Performing Aggregations

Let us understand how to aggregate the data.

- We can perform global aggregations as well as aggregations by key.
- Global Aggregations
    - Get total number of orders.
    - Get revenue for a given order id.
    - Get number of records with order_status either COMPLETED or CLOSED.
- Aggregations by key - using `GROUP BY`
    - Get number of orders by date or status.
    - Get revenue for each order_id.
    - Get daily product revenue (using order date and product id as keys).
- We can also use `HAVING` clause to apply filtering on top of aggregated data.
    - Get daily product revenue where revenue is greater than $500 (using order date and product id as keys).
- Rules while using `GROUP BY`.
    - We can have the columns which are specified as part of `GROUP BY` in `SELECT` clause.
    - On top of those, we can have derived columns using aggregate functions.
    - We cannot have any other columns that are not used as part of `GROUP BY` or derived column using non aggregate functions.
    - We will not be able to use aggregate functions or aliases used in the select clause as part of the where clause.
    - If we want to filter based on aggregated results, then we can leverage `HAVING` on top of `GROUP BY` (specifying `WHERE` is not an option)
- Typical query execution - FROM -> WHERE -> GROUP BY -> SELECT

```
%load_ext sql
```

```
The sql extension is already loaded. To reload it, use:
  %reload_ext sql
```

```
%env
DATABASE_URL=postgresql://itversity_retail_user:retail_password@localhost:5432/itversity_retail_db
```

```
env:
DATABASE_URL=postgresql://itversity_retail_user:retail_password@localhost:5432/itversity_retail_db
```

```
%sql SELECT count(order_id) FROM orders
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
1 rows affected.
```

count
68883

```
%sql SELECT count(DISTINCT order_date) FROM orders
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
1 rows affected.
```

count
364

```
%%sql

SELECT *
FROM order_items
WHERE order_item_order_id = 2
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
3 rows affected.
```

order_item_id	order_item_order_id	order_item_product_id	order_item_quantity	order_item_sub
2	2	1073	1	19
3	2	502	5	2
4	2	403	1	12

```
%%sql

SELECT round(sum(order_item_subtotal::numeric), 2) AS order_revenue
FROM order_items
WHERE order_item_order_id = 2
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
1 rows affected.
```

order_revenue
579.98

```
%%sql

SELECT count(1)
FROM orders
WHERE order_status IN ('COMPLETE', 'CLOSED')
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
1 rows affected.
```

count
30455

```
%%sql

SELECT order_date,
    count(1)
FROM orders
GROUP BY order_date
ORDER BY order_date
LIMIT 10
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
10 rows affected.
```

order_date	count
2013-07-25 00:00:00	143
2013-07-26 00:00:00	269
2013-07-27 00:00:00	202
2013-07-28 00:00:00	187
2013-07-29 00:00:00	253
2013-07-30 00:00:00	227
2013-07-31 00:00:00	252
2013-08-01 00:00:00	246
2013-08-02 00:00:00	224
2013-08-03 00:00:00	183

```
%%sql

SELECT order_status,
    count(1) AS status_count
FROM orders
GROUP BY order_status
ORDER BY order_status
LIMIT 10
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
9 rows affected.
```

order_status	status_count
CANCELED	1428
CLOSED	7556
COMPLETE	22899
ON_HOLD	3798
PAYMENT_REVIEW	729
PENDING	7610
PENDING_PAYMENT	15030
PROCESSING	8275
SUSPECTED_FRAUD	1558

```
%%sql

SELECT order_item_order_id,
    sum(order_item_subtotal) AS order_revenue
FROM order_items
GROUP BY order_item_order_id
LIMIT 10
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
10 rows affected.
```

order_item_order_id	order_revenue
44127	179.97
26264	334.96000000000004
37876	699.97
55864	600.94
31789	129.99
56903	479.97
40694	1129.75
48663	969.9200000000001
47216	1219.89
37922	1029.9

Error

This query using `round` will fail as `sum(order_item_subtotal)` will not return the data accepted by `round`. We have to convert the data type of `sum(order_item_subtotal)` to `numeric`.

```
%%sql

SELECT order_item_order_id,
    round(sum(order_item_subtotal), 2) AS order_revenue
FROM order_items
GROUP BY order_item_order_id
LIMIT 10
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
(psycopg2.errors.UndefinedFunction) function round(double precision, integer) does not exist
LINE 1: SELECT order_item_order_id, round(sum(order_item_subtotal), ...
                                    ^
HINT:  No function matches the given name and argument types. You might need to add explicit
type casts.

[SQL: SELECT order_item_order_id, round(sum(order_item_subtotal), 2) AS order_revenue
FROM order_items
GROUP BY order_item_order_id
LIMIT 10]
(Background on this error at: http://sqlalche.me/e/13/f405)
```

```
%%sql

SELECT order_item_order_id,
    round(sum(order_item_subtotal)::numeric, 2) AS order_revenue
FROM order_items
GROUP BY order_item_order_id
LIMIT 10
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
10 rows affected.
```

order_item_order_id	order_revenue
44127	179.97
26264	334.96
37876	699.97
55864	600.94
31789	129.99
56903	479.97
40694	1129.75
48663	969.92
47216	1219.89
37922	1029.90

```
%%sql

SELECT o.order_date,
    oi.order_item_product_id,
    round(sum(oi.order_item_subtotal::numeric), 2) AS revenue
FROM orders o JOIN order_items oi
    ON o.order_id = oi.order_item_order_id
WHERE o.order_status IN ('COMPLETE', 'CLOSED')
GROUP BY o.order_date,
    oi.order_item_product_id
LIMIT 10
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
10 rows affected.
```

order_date	order_item_product_id	revenue
2013-07-25 00:00:00	24	319.96
2013-07-25 00:00:00	93	74.97
2013-07-25 00:00:00	134	100.00
2013-07-25 00:00:00	191	5099.49
2013-07-25 00:00:00	226	599.99
2013-07-25 00:00:00	365	3359.44
2013-07-25 00:00:00	403	1949.85
2013-07-25 00:00:00	502	1650.00
2013-07-25 00:00:00	572	119.97
2013-07-25 00:00:00	625	199.99

Note

We cannot use the aliases in select clause in `WHERE`. In this case **revenue** cannot be used in `WHERE` clause.

```
%%sql

SELECT o.order_date,
    oi.order_item_product_id,
    round(sum(oi.order_item_subtotal::numeric), 2) AS revenue
FROM orders o JOIN order_items oi
    ON o.order_id = oi.order_item_order_id
WHERE o.order_status IN ('COMPLETE', 'CLOSED')
    AND revenue >= 500
GROUP BY o.order_date,
    oi.order_item_product_id
LIMIT 10
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
(psycopg2.errors.UndefinedColumn) column "revenue" does not exist
LINE 5:     AND revenue >= 500
                ^

[SQL: SELECT o.order_date, oi.order_item_product_id,
round(sum(oi.order_item_subtotal::numeric), 2) AS revenue
FROM orders o JOIN order_items oi
    ON o.order_id = oi.order_item_order_id
WHERE o.order_status IN ('COMPLETE', 'CLOSED')
    AND revenue >= 500
GROUP BY o.order_date,
    oi.order_item_product_id
LIMIT 10]
(Background on this error at: http://sqlalche.me/e/13/f405)
```

Note

We cannot use aggregate functions in `WHERE` clause.

```
%%sql

SELECT o.order_date,
    oi.order_item_product_id,
    round(sum(oi.order_item_subtotal::numeric), 2) AS revenue
FROM orders o JOIN order_items oi
    ON o.order_id = oi.order_item_order_id
WHERE o.order_status IN ('COMPLETE', 'CLOSED')
    AND round(sum(oi.order_item_subtotal::numeric), 2) >= 500
GROUP BY o.order_date,
    oi.order_item_product_id
LIMIT 10
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
(psycopg2.errors.GroupingError) aggregate functions are not allowed in WHERE
LINE 5:     AND round(sum(oi.order_item_subtotal::numeric), 2) >= 50...
                      ^

[SQL: SELECT o.order_date, oi.order_item_product_id,
round(sum(oi.order_item_subtotal::numeric), 2) AS revenue
FROM orders o JOIN order_items oi
    ON o.order_id = oi.order_item_order_id
WHERE o.order_status IN ('COMPLETE', 'CLOSED')
    AND round(sum(oi.order_item_subtotal::numeric), 2) >= 500
GROUP BY o.order_date,
    oi.order_item_product_id
LIMIT 10]
(Background on this error at: http://sqlalche.me/e/13/f405)
```

```
%%sql

SELECT o.order_date,
    oi.order_item_product_id,
    round(sum(oi.order_item_subtotal::numeric), 2) AS revenue
FROM orders o JOIN order_items oi
    ON o.order_id = oi.order_item_order_id
WHERE o.order_status IN ('COMPLETE', 'CLOSED')
GROUP BY o.order_date,
    oi.order_item_product_id
HAVING round(sum(oi.order_item_subtotal::numeric), 2) >= 500
ORDER BY o.order_date, revenue DESC
LIMIT 25
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
25 rows affected.
```

order_date	order_item_product_id	revenue
2013-07-25 00:00:00	1004	5599.72
2013-07-25 00:00:00	191	5099.49
2013-07-25 00:00:00	957	4499.70
2013-07-25 00:00:00	365	3359.44
2013-07-25 00:00:00	1073	2999.85
2013-07-25 00:00:00	1014	2798.88
2013-07-25 00:00:00	403	1949.85
2013-07-25 00:00:00	502	1650.00
2013-07-25 00:00:00	627	1079.73
2013-07-25 00:00:00	226	599.99
2013-07-26 00:00:00	1004	10799.46
2013-07-26 00:00:00	365	7978.67
2013-07-26 00:00:00	957	6899.54
2013-07-26 00:00:00	191	6799.32
2013-07-26 00:00:00	1014	4798.08
2013-07-26 00:00:00	502	4250.00
2013-07-26 00:00:00	1073	3999.80
2013-07-26 00:00:00	403	3249.75
2013-07-26 00:00:00	627	3039.24
2013-07-27 00:00:00	1004	9599.52
2013-07-27 00:00:00	191	5999.40
2013-07-27 00:00:00	957	5699.62
2013-07-27 00:00:00	1073	5399.73
2013-07-27 00:00:00	365	5099.15
2013-07-27 00:00:00	502	5050.00

```
%%sql

SELECT count(1) FROM (
    SELECT o.order_date,
        oi.order_item_product_id,
        round(sum(oi.order_item_subtotal::numeric), 2) AS revenue
    FROM orders o JOIN order_items oi
        ON o.order_id = oi.order_item_order_id
    WHERE o.order_status IN ('COMPLETE', 'CLOSED')
    GROUP BY o.order_date,
        oi.order_item_product_id
) q
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
1 rows affected.
```

count
9120

```
%%sql

SELECT count(1) FROM (
    SELECT o.order_date,
        oi.order_item_product_id,
        round(sum(oi.order_item_subtotal::numeric), 2) AS revenue
    FROM orders o JOIN order_items oi
        ON o.order_id = oi.order_item_order_id
    WHERE o.order_status IN ('COMPLETE', 'CLOSED')
    GROUP BY o.order_date,
        oi.order_item_product_id
    HAVING round(sum(oi.order_item_subtotal::numeric), 2) >= 500
) q
```

```
 * postgresql://itversity_retail_user:***@localhost:5432/itversity_retail_db
1 rows affected.
```

count
3339

By Durga Gadiraju