Advanced Databases
INZ000109P
Project

Assignment 4 - Database workload (dev.)

Group: A2

Furkan ÖCALAN 257638

Mustafa Tayyip BAYRAM 257639

❖ Clear cache commands

● To get consistent time measurements without being affected by caches, you'll have to clear all caches before executing a query under test:

1. alter system flush buffer_cache;

● The buffer_cache is part of Oracle's System Global Area and buffers blocks read from disk to minimize disk IO.

2. alter system flush shared_pool;

● The shared_pool is another part of the System Global Area and
is used for caching the library, dictionary and session information.

EXECUTION TIMES AND QUERIES

Try\Query	1	2	3	4 (UPDATE)	5 (UPDATE)	6 (UPDATE)	7 (DELETE)
1	1.879	0.803	0.667	0.697	1.398	0.790	0.118
2	1.823	0.838	0.596	1.024	2.197	1.749	
3	1.820	0.719	0.418	2.347	2.796	0.429	
4	2.123	0.759	0.426	1.038	1.205	0.465	
5	1.856	0.993	0.417	0.860	1.635	0.515	
6	1.892	0.693	0.417	1.399	1.521	0.532	
7	1.884	0.734	0.412	1.987	1.459	0.589	
8	1.853	0.791	0.416	0.966	1.210	0.878	
9	1.849	0.786	0.417	0.534	2.240	0.298	
10	1.830	1.085	0.419	0.833	1.905	0.289	
MAX	2.123	1.085	0.667	1.987	2.796	1.749	
MIN	1.820	0.693	0.412	0.534	1.210	0.289	
Avg	1.880	0.820	0.460	1.168	1.756	0.653	

1ST 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 > 990000 AND model_year > 2012
GROUP BY product_id,brand_name ,product_name, model_year, list_price, category_name
ORDER BY product_id ASC;
```

2nd 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
WHERE staffs.salary > staffs.avgsalary
GROUP BY store_name, first_name, salary, city, state, last_name, active
having (avg(staffs.salary) > 10000 AND state IS NOT NULL) OR (city = 'Aberdeen' AND active = 1)
ORDER BY store_name asc;
```

3rd 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)
WHERE (order_date = '04/26/2017' AND required_date = '04/29/2000')
OR (order_date = '04/26/2017' AND required_date = '04/29/2017')
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
Order by order_status;
```

“ The purpose of this query is to calculate total cost of same products as joining each tables with the criteria of order and required date. “

4th Query (UPDATE)

```
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
              inner join MUTABAY.products products on
                 (stocks.product_id = products.product_id)
                    where ((model_year between 2000 and 1958) or (list_price < 890.00))
                        or product_name LIKE 'C%'
                    fetch first 1 rows only
              )
        fetch first 1 rows only
                        );
```

“ The purpose of this query is to update model year of the product. To do this we joined some tables each other and search store_id to go through the product_id with some criteria. “

5th Query (UPDATE)

```
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 (to_date(shipped_date, 'MM-DD-YYYY') - to_date(required_date , 'MM-DD-YYYY') = 1 ))
          OR
          (order_status = 2 AND (to_date(shipped_date, 'MM-DD-YYYY') - to_date(required_date , 'MM-DD-YYYY') = 0 ))
        )

      )fetch next 1 rows only
);
```

“ The purpose of this statement is to update quantity_id from order_items table. To do this we did some complex things to reach quantity_id as joining lots of tables each other and then put some criteria while searching values. “

6th Query (UPDATE)

```
UPDATE MUTABAY.stores
SET MUTABAY.stores.store_name = (
   SELECT store_name FROM MUTABAY.stores stores
   FULL OUTER 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
         (order_status = 3 AND (to_date(shipped_date, 'MM-DD-YYYY') - to_date(required_date , 'MM-DD-YYYY') = 1 ))
         OR
         (order_status = 2 AND (to_date(shipped_date, 'MM-DD-YYYY') - to_date(required_date , 'MM-DD-YYYY') = 0 ))

      ) order_i
      ON order_i.staff_id = staffs.staff_id
      WHERE (discount > 0.48 AND discount < 0.5) OR salary< 5000
      OR
      (active = 1 AND discount = 0.4) OR quantity_id=1
   )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
);
```

7th Query (DELETE)

```
DELETE FROM MUTABAY.order_items order_items
WHERE order_items.order_id = (
    SELECT order_id from MUTABAY.orders orders
      full outer join MUTABAY.customers customers on
        customers.customer_id = orders.customer_id
      where customers.street LIKE 'A%'
      FETCH FIRST 1 ROWS ONLY
      );
```

“The purpose of this query is to delete order_id from order_items table as using some subqueries and join command to search something on another table”