

# Advanced Databases

## INZ000109P

### Project

#### 12 - Summary (spec.)

Group: A2

Mustafa Tayyip BAYRAM 257639

Furkan ÖCALAN 257638

Bauyrzhan MARAT 258395

- **FIRST QUERY**

```
SELECT product_id,brand_name ,product_name, model_year, list_price, category_name FROM
(
  (SELECT * FROM
    (SELECT * FROM
      ( SELECT * from MUTABAY.products prod_outer
        where 1 = (
          SELECT COUNT(Distinct list_price)
          FROM MUTABAY.products prod_inner
          WHERE prod_outer.brand_id = prod_inner.brand_id
          AND prod_outer.list_price < prod_inner.list_price
        )
      ) prod
    FULL OUTER JOIN
      MUTABAY.brands brands on brands.brand_id = prod.brand_id
    ) prod_brand
  FULL OUTER JOIN
    MUTABAY.categories categories on categories.category_id = prod_brand.category_id
  )prod_brand_cat
)
where list_price > 980000 AND model_year < 2020

GROUP BY product_id,brand_name ,product_name, model_year, list_price, category_name
ORDER BY product_id ASC;
```

## EXECUTION TIME COMPARISON

TRYING	WITHOUT IMPROVEMENTS	AFTER PARTITIONS	AFTER INDEXES
1	1.779	1.318	0.410
2	1.73	0.918	0.424
3	1.822	0.944	0.404
4	1.751	0.921	0.408
5	1.73	0.918	0.404
6	1.74	0.942	0.411
7	1.724	0.946	0.425
8	1.744	0.912	0.408
9	1.742	0.893	0.415
10	1.743	0.94	0.408
MAX	1.822	1.318	0.425
MIN	1.73	0.893	0.404
AVERAGE	1.7505	0.9652	0.410
COSTS	23220	637K	22733

WITHOUT ANY IMPROVEMENTS AVERAGE TIME = 1.7505

PARTITIONED AVG. TIME = 0.9652

INDEXED AVG. TIME = 0.410

COLUMNAR STORAGE AVG. TIME = 1.637

- **SECOND QUERY**

```
SELECT first_name,last_name, active, salary, stores.store_name, stores.city, stores.state
FROM
(
    SELECT staffs.*,avg(salary) over (partition by store_id) as avgSalary
    from MUTABAY.staffs staffs
)staffs
FULL OUTER JOIN MUTABAY.stores stores
ON staffs.store_id=stores.store_id
FULL OUTER JOIN MUTABAY.orders orders
ON stores.store_id = orders.store_id
FULL OUTER JOIN MUTABAY.order_items order_items
ON orders.order_id = order_items.order_id
FULL OUTER JOIN MUTABAY.products products
ON order_items.product_id = products.product_id
WHERE staffs.salary < staffs.avgsalary or order_items.discount > 0.05 OR customer_id > 1500
GROUP BY store_name, first_name, salary, city, state, last_name, active
having (avg(staffs.salary) > 1000 OR state IS NOT NULL) OR (city = 'Aberdeen' AND active = 1)
ORDER BY store_name asc;
```

TRYING	WITHOUT IMPROVEMENTS	AFTER PARTITIONS	AFTER COLUMNAR STORAGE
1	1.67	1.352	1.475
2	1.208	0.863	1.038
3	1.188	0.565	1.202
4	1.194	0.569	1.347
5	1.217	0.588	1.084
6	1.164	0.587	1.073
7	1.314	0.573	1.064
8	1.357	0.582	1.104
9	3.513	0.603	1.074
10	3.329	0.607	1.074
MAX	3.513	1.352	1.475
MIN	1.164	0.565	1.038
AVERAGE	1.7514	0.6889	1.1535
COSTS	165K	14194	165K

**WITHOUT ANY IMPROVEMENTS AVERAGE TIME = 1.751**

**PARTITIONED AVG. TIME = 0.6889**

**INDEXED AVG. TIME = 1.447**

**COLUMNAR STORAGE AVG. TIME = 1.1535**

- **THIRD QUERY**

```

SELECT products.product_id, products.product_name, products.list_price,
       orders.order_date, orders.required_date, orders.order_status,
       categories.category_name, brands.brand_name, quantity_id,
       discount, COUNT(quantity_id) quantity_count, (quantity_id * discount * products.list_price) total
FROM MUTABAY.order_items order_items
full outer join MUTABAY.orders orders on
    (orders.order_id = order_items.order_id)
full outer join MUTABAY.products products on
    (products.product_id = order_items.product_id)
full outer join MUTABAY.brands brands on
    (brands.brand_id = products.brand_id)
full outer join MUTABAY.categories categories on
    (categories.category_id = products.category_id)
full outer join MUTABAY.staffs staffs on
    (staffs.staff_id = orders.staff_id)
WHERE (shipped_date - order_date) > 2 OR
      (shipped_date - order_date) = 0 OR
      (shipped_date - order_date) < 0
GROUP BY products.product_id, products.product_name, products.list_price,
         orders.order_date, orders.required_date, orders.order_status,
         categories.category_name, brands.brand_name, quantity_id,
         discount
having AVG(list_price) > 10000
Order by order_status;

```

TRYING	WITHOUT IMPROVEMENTS	AFTER INDEXES	AFTER COLUMNAR STORAGE
1	0.713	0.543	0.618
2	0.707	0.571	0.538
3	0.722	0.525	0.517
4	0.952	0.500	0.534
5	0.912	0.584	0.539
6	0.772	0.560	0.547
7	0.822	0.638	0.531
8	0.795	0.558	0.534
9	0.714	0.508	0.537
10	0.926	0.505	0.535
MAX	0.952	0.638	0.618
MIN	0.707	0.5	0.517
AVERAGE	0.7985	0.549	0.543
COSTS	9204	9204	9384

WITHOUT ANY IMPROVEMENTS AVERAGE TIME = 0.7985

**PARTITIONED AVG. TIME = 0.7454**

**INDEXED AVG. TIME = 0.549**

**COLUMNAR STORAGE AVG. TIME = 0.543**

- FOURTH QUERY**

update MUTABAY.products products

set model\_year =

```
(
  select distinct(product_id) as total_product
  from MUTABAY.order_items order_items
  full outer join MUTABAY.orders orders on
    (orders.order_id = order_items.order_id)
  full outer join MUTABAY.stores stores on
    (orders.store_id = stores.store_id)
  where stores.store_id =
    (
      select store_id from MUTABAY.stocks
      full outer join MUTABAY.products products on
        (stocks.product_id = products.product_id)
      where ((model_year between 2020 and 1958) or (ROUND(list_price) < 990.000))
      or UPPER ( SUBSTR(product_name,2,3 ) )LIKE 'D%'
      fetch first 1 rows only
    )
  fetch first 1 rows only
);
```

TRYING	WITHOUT IMPROVEMENTS	AFTER PARTITIONS	AFTER INDEXES
1	1.379	0.63	0.657
2	0.813	0.76	0.440
3	0.85	0.394	0.446
4	0.741	0.945	1.402
5	1.251	0.593	0.451
6	0.653	0.598	0.439
7	0.798	0.595	0.442
8	0.666	0.757	0.872
9	0.731	0.407	0.765
10	0.667	0.618	0.468
MAX	1.379	0.945	1.402
MIN	0.653	0.394	0.442
AVERAGE	0.8549	0.6297	0.726
COSTS	996	3410	891

**WITHOUT ANY IMPROVEMENTS AVERAGE TIME = 0.8549**

**PARTITIONED AVG. TIME = 0.6297**

**INDEXED AVG. TIME = 0.726**

**COLUMNAR STORAGE AVG. TIME = 1.1459**

- FIFTH QUERY**

update MUTABAY.order\_items set quantity\_id =

```
(
  select quantity_id from MUTABAY.products products
  full outer join MUTABAY.order_items order_items on order_items.product_id = products.product_id
  full outer join MUTABAY.stocks stocks on stocks.product_id = products.product_id
  full outer join MUTABAY.orders orders on order_items.order_id=orders.order_id
  full outer join MUTABAY.customers customers on orders.customer_id=customers.customer_id
  full outer join MUTABAY.stores stores on orders.store_id=stores.store_id
  full outer join MUTABAY.staffs staffs on orders.staff_id=staffs.staff_id
  where products.product_id in
  (
    Select product_id from MUTABAY.order_items where order_id in
    (
      Select order_id from MUTABAY.orders
      WHERE
      (order_status = 1 AND ( shipped_date - required_date = 1 ))
      OR
      (order_status = 2 AND (shipped_date - required_date = 0 ))
    )
  )
  )fetch next 1 rows only
);
```

TRYING	WITHOUT IMPROVEMENTS	AFTER PARTITIONS	AFTER INDEXES
1	3.365	1.504	2.782
2	3.598	1.372	0.986
3	4.786	1.404	0.971
4	2.713	1.443	2.177
5	2.517	1.612	0.968
6	3.69	1.022	0.980
7	2.986	1.356	1.417
8	2.786	1.331	0.973
9	1.884	1.341	0.973
10	2.616	1.534	1.905
MAX	4.786	1.612	2.782
MIN	1.884	1.022	0.968
AVERAGE	3.0941	1.3919	1.413

<b>COSTS</b>	<b>5372</b>	<b>4091</b>	<b>5372</b>
--------------	-------------	-------------	-------------

**WITHOUT ANY IMPROVEMENTS AVERAGE TIME = 3.0941**

**PARTITIONED AVG. TIME = 1.3919**

**INDEXED AVG. TIME = 1.413**

**COLUMNAR STORAGE AVG. TIME = 2.6716**

- SIXTH QUERY**

```

UPDATE MUTABAY.stores
SET MUTABAY.stores.store_name = (
    SELECT store_name FROM MUTABAY.stores stores
    INNER JOIN
    (
        SELECT order_i.staff_id, first_name, last_name, phone, email, order_i.store_id, manager_id, active,
        salary ,
        order_i.item_id ,order_i.product_id ,order_i.quantity_id ,order_i.discount ,order_i.customer_id
        ,order_i.order_status ,
        order_i.order_date ,order_i.required_date ,order_i.shipped_date
        FROM MUTABAY.staffs staffs
        FULL OUTER JOIN
        (
            SELECT orders.order_id, item_id, product_id, quantity_id, discount, orders.customer_id,
            orders.order_status,
            orders.order_date, orders.required_date, orders.shipped_date, orders.store_id, orders.staff_id
            FROM MUTABAY.order_items order_items
            FULL OUTER JOIN MUTABAY.orders orders
            ON orders.order_id = order_items.order_id
            WHERE ORDERS.ORDER_ID IN (SELECT ORDER_ID FROM MUTABAY.ORDER_ITEMS WHERE
            DISCOUNT > (SELECT AVG(DISCOUNT) FROM MUTABAY.ORDER_ITEMS))
            OR
            (order_status = 2)
        ) order_i
        ON order_i.staff_id = staffs.staff_id
        WHERE (discount > 0.48 AND discount < 0.05) AND salary > 5000
        OR
        (active = 1 AND discount = 0.4)
        )order_i_staff
        ON order_i_staff.store_id = stores.store_id
        WHERE street='1 Fremont Point' or STATE IS NOT NULL
        fetch first 1 rows only
    );

```

<b>TRYING</b>	<b>WITHOUT IMPROVEMENTS</b>	<b>AFTER PARTITIONS</b>	<b>AFTER COLUMNAR STORAGE</b>
<b>1</b>	<b>8.448</b>	<b>4.889</b>	<b>7.341</b>

2	6.416	4.975	6.312
3	6.016	4.844	5.585
4	5.839	5.359	5.961
5	5.698	4.942	6.362
6	6.269	4.85	6.576
7	5.893	4.449	5.841
8	5.741	4.873	5.623
9	5.745	4.921	5.669
10	5.763	4.725	5.699
MAX	8.448	5.359	7.341
MIN	5.698	4.449	5.585
AVERAGE	6.1828	4.8827	6.0969
COSTS	1105	5228	957

WITHOUT ANY IMPROVEMENTS AVERAGE TIME = 6.1828

PARTITIONED AVG. TIME = 4.8827

INDEXED AVG. TIME = 20.1

COLUMNAR STORAGE AVG. TIME = 6.0969