**Snowflake semi structure data: VARIANT and lateral flatten**

Lets create a table inserting a semi structured dataset as below:

CREATE TABLE pets (v variant);

INSERT INTO pets SELECT PARSE\_JSON ('{"species":"dog", "name":"Fido", "is\_dog":"true"} ');

INSERT INTO pets SELECT PARSE\_JSON ('{"species":"cat", "name":"Bubby", "is\_dog":"false"}');

INSERT INTO pets SELECT PARSE\_JSON ('{"species":"cat", "name":"dog terror", "is\_dog":"false"}');

--Let’s query the table

SELECT a.v, b.key, b.value FROM pets a,LATERAL FLATTEN(input => a.v) b

WHERE b.value LIKE '%dog%';

+-------------------------+---------+--------------+

| V | KEY | VALUE |

|-------------------------+---------+--------------|

| { | species | "dog" |

| "is\_dog": "true", | | |

| "name": "Fido", | | |

| "species": "dog" | | |

| } | | |

| { | name | "dog terror" |

| "is\_dog": "false", | | |

| "name": "dog terror", | | |

| "species": "cat" | | |

| } | | |

+-------------------------+---------+--------------+

Explanation:

1. a and b are table aliases: a refers to the main table pets.

b is the alias for the result of the FLATTEN function applied to each a.v.

2. v is the column name in the pets table. It contains the JSON object (stored as VARIANT).

So a.v refers to the full JSON object like: {"species":"dog", "name":"Fido", "is\_dog":"true"}

3. FLATTEN(input => a.v):

This function breaks the JSON object into its individual key-value pairs.

For example, flattening {"species":"dog", "name":"Fido", "is\_dog":"true"} will produce 3 rows:

key: species, value: "dog"

key: name, value: "Fido"

key: is\_dog, value: "true"

Each row has several fields, but you’re using:

b.key: the JSON key

b.value: the value corresponding to that key

4. WHERE b.value LIKE '%dog%':

This filters for only those key-value pairs where the value contains the string "dog" (case-sensitive).

5. Output:

Each output row shows:

The full JSON object (a.v)

The key from that JSON that had a value matching %dog%

The matching value itself

Output:

v key value

{"species":"dog", "name":"Fido", "is\_dog":"true"} species "dog"

{"species":"dog", "name":"Fido", "is\_dog":"true"} is\_dog "true"

{"species":"cat", "name":"dog terror", "is\_dog":"false"} name "dog terror"

Why the second row matches "true":

Actually, it shouldn't — LIKE '%dog%' shouldn't match "true" unless there's a mistake. But assuming everything works as expected, it should only return rows where value literally contains "dog" — such as "dog", "dog terror", etc.

**So, this query:**

Extracts all key-value pairs from each JSON row in pets

Filters those pairs where the value contains "dog"

Shows the original JSON (a.v), and the specific key and value that matched

For detail: <https://docs.snowflake.com/en/user-guide/querying-semistructured>