

Thư viện QueryString

I. Tổng quan

1. Mục đích

- query-string là 1 thư viện được tạo ra để chèn thêm 1 tầng phía trên các câu truy vấn database, nhằm giảm bớt độ phức tạp khi giao tiếp giữa các bên
- MySQL, MongoDB, Elasticsearch, ArrangoDB ... Mỗi database có cú pháp truy vấn, luật lệ của riêng mình. Nếu mỗi API tự handle logic truyền vào từ request body, từ đấy tạo nên câu truy vấn, sẽ khiến code rất dài và phức tạp, gây khó khăn cho việc maintain. Khi muốn đổi database, sẽ phải chỉnh sửa rất nhiều code, hoặc tạo hẳn API mới. Đồng thời, cũng tạo nên tình trạng nhiều API (được sinh ra để support cho các edge-case) cùng cho 1 mục đích search

2. Ví dụ sử dụng

- Về cơ bản, query-string tiếp nhận 1 string X và 1 bộ luật R, sẽ đi qua các bước tính toán để trả về 1 native database query Q
- R được quy định từ trước bên trong từng service sử dụng query-string. Người dùng chỉ việc biết về R, và gửi vào X thỏa mãn những ràng buộc
- Ví dụ:
 - $R = [$
 $\{ \text{field: "age", dataType: "STRING" } \},$
 $\{ \text{field: "income", dataType: "INTEGER" } \},$
 $\{ \text{field: "revenue", dataType: "DOUBLE" } \},$
 $\{ \text{field: "lendsAt", dataType: "DATE" } \}$
 $]$
 - lending portal cần lọc ra những user 18 tuổi trở lên, có thu nhập ít nhất 20 triệu
 $\Rightarrow X = \{ \text{"age"} \} \geq 18 \text{ and } \{ \text{"income"} \} \geq 20000000$
 - lending portal cần lọc ra những người có thu nhập ít nhất 20 triệu hoặc doanh thu ít nhất 15 triệu, và đã vay tiền trong tháng 7
 $\Rightarrow X = (\{ \text{"income"} \} \geq 20000000 \text{ or } \{ \text{"revenue"} \} \geq 15000000) \text{ and } (\{ \text{"lendsAt"} \} \geq \#2022-07-01T00:00:00Z \text{ and } \{ \text{"lendsAt"} \} < \#2022-08-01T00:00:00Z)$

3. Cú pháp

1. Toán tử

Loại	Biểu diễn	Chú thích
VARIABLE	{X}	đại diện cho biến X trong bộ luật
INTEGER	10, -3, 0	số nguyên
DOUBLE	1.32, -15.0	số thực
STRING	"welcome to nextpay"	xâu, chuỗi ký tự

DATE	#2022-07-01T12:00:00Z#	thời gian, ngày tháng
BOOLEAN	true, false	giá trị nhị phân (đúng / sai)
ARRAY INTEGER	[100, 200]	mảng số nguyên
ARRAY DOUBLE	[-1.1, 3.57, +10.0]	mảng số thực
ARRAY STRING	["mpos", "nextpay", "vimo"]	mảng chuỗi
ARRAY DATE	[#2022-07-01T12:00:00Z#]	mảng giá trị thời gian
ARRAY BOOLEAN	[true, false, true]	mảng giá trị nhị phân

2. Toán hạng

Loại	Biểu diễn	Độ ưu tiên	Chú thích
Not Clause	not	2	
Eq Filter	==	4	
Ne Filter	!=	4	
Gte Filter	>=	4	
Gt Filter	>	4	
Lte Filter	<=	4	
Lt Filter	<	4	so sánh biến với 1 giá trị
Like Filter	like	4	tìm chuỗi chứa đoạn ký tự được đề cập
In Filter	in	5	tìm giá trị nằm trong mảng
Nin Filter	nin	5	tìm giá trị không trong mảng
Range Filter	range	5	tìm kiếm giá trị trong khoảng cho trước
Fm Filter	===	6	cho bài toán full-match search trong Elasticsearch
And Clause	and	7	liên kết các mệnh đề đúng sai
Or Clause	or	8	liên kết các mệnh đề đúng sai

Lưu ý: chỉ số độ ưu tiên càng thấp, tức là càng được xử lý sớm

II. Một số tình huống sử dụng

- Truy vấn resource theo thông tin từ token
 - Với các API Backend, thường sẽ phải lấy ra thông tin của user từ token, rồi từ đấy

chọn ra các record thỏa mãn điều kiện. 1 số trường thông tin có thể lấy: username, userId ...

- Thường ta trả về cho user các record vẫn còn valid (như isActive = true, isAlive = true ...)

- Để đơn giản hóa code, phía client vẫn gửi request GET với query Q, còn code server sẽ dùng thêm condition C để trả về kết quả thỏa mãn

- Ví dụ:

- anh Hào cần xem danh sách các cấp dưới của mình
- phía client sẽ gọi request GET /users có access token T
- phía server sẽ phân giải T, lấy ra user_id = 10
- querystring được server sử dụng: "{manager_id} == 10"
- querystring được parse thành MySQL query: manager_id = 10
- full native MySQL query: SELECT * FROM users WHERE manager_id = 10

III. Một số mẫu truy vấn

- {objectClass} == "Hello"
- {percent} == 400
- {percent} == 400.756
- {isArchived} == false
- {isArchived} != false
- {createdAt} != #2020-12-19T01:39:22.713Z#
- {createdAt} <= #2020-12-19T01:39:22.713Z#
- {version} >= 8
- {name} in ["Name update unit test", "name", "VŨỜN AO CHUÔNG"]
- {createdAt} in [#2020-12-19T01:39:22.713+00:00#, #2020-12-19T01:39:30.062+00:00#, #2020-12-19T01:41:30.466+00:00#]
- {percent} in [30.6, 400, 10]
- {name} nin ["Name update unit test", "name", "VŨỜN AO CHUÔNG"]
- {percent} nin [30.6, 400, 10]
- ({percent} == 10 or {name} == "VŨỜN AO CHUÔNG") and ({appToCreate} == "5e86f1882c185a6b907057c1")
- ({percent} == 10 or {name} == "VŨỜN AO CHUÔNG") and ({appToCreate} == "5e86f1882c185a6b907057c1" or {objectClass} == "Hello 1")

IV. Lưu ý

1. **Không** hỗ trợ những truy vấn có dạng "... and true and ..."

Cách viết này có thể gây ra warning khi dùng SonarQube quét qua project, vì dễ trở thành SQL Injection. 1 nhầm lẫn giữa AND và OR sẽ deny cả câu truy vấn