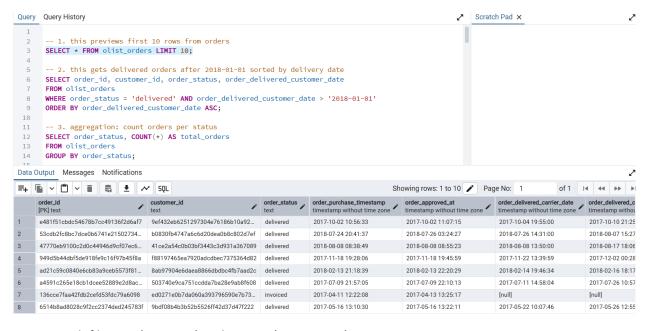
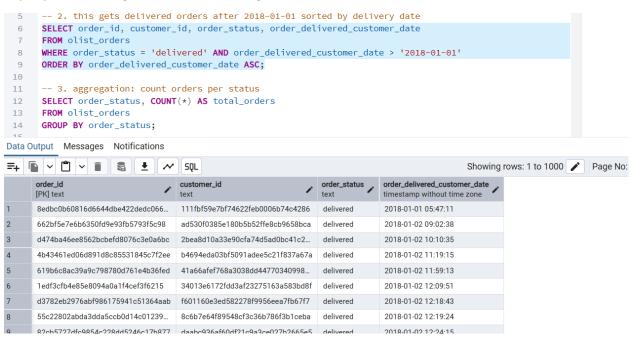
- 4. (30 points) Import your chosen dataset.
- a. Check that the data structures were imported correctly; if necessary, fix them manually (data types, column names, remove unnecessary columns).
- b. Perform basic checks and queries:

SELECT * FROM table LIMIT 10;



a query with filtering (WHERE) and sorting (ORDER BY);



aggregation (GROUP BY) with functions COUNT, AVG, MIN, MAX;

```
-- 3. aggregation: count orders per status
 11
       SELECT order_status, COUNT(*) AS total_orders
 12
       FROM olist_orders
 13
       GROUP BY order_status;
 14
 15
       -- 4. aggregation with AVG: aerage payment value per pag
 16
 17
       SELECT payment_type, AVG(payment_value) AS avg_payment
       FROM olist_order_payments
 18
       GROUP BY payment_type;
 19
 20
       -- 5. aggregation with MIN/MAX: Earliest and latest pur-
 21
       SELECT MIN(order_purchase_timestamp) AS first_order,
 22
               MAY/ardar nurahasa timaatama) AC laat ar
Data Output Messages Notifications
=+
                            +
                                      SQL
                  total_orders
     order_status
     text
                  bigint
1
     shipped
                         1107
2
     unavailable
                          609
     invoiced
3
                          314
     created
                            5
4
                            2
5
      approved
      processing
                          301
6
      delivered
                         96478
7
```

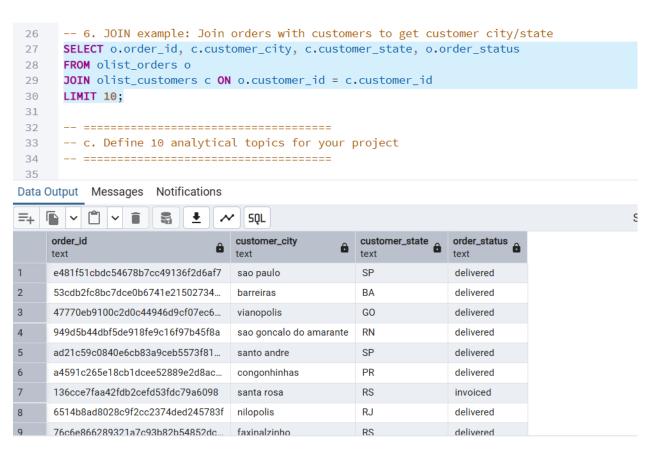
625

canceled

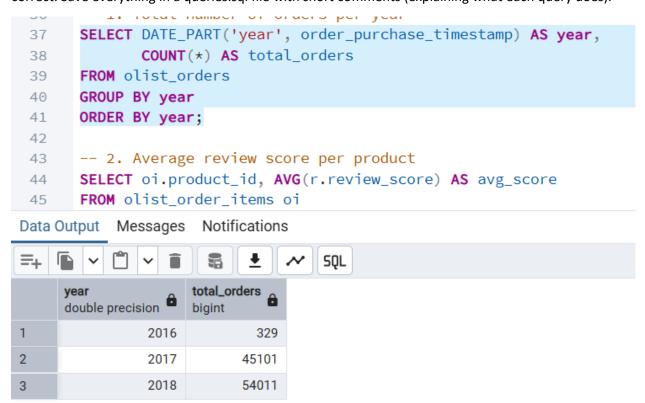
8

```
21 -- 5. aggregation with MIN/MAX: Earliest and latest purchase date
      SELECT MIN(order_purchase_timestamp) AS first_order,
  22
  23
            MAX(order_purchase_timestamp) AS last_order
  24
      FROM olist_orders;
  25
  26 -- 6. JOIN example: Join orders with customers to get customer city/state
 Data Output Messages Notifications
 =+ | • | • | • | • |
                   first order
                         last_order
     timestamp without time zone
                        timestamp without time zone
     2016-09-04 21:15:19
                         2018-10-17 17:30:18
 15
        -- 4. aggregation with AVG: aerage payment value per pay
 16
        SELECT payment_type, AVG(payment_value) AS avg_payment
 17
 18
        FROM olist_order_payments
 19
        GROUP BY payment_type;
 20
        -- 5. aggregation with MIN/MAX: Earliest and latest purc
 21
        SELECT MIN(order_purchase_timestamp) AS first_order,
 22
 23
                 MAX(order_purchase_timestamp) AS last_order
 24
        FROM olist_orders;
 25
 26
        -- 6. JOIN example: Join orders with customers to get cu
Data Output
             Messages
                         Notifications
                                          SQL
=₊
      payment_type
                            avg_payment
      character varying (50)
                            numeric
      not defined
1
                             0.00000000000000000000
2
      boleto
                               145.0344354023453296
3
      debit card
                               142.5701700457815566
      voucher
                                65.7033541125541126
4
5
      credit_card
                               163.3190206393645420
```

at least one JOIN between tables.

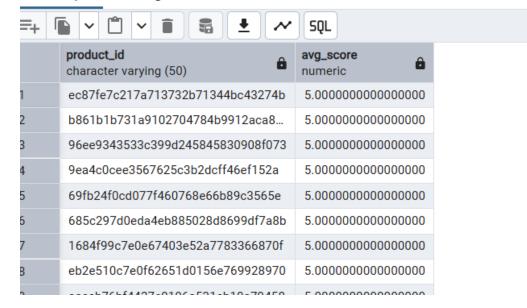


c. Define 10 analytical topics for your project (e.g., average scores per subject, number of admissions by year/department, sales by region, etc.). Write an SQL query for each topic and make sure the results are correct. Save everything in a queries.sql file with short comments (explaining what each query does).

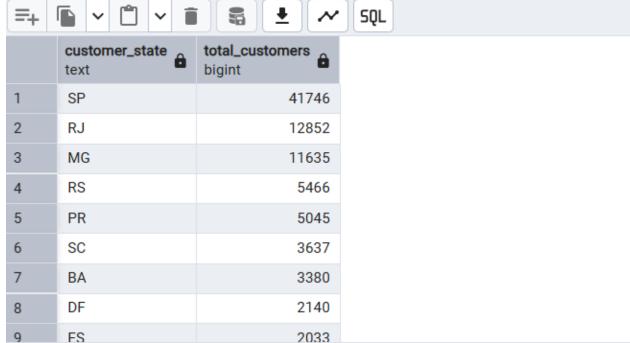


```
SELECT oi.product_id, AVG(r.review_score) AS avg_score
44
45
     FROM olist_order_items oi
     JOIN olist_order_reviews r ON oi.order_id = r.order_id
46
     GROUP BY oi.product_id
47
     ORDER BY avg_score DESC;
48
49
     -- 3. Top 10 states by number of customers
50
     SELECT customer_state, COUNT(*) AS total_customers
51
     FROM olist_customers
52
```

Data Output Messages Notifications

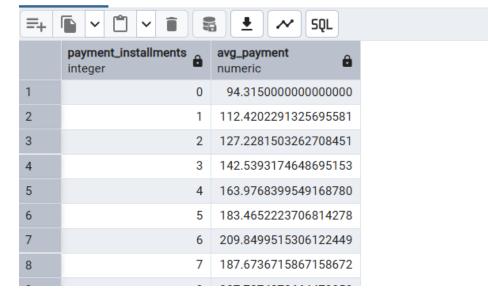


```
-- 3. Top 10 states by number of customers
 50
      SELECT customer_state, COUNT(*) AS total_customers
 51
      FROM olist customers
 52
 53
      GROUP BY customer_state
      ORDER BY total_customers DESC
 54
      LIMIT 10;
 55
 56
     -- 4. Average payment value per installment count
 57
      SELECT payment_installments, AVG(payment_value) AS avg
 58
      FROM olist_order_payments
 59
Data Output Messages Notifications
=+
                                   SQL
```



```
57
     -- 4. Average payment value per installment count
     SELECT payment_installments, AVG(payment_value) AS avg_payment
58
     FROM olist_order_payments
59
     GROUP BY payment_installments
60
     ORDER BY payment_installments;
61
62
     -- 5. Number of orders by delivery status (delivered, shipped, cance
63
     SELECT order_status, COUNT(*) AS total
64
65
     FROM olist_orders
```

Data Output Messages Notifications



```
-- o. Number of orders by decivery scalus (decive
 CO
       SELECT order_status, COUNT(*) AS total
 64
       FROM olist orders
 65
       GROUP BY order_status
 66
       ORDER BY total DESC;
 67
 68
       -- 6. Average delivery time (customer date - purc
 69
       SELECT AVG(order_delivered_customer_date - order_
 70
       FROM olist orders
 71
       WHERE order_status = 'delivered';
 72
 73
 74
       -- 7. Top 10 most sold products
       SELECT oi.product_id, COUNT(*) AS total_sold
 75
            Messages Notifications
Data Output
=+
                                       SQL.
     order_status
                   total
                   bigint 🎰
     text
1
      delivered
                     96478
2
      shipped
                      1107
3
      canceled
                       625
      unavailable
                       609
4
5
      invoiced
                       314
      processing
                       301
6
7
      created
                        5
                        2
8
      approved
```

```
69
     -- 6. Average delivery time (customer date - purchase date)
 70 SELECT AVG(order_delivered_customer_date - order_purchase_timestamp) AS avg_delivery_days
 71
      FROM olist_orders
 72
      WHERE order_status = 'delivered';
 73
 74
     -- 7. Top 10 most sold products
 75 SELECT oi.product_id, COUNT(*) AS total_sold
 76
     FROM olist_order_items oi
 77
      GROUP BY oi.product_id
     ORDER BY total_sold DESC
 79 LIMIT 10;
Data Output Messages Notifications
=+ 🖺 ∨ 📋 ∨ 🝵 🔱 👲 💉 SQL
                                                                                          Showing rows: 1
    avg_delivery_days
  12 days 13:23:49.957272
```

```
-- 7. Top 10 most sold products
 74
       SELECT oi.product_id, COUNT(*) AS total_sold
 75
       FROM olist_order_items oi
 76
 77
       GROUP BY oi.product_id
       ORDER BY total_sold DESC
 78
       LIMIT 10;
 79
 80
 81
       -- 8. Seller performance: total sales value per seller
       SELECT oi.seller_id, SUM(oi.price) AS total_sales
 82
       FROM olist_order_items oi
 83
       GROUP BY oi.seller_id
 84
       ORDER BY total_sales DESC
 85
 26
       ITMTT 10.
Data Output Messages Notifications
=+
                                       SQL
                                     total_sold
     product_id
     character varying (50)
                                      bigint
1
      aca2eb7d00ea1a7b8ebd4e68314663af
                                            527
2
      99a4788cb24856965c36a24e339b60...
                                            488
      422879e10f46682990de24d770e7f83d
3
                                            484
      389d119b48cf3043d311335e499d9c6b
                                            392
4
5
      368c6c730842d78016ad823897a372...
                                            388
6
      53759a2ecddad2bb87a079a1f1519f73
                                            373
      d1c427060a0f73f6b889a5c7c61f2ac4
                                            343
7
```

323

281

53b36df67ebb7c41585e8d54d6772e08

154e7e31ebfa092203795c972e5804a6

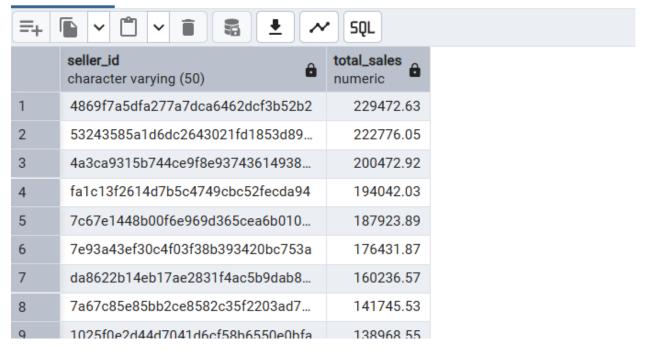
Query complete 00:00:00.139

8

Total rows: 10

```
82
     SELECT oi.seller_id, SUM(oi.price) AS total_sales
     FROM olist_order_items oi
83
     GROUP BY oi.seller id
84
     ORDER BY total_sales DESC
85
     LIMIT 10;
86
87
     -- 9. Average freight value by state
88
     SELECT c.customer_state, AVG(oi.freight_value) AS avg_fre
89
     FROM olist_order_items oi
90
     JOIN olist_orders o ON oi.order_id = o.order_id
91
     JOIN olist_customers c ON o.customer_id = c.customer_id
92
```

Data Output Messages Notifications



```
SELECT c.customer_state, AVG(oi.freight_value) AS avg_freight
 89
       FROM olist_order_items oi
 90
       JOIN olist_orders o ON oi.order_id = o.order_id
 91
       JOIN olist_customers c ON o.customer_id = c.customer_id
 92
       GROUP BY c.customer_state
 93
       ORDER BY avg_freight DESC;
 95
       -- 10. Number of reviews per score (distribution of ratings)
 96
       SELECT review_score, COUNT(*) AS review_count
 97
       FROM olist_order_reviews
 98
       GROUP BY review_score
 99
       ORDER BY review_score;
100
Data Output Messages Notifications
=+
                                     SQL
     customer_state
                     avg_freight
     text
                    numeric
     RR
                     42.9844230769230769
     PB
                     42.7238039867109635
     RO
                     41.0697122302158273
     AC
                     40.0733695652173913
                     39.1479704797047970
     PΙ
                     38.2570024271844660
     MA
     TO
                     37.2466031746031746
                     36.6531688311688312
     SE
                     35.8436711711711712
       -- 10. Number of reviews per score (distribution of ratings)
 96
 97
        SELECT review_score, COUNT(*) AS review_count
        FROM olist_order_reviews
 98
        GROUP BY review score
 99
        ORDER BY review_score;
100
101
Data Output
            Messages
                       Notifications
                                       SQL
=+
      review_score
                    review_count
                    bigint
      integer
1
                 1
                           11424
2
                 2
                            3151
3
                 3
                            8179
4
                 4
                           19142
```

5

5

57328