Atitit 数据查询法

目录

[1. 数据查询语言QL （推荐） 1](#_Toc28096)

[1.1. Sql 1](#_Toc15811)

[1.2. 对象查询语言（OQL） 1](#_Toc21959)

[1.3. Atitit QL查询语言总结Jpql Ongl GraphQL 1](#_Toc2948)

[1.4. 查询表达式css 选择器 xmlpath jsonpath 2](#_Toc28449)

[2. Css选择器 2](#_Toc31554)

[2.1. 属性选择filter 2](#_Toc8615)

[2.2. 多个属性选择器[]and逻辑 多个属性选择器并排写，选取结果为取交集。 2](#_Toc16144)

[2.3. 多个选择器 相当于多数据集 3](#_Toc8759)

[3. 使用数据查询api 3](#_Toc16136)

# 数据查询语言QL （推荐）

## Sql

## 对象查询语言（OQL）

搜索结果

网络搜索结果

JVM 对象查询语言（OQL）\_潘建南的博客-CSDN博客

https://blog.csdn.net › pange1991 › article › details

2018年8月24日 - OQL是用于查询Java堆的类SQL查询语言。OQL允许过滤/选择从Java堆中获取的信息。虽然HAT已经支持预定义的查询，例如“显示类X的所有

## Atitit QL查询语言总结Jpql Ongl GraphQL

目录

1. QL = Query Language, 是查询语言的简称 1

2. 模型 2

2.1. 内嵌语言执行 2

2.2. 语言互操作 2

3. 具体实现 2

3.1. Httpparam ql 2

3.2. Aql 受限自然语言QL 4.5G 2

3.3. Jpql 2

3.4. Spel （调用java等扩展，集合投影选择等） 2

3.5. Ongl （调用java等扩展，集合投影选择等） 2

3.6. 安全sql 3

4. 流程控制pcl类ql 3

4.1. SQL pcl 3

4.2. Js pcl 3

4.3. Apcl 3

5. 其他感觉可以放后的东东 4

5.1. Es query string search 4

5.2. Es json query 4

5.3. GraphQL 4

5.4. 认识APIJSON （ ） 5

6. 问题 7

6.1. 语言的内嵌解决，通过特定api解决 类似jdbc等 7

6.2. 流程控制的ast互相转换问题 7

7. ref 7

## 查询表达式css 选择器 xmlpath jsonpath

# Css选择器

## 属性选择filter

|  |  |  |  |
| --- | --- | --- | --- |
| [[](https://www.w3school.com.cn/cssref/selector_attribute_value.asp" \o "CSS [attribute=value] 选择器)*[attribute](https://www.w3school.com.cn/cssref/selector_attribute_value.asp" \o "CSS [attribute=value] 选择器)*[=](https://www.w3school.com.cn/cssref/selector_attribute_value.asp" \o "CSS [attribute=value] 选择器)*[value](https://www.w3school.com.cn/cssref/selector_attribute_value.asp" \o "CSS [attribute=value] 选择器)*[]](https://www.w3school.com.cn/cssref/selector_attribute_value.asp" \o "CSS [attribute=value] 选择器) | [target=\_blank] | 选择 target="\_blank" 的所有元素。 | 2 |
| [[](https://www.w3school.com.cn/cssref/selector_attribute_value_contain.asp" \o "CSS [attribute~=value] 选择器)*[attribute](https://www.w3school.com.cn/cssref/selector_attribute_value_contain.asp" \o "CSS [attribute~=value] 选择器)*[~=](https://www.w3school.com.cn/cssref/selector_attribute_value_contain.asp" \o "CSS [attribute~=value] 选择器)*[value](https://www.w3school.com.cn/cssref/selector_attribute_value_contain.asp" \o "CSS [attribute~=value] 选择器)*[]](https://www.w3school.com.cn/cssref/selector_attribute_value_contain.asp" \o "CSS [attribute~=value] 选择器) | [title~=flower] | 选择 title 属性包含单词 "flower" 的所有元素。 | 2 |
| [[](https://www.w3school.com.cn/cssref/selector_attribute_value_start.asp" \o "CSS [attribute|=value] 选择器)*[attribute](https://www.w3school.com.cn/cssref/selector_attribute_value_start.asp" \o "CSS [attribute|=value] 选择器)*[|=](https://www.w3school.com.cn/cssref/selector_attribute_value_start.asp" \o "CSS [attribute|=value] 选择器)*[value](https://www.w3school.com.cn/cssref/selector_attribute_value_start.asp" \o "CSS [attribute|=value] 选择器)*[]](https://www.w3school.com.cn/cssref/selector_attribute_value_start.asp" \o "CSS [attribute|=value] 选择器) | [lang|=en] | 选择 lang 属性值以 "en" 开头的所有元素。 |  |

## 多个属性选择器[]and逻辑 多个属性选择器并排写，选取结果为取交集。

jQuery("b[id=b1][vv=v3]");

1

多个过滤选择器:（不含eq）

在属性选择器前面连着写

jQuery("input:input:visible:odd[id\*=1]");

1

eq

写在后面

jQuery("input:input:visible:odd[id\*=1]:eq(0)");

1

## 多个选择器 相当于多数据集

之前的每一个例子的选择器视为一个完整的选择器。多个完整的选择器取并集需用逗号（,）分隔

jQuery("p[id],input:input:visible:odd[id\*=1]:eq(0)");

————————————————

版权声明：本文为CSDN博主「nayi\_224」的原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接及本声明。

原文链接：https://blog.csdn.net/nayi\_224/article/details/80361417

# 使用数据查询api