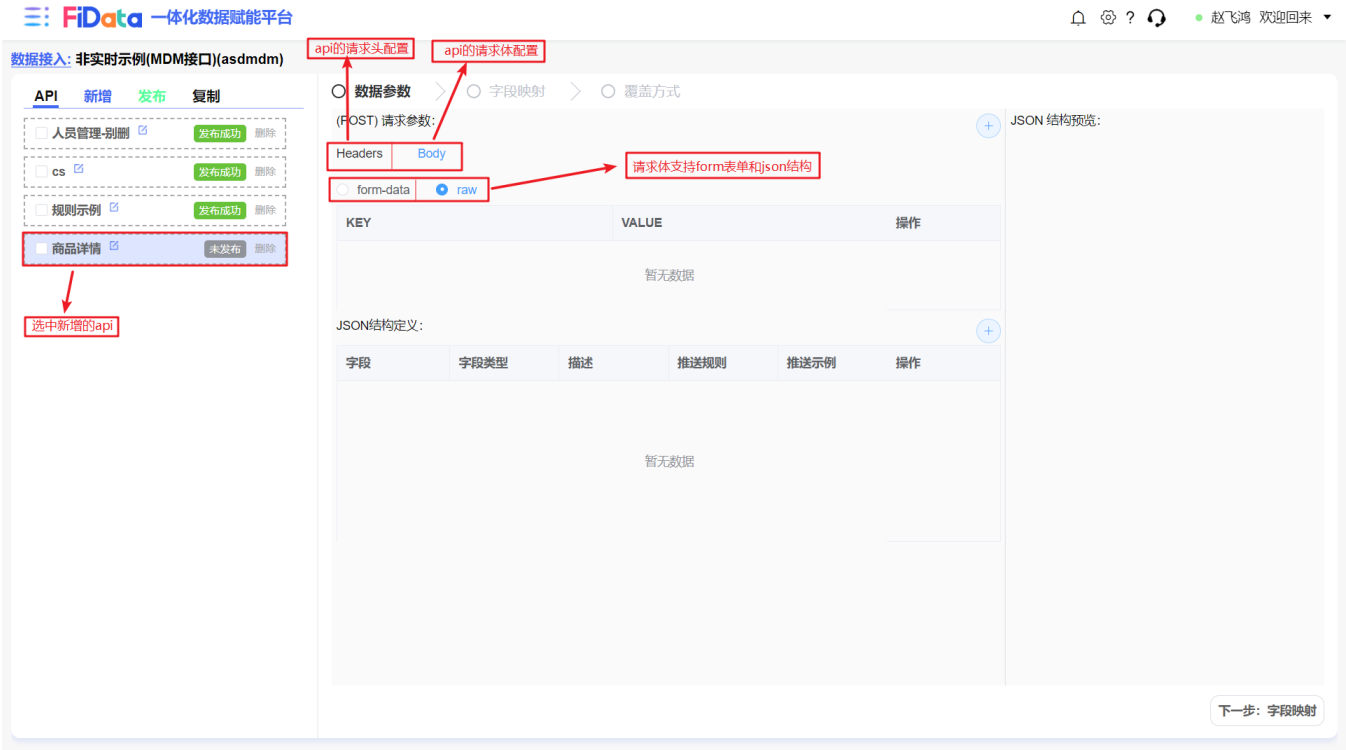
API地址：一般是API的访问地址。

请求方式：调用API时指定的请求方式。

JSON定义：API的JSON解析根节点定义。



（2）配置API请求方式



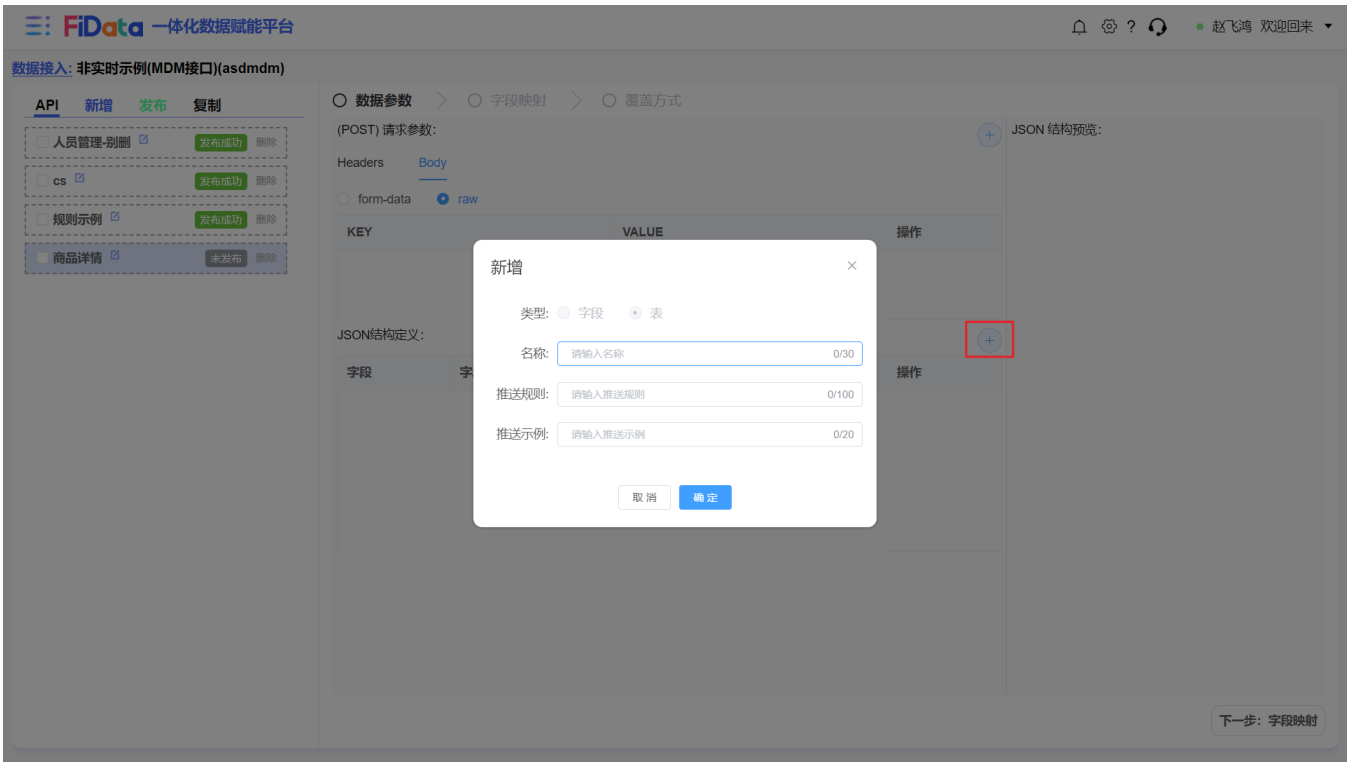
（3）配置API的JSON结构定义

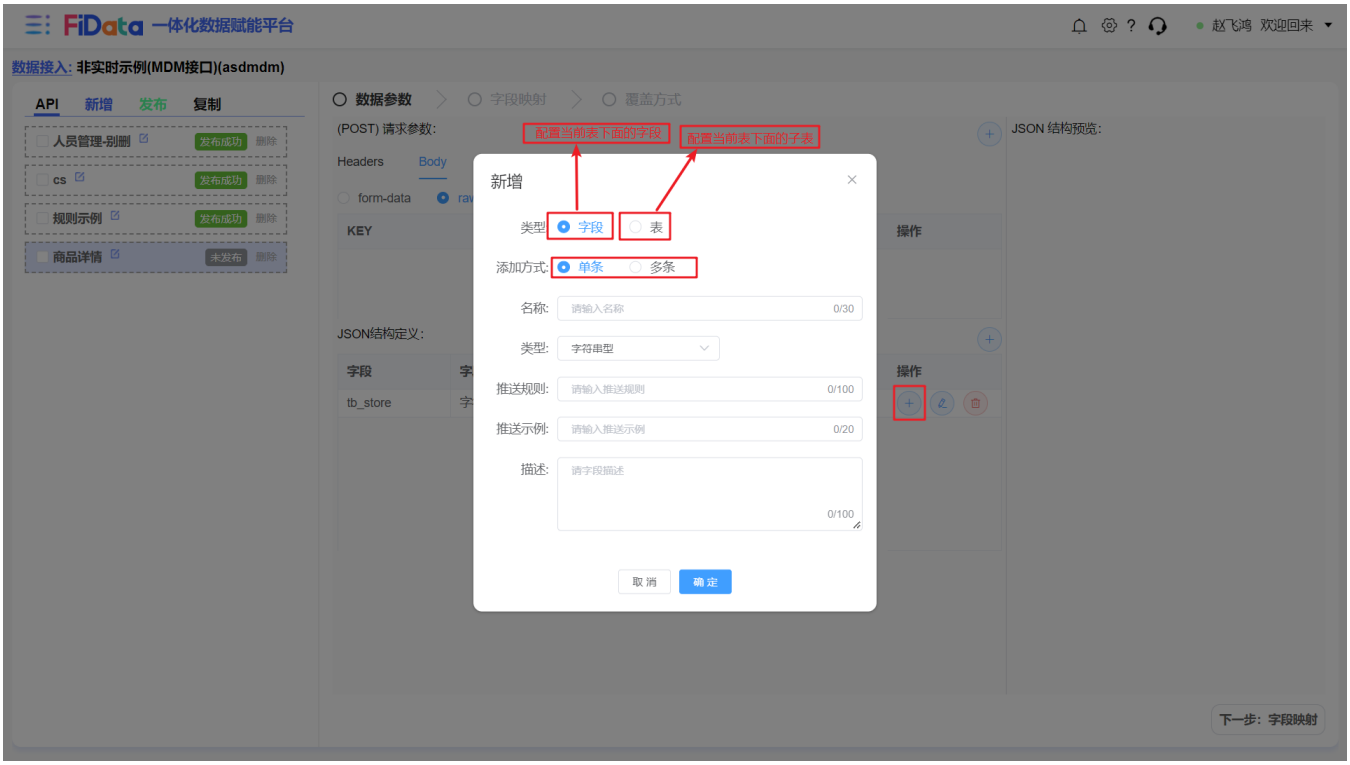
名称：一般是表英文名称or字段英文名称

类型：字段类型

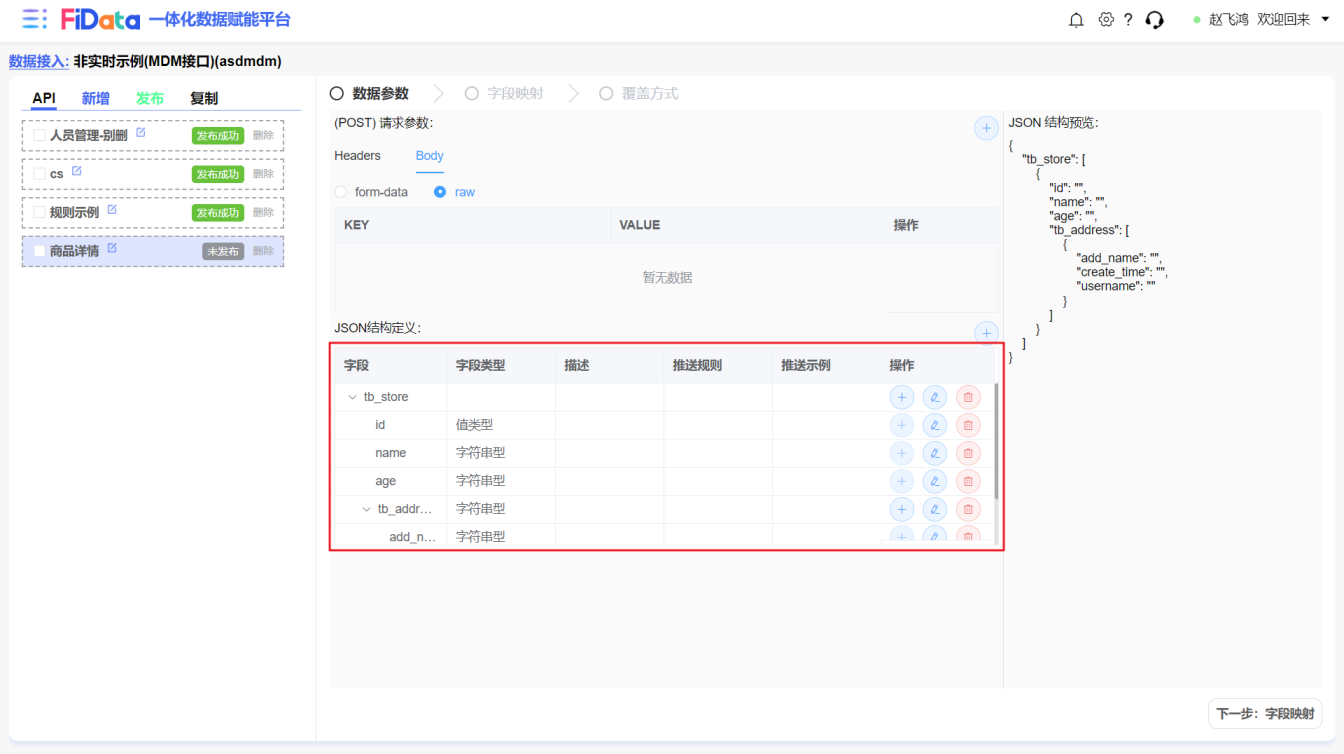
推送规则：一般是字段的推送规则描述，如字段不为空

推送示例：一般是API推送的测试数据，如设置了name字段推送示例为“上海”，在发布API的时候选择按照配置调用一次API，将name字段赋值“上海”

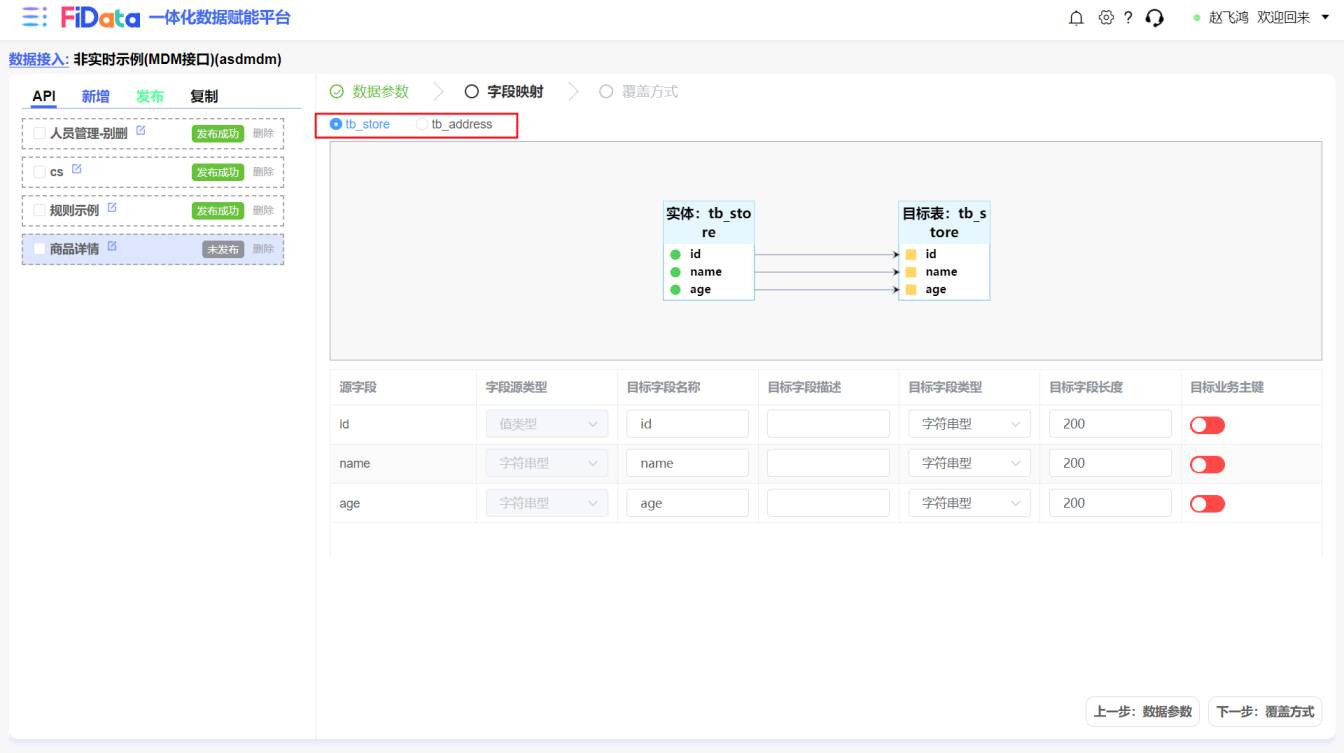

FiData 一体化数据赋能平台
赵飞鸿 欢迎回来
数据接入: 非实时示例(MDM接口)(asdmdm)
API 新增 发布 复制
人员管理-别删 发布成功 删除
cs 发布成功 删除
规则示例 发布成功 删除
商品详情 未发布 删除
数据参数
字段映射
覆盖方式
(POST) 请求参数:
Headers Body
form-data raw
KEY VALUE 操作
JSON结构定义:
字段 操作
JSON 结构预览:
新增
类型: 字段 表
名称: 请输入名称 0/30
推送规则: 请输入推送规则 0/100
推送示例: 请输入推送示例 0/20
取 消 确 定
下一步: 字段映射


FiData 一体化数据赋能平台
赵飞鸿 欢迎回来
数据接入: 非实时示例(MDM接口)(asdmdm)
API 新增 发布 复制
人员管理-别删 发布成功 删除
cs 发布成功 删除
规则示例 发布成功 删除
商品详情 未发布 删除
数据参数
字段映射
覆盖方式
(POST) 请求参数:
配置当前表下面的字段
配置当前表下面的子表
Headers Body
form-data
KEY 操作
JSON结构定义:
字段 操作
tb_store
JSON 结构预览:
新增
类型: 字段 表
添加方式: 单条 多条
名称: 请输入名称 0/30
类型: 字符串型
推送规则: 请输入推送规则 0/100
推送示例: 请输入推送示例 0/20
描述: 请字段描述 0/100
取 消 确 定
下一步: 字段映射

字段映射

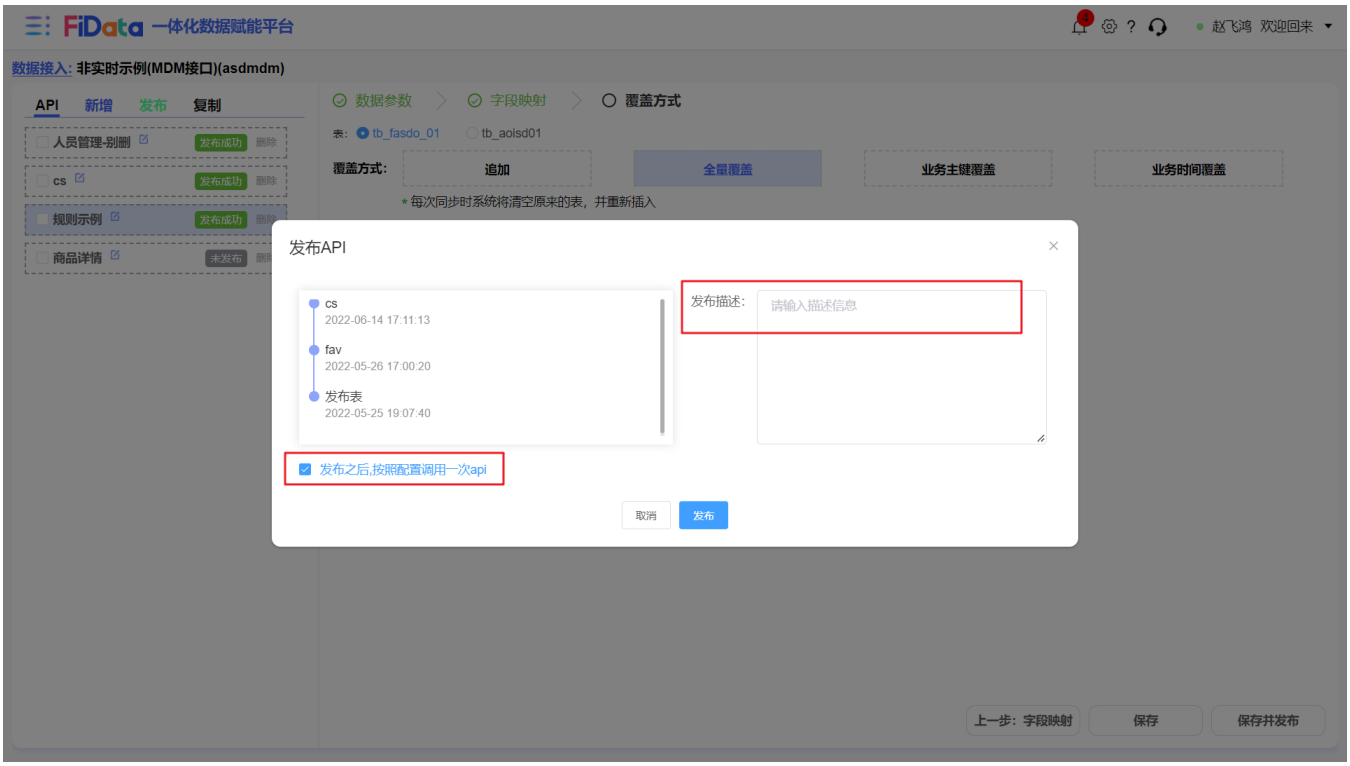


比如配置的JSON结构为一主一从结构，那么字段映射页面就要配置两张表的字段



覆盖方式同时也需要配置多表

勾选选项，发布后，会根据配置调用一次API，并将API的数据推送到本平台中



3.1.4 使用非实时API

1.点击“开始”并配置，选择”数据湖表任务“→ 2.选择非实时API应用

3.点击”+“，添加API任务→ 4.选择具体的API，支持多API操作→5.指定到右边，方便调整API的调用顺序→ 6.保存API

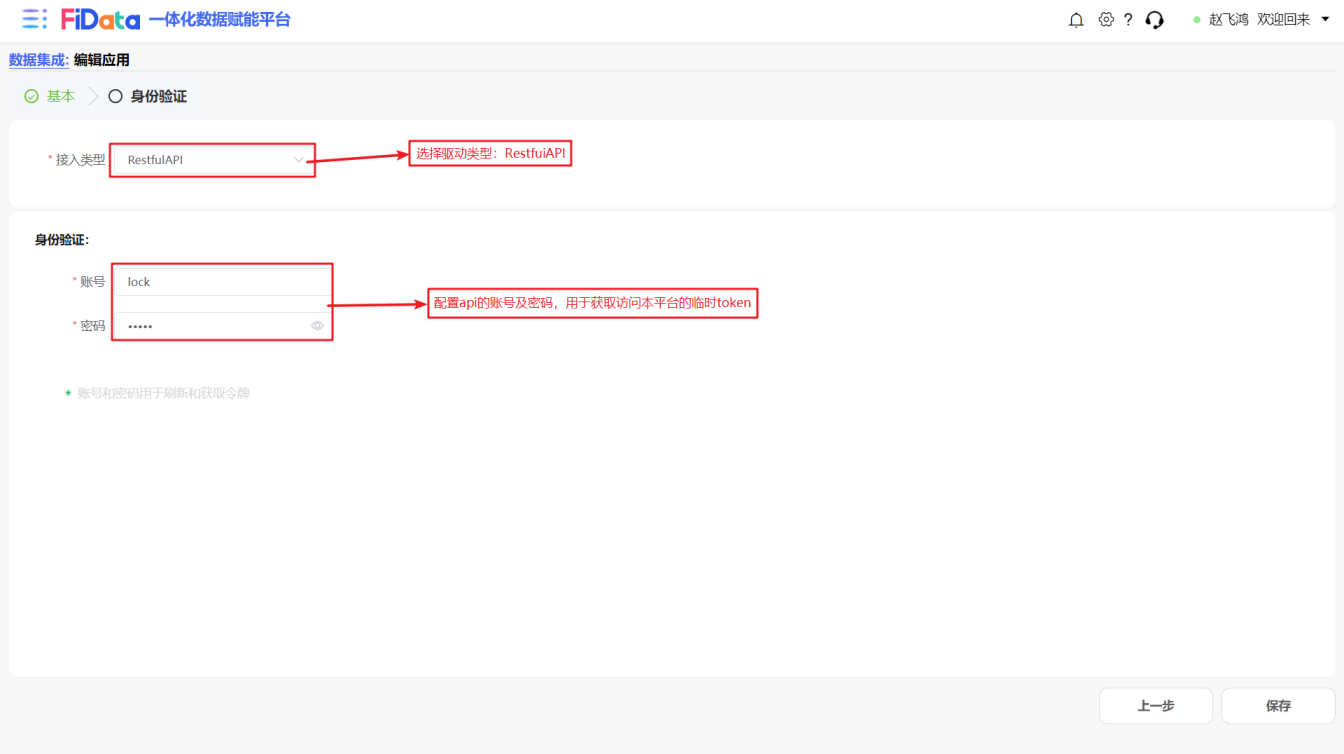
7.保存并发布

管道的创建和基本使用在数据调度介绍文档。

3.2 创建实时应用

3.2.1 创建RestfulAPI驱动类型的应用

(1)创建API 及构建json结构同上述方法一致，本平台实时API没有配置API请求方式



(2) 使用实时API

点击生成文档，在文档中有详细的调用方式。

以postman为例：

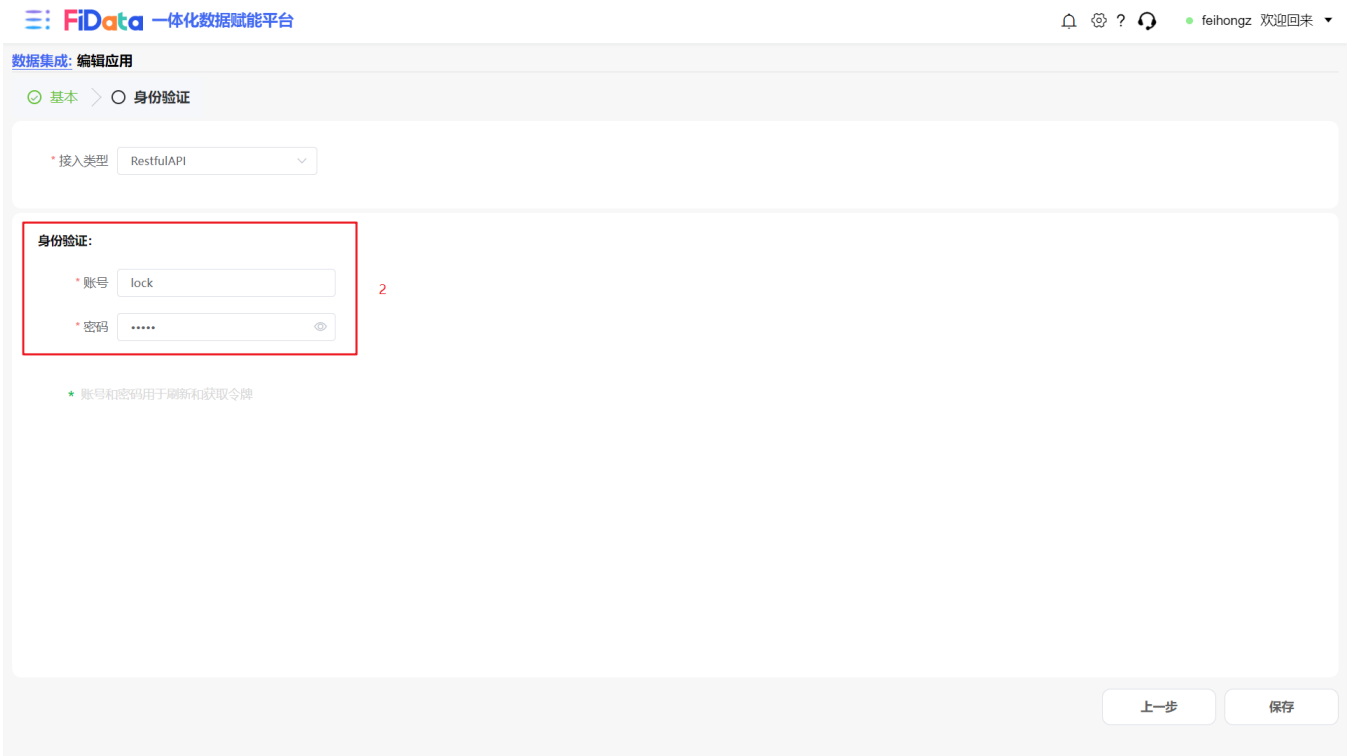
1.复制pdf文档中获取临时token的接口地址；

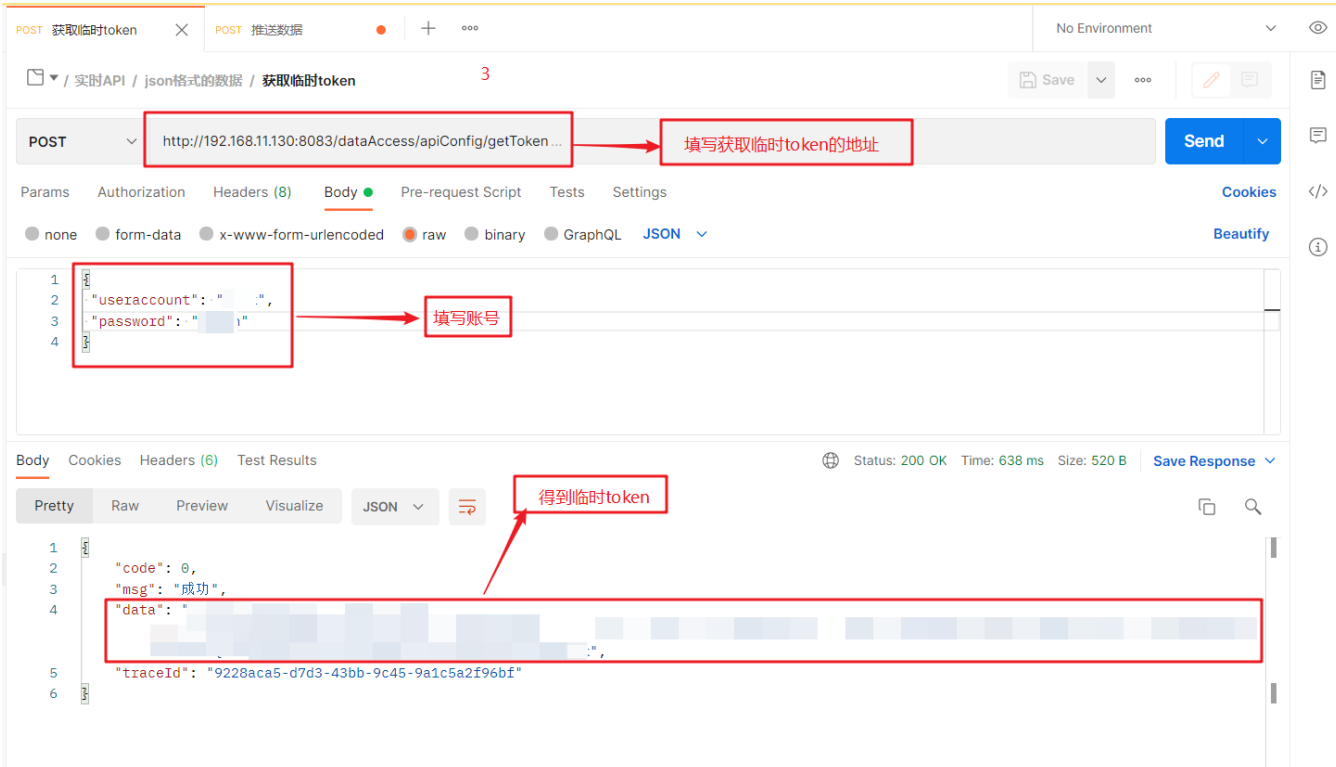
2.获取当前应用的账号及密码；

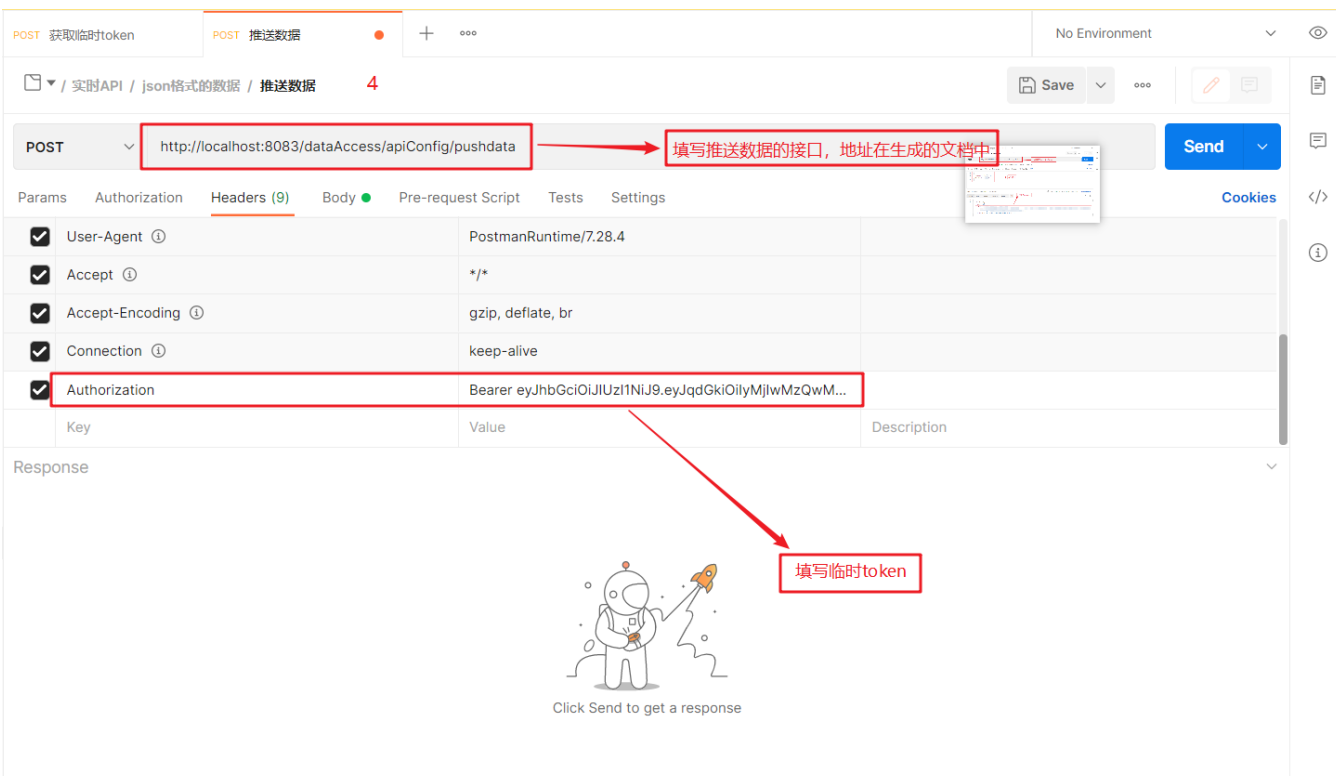
3.使用postman，填写接口地址，填写请求参数的账号及密码，调用接口，获取临时token（有效期60min）；

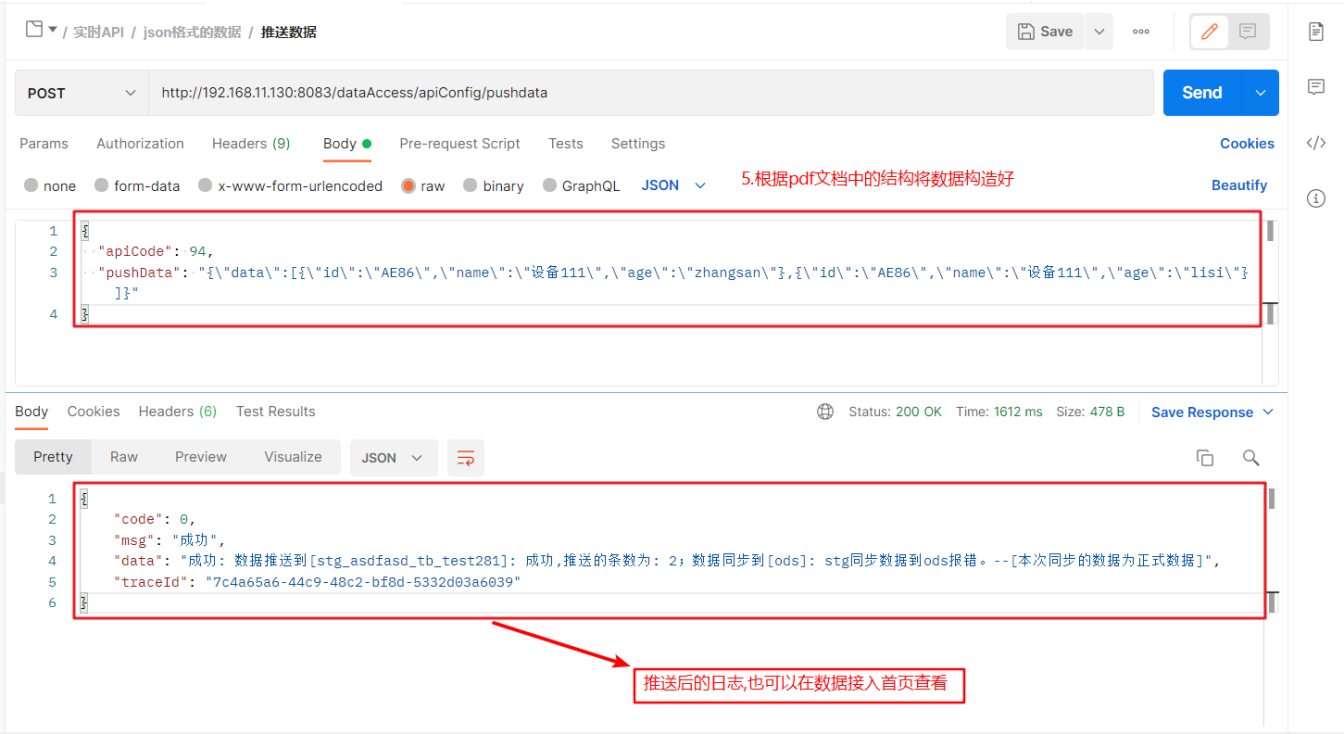
4.查看pdf文档中推送数据的接口地址，在postman中填写地址，填写请求头信息，并将临时token写入请求头中；

5.根据pdf文档中的接口构造好需要推送的数据，调用接口即可，可也通过接口返回值查看接口的调用结果，也可以在数据接入首页应用查看日志。

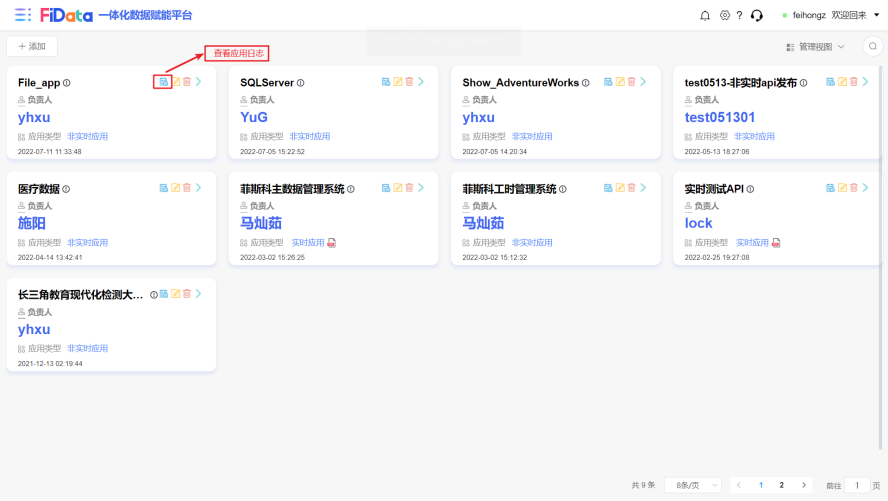


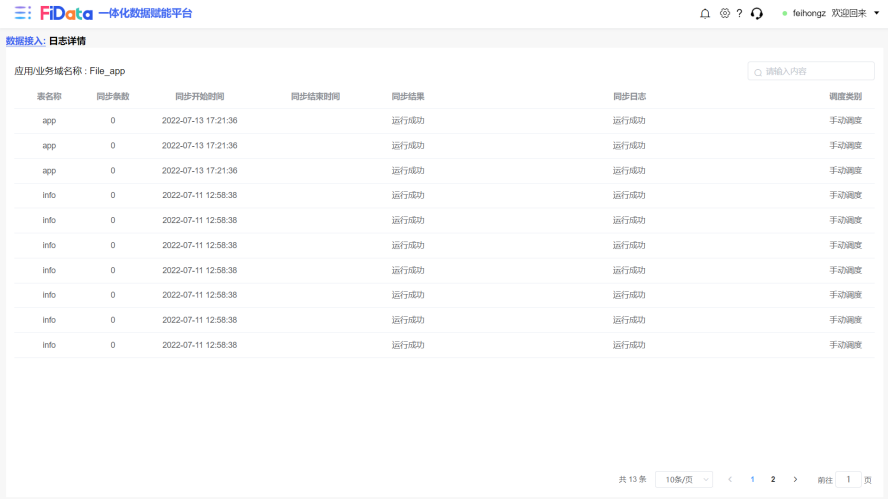

POST 获取临时token
POST 推送数据
No Environment
/ 实时API / json格式的数据 / 获取临时token
3
Save
POST
http://192.168.11.130:8083/dataAccess/apiConfig/getToken ...
填写获取临时token的地址
Send
Params Authorization Headers (8) Body Pre-request Script Tests Settings
Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Beautify
"useraccount": "
"password": "
填写账号
Body Cookies Headers (6) Test Results
Status: 200 OK Time: 638 ms Size: 520 B
Save Response
Pretty Raw Preview Visualize JSON
得到临时token
"code": 0,
"msg": "成功",
"data": "
"traceId": "9228aca5-d7d3-43bb-9c45-9a1c5a2f96bf"


POST 获取临时token
POST 推送数据
No Environment
/ 实时API / json格式的数据 / 推送数据
4
Save
POST
http://localhost:8083/dataAccess/apiConfig/pushdata
填写推送数据的接口，地址在生成的文档中
Send
Params Authorization Headers (9) Body Pre-request Script Tests Settings
Cookies
User-Agent PostmanRuntime/7.28.4
Accept */*
Accept-Encoding gzip, deflate, br
Connection keep-alive
Authorization Bearer eyJhbGciOiJIUzI1NiJ9.eyJqdGkiOilyMjlwMzQwM...
Key Value Description
Response
填写临时token
Click Send to get a response


/ 实时API / json格式的数据 / 推送数据
Save
POST
http://192.168.11.130:8083/dataAccess/apiConfig/pushdata
Send
Params
Authorization
Headers (9)
Body
Pre-request Script
Tests
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
5.根据pdf文档中的结构将数据构造好
Beautify
"apiCode": 94,
"pushData": "{\"data\":[{\"id\":\"AE86\",\"name\":\"设备111\",\"age\":\"zhangsan\"},{\"id\":\"AE86\",\"name\":\"设备111\",\"age\":\"lisi\"}]}"
Body
Cookies
Headers (6)
Test Results
Status: 200 OK
Time: 1612 ms
Size: 478 B
Save Response
Pretty
Raw
Preview
Visualize
JSON
"code": 0,
"msg": "成功",
"data": "成功: 数据推送到[stg_asdfasd_tb_test281]: 成功,推送的条数为: 2; 数据同步到[ods]: stg同步数据到ods报错。--[本次同步的数据为正式数据]",
"traceId": "7c4a65a6-44c9-48c2-bf8d-5332d03a6039"
推送后的日志,也可以在数据接入首页查看

4. 查看应用日志





表名称	同步条数	同步开始时间	同步结束时间	同步结果	同步日志	调度类别
app	0	2022-07-13 17:21:36		运行成功	运行成功	手动调度
app	0	2022-07-13 17:21:36		运行成功	运行成功	手动调度
app	0	2022-07-13 17:21:36		运行成功	运行成功	手动调度
info	0	2022-07-11 12:58:38		运行成功	运行成功	手动调度
info	0	2022-07-11 12:58:38		运行成功	运行成功	手动调度
info	0	2022-07-11 12:58:38		运行成功	运行成功	手动调度
info	0	2022-07-11 12:58:38		运行成功	运行成功	手动调度
info	0	2022-07-11 12:58:38		运行成功	运行成功	手动调度
info	0	2022-07-11 12:58:38		运行成功	运行成功	手动调度
info	0	2022-07-11 12:58:38		运行成功	运行成功	手动调度