João Paulo Oliveira Cruz

Database Systems Management

Practice Exercise 03

1

a

I believe the most efficient query is the second one. It uses a *WHERE* clause, which only selects the needed rows.

b

The order of execution is not necessairily the order you write your query in. The order of execution is:

1. Choose rows based on **WHERE** clause
2. Group those rows based on **GROUP BY** clause
3. Calculate the results of the group functions for each group
4. Choose and eliminate groups based on the **HAVING** clause
5. Order the groups based on the results of the group functions in the **ORDER BY** clause.

c

I believe both queries will look through the same amount of rows. But the query below will be faster because the table probably has less rows with a date of birth above the specified one, in comparison to dates of birth that are different from the specified one. (unless that’s the birthday of the oldest person on the table. In that case, both queries should be equal.

2

The optimized queries are as follows:

```
SELECT name, address, custid, dob, sex, phone, email, status
FROM cusm c
WHERE custid EXISTS (select custid from actm);
```

```
SELECT actm.custid, actm.accountbalance, actm.status, cusm.dob, cusm.custid
FROM actm
INNER JOIN cusm ON actm.custid = cusm.custid
WHERE actm.status = 'Closed' AND ROWNUM = 1
ORDER BY accountbalance DESC,
         dob DESC;
```

```
SELECT c.custid, c.cname
FROM cusm c
WHERE EXISTS(SELECT 'X' FROM actm a WHERE a.custid = c.custid);
```

```
SELECT custid, cname, dob
FROM cusm
UNION ALL
SELECT custid, actid
FROM actm;
```

References:

https://logicalread.com/2015/06/03/order-query-execution-oracle-12c-mc06/#.XnzNbVHQi00