## DBMS LAB ASSIGNMENT - 5

NAME: Chamakuri Shivathmika

**REG NO: 19BCS029** 

Q1) Illustrate logical ANY, ALL and LIKE operator- the queries should be relevant to your

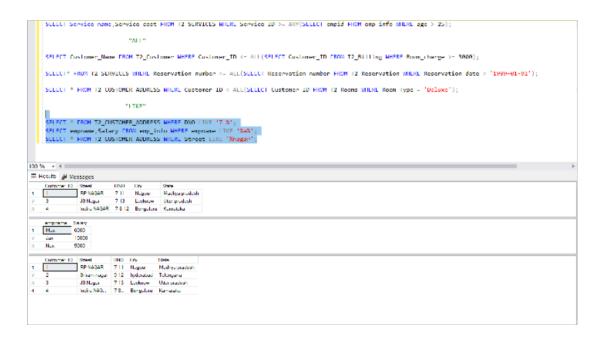
respective databases 3 queries for each operator. One query explaining the difference between ANY and ALL

## QUERIES FOR "ANY"

QUERIES FOR "All"



## QUERIES FOR "Like"



Q2) One query for each Aggregate function.

The aggregate functions are MIN(), MAX(), COUNT(), AVG(), SUM()

AVG() - return the average of the set

MIN() - returns the minimum value in a set

MAX() - returns the maximum value in set

SUM() – returns the sum of all distinct values of a set

COUNT() - returns the number of items in a set

Q3) Illustrate the usage of order by, group by and having clause (2 queries for each case)

### **ORDER BY**



#### **GROUP BY**

```
SILICT * FRON T2_Customer ORDER DY Customer_Name ASC;
SILICT * FRON T2_Customer ORDER DY age DESC;

SILICT * FRON Emp_info ORDER DY age DESC;

SILICT Zipcode, COUNT(*) FROM T2_Customer GROUP DY Zipcode;

SILICT Zipcode, COUNT(*) FROM T2_Customer GROUP BY Room_Type HAVING COUNT(Room_number) >= 1;

SILICT Zipcode, COUNT(*) FROM T2_Customer GROUP BY Room_Type HAVING COUNT(Room_number) >= 1;

SILICT Zipcode, COUNT(*) FROM T2_Customer GROUP BY Room_Type HAVING COUNT(Room_number) >= 1;

SILICT Zipcode (Reservation_number), EFFI(Reservation_date,4) FROM T2_Reservation GROUP BY LEFFI(Reservation_date,4) HAVING COUNT(Reservation_number) >= 1;

SILICT * FRON T2_Customer GROUP BY Room_Type HAVING COUNT(Room_number) >= 1;

SILICT * FRON T2_Customer GROUP BY Room_Type HAVING COUNT(Room_number) >= 1;

SILICT * FRON T2_Customer GROUP BY Room_Type HAVING COUNT(Room_number) >= 1;

SILICT * FRON T2_Customer GROUP BY Room_Type HAVING COUNT(Room_number) >= 1;

SILICT * FRON T2_Customer GROUP BY Room_Type HAVING COUNT(Room_number) >= 1;

SILICT * FRON T2_Customer GROUP BY Room_Type HAVING COUNT(Room_number) >= 1;

SILICT * FRON T2_Customer GROUP BY ROOM_TYPE HAVING COUNT(Room_number) >= 1;

SILICT * FRON T2_Customer GROUP BY ROOM_TYPE HAVING COUNT(Room_number) >= 1;

SILICT * SILICT * FRON T2_Customer GROUP BY ROOM_TYPE HAVING COUNT(Room_number) >= 1;

SILICT * SILICT * FRON T2_Customer GROUP BY ROOM_TYPE HAVING COUNT(Room_number) >= 1;

SILICT * CUSTOMER BY ROOM_TYPE HAVING COUNT(Room_number) >= 1;

SILICT * FRON T2_Customer GROUP BY ROOM_TYPE HAVING COUNT(Room_number) >= 1;

SILICT * CUSTOMER BY ROOM_TYPE HAVING COUNT(Room_number) >= 1;

SILICT * CUSTOMER BY ROOM_TYPE HAVING COUNT(Room_number) >= 1;

SILICT * CUSTOMER BY ROOM_TYPE HAVING COUNT(Room_number) >= 1;

SILICT * CUSTOMER BY ROOM_TYPE HAVING COUNT(Room_number) >= 1;

SILICT * CUSTOMER BY ROOM_TYPE HAVING COUNT(Room_number) >= 1;

SILICT * CUSTOMER BY ROOM_TYPE HAVING COUNT(Room_number) >= 1;

SILICT * CUSTOMER BY ROOM_TYPE HAVING COUNT(Room_number) >= 1;

SILICT * CUSTOM
```

### HAVING CLAUSE

```
SELECT * FROM T2_customer OMDER BY Customer_Name ASC;

SELECT * FROM remp_info ORDER BY age DESC;

SELECT * FROM remp_info ORDER BY age DESC;

SELECT number_of_bods, COUNT(*) $5 number_of_rooms FROM T2_Rooms GROUP DY number_of_bods;

SELECT Zipcode; COUNT(*) $600 T2_Customer GROUP BY Zipcode;

**HAVING**

**HAVING**

**HAVING**

**HAVING**

**HAVING**

**HAVING**

**HAVING**

**HAVING**

**LECT COUNT(Reservation_number), LEFT(Reservation_date,4) $FROM T2_Reservation GROUP BY LEFT(Reservation_date,4) **HAVING**

**DOTAL **

**Reservation_number**

**PROM_T2_Reservation_date,4) $FROM T2_Reservation GROUP BY LEFT(Reservation_date,4) **HAVING**

**PROM_T2_Reservation_date,4) **HAVING**

**PROM_T2_
```

# AVG():

## COUNT():

```
EUSE HOTEL;

SELECT AWG(number_of_beds) FROM T2_Rooms GROUP BY Room_location HAVING Room_location LIKE 'block$';

SELECT COUNT(Customer_ID) FROM T2_Reservation GROUP BY Check_in_date HAVING Check_in_date >= '1992-02-03']

SELECT MIN(Salary) FROM emp_info GROUP BY age HAVING age >= 25;

SELECT MIN(Room_charge) FROM T2_Billing GROUP BY LEFT(Payment_date,7) HAVING LEFT(Payment_date,7) LIKE '2021-$';

SELECT SUM(Service_cost) FROM T2_SERVICES GROUP BY Service_cost HAVING Service_cost BETHEEN 4000 AND 6000;

B Results | M Messages | (No outurn number) | 1 | 2 | 2 | 1
```

# MIN():

# MAX():

# SUM():

```
BUSE HOTEL;

SELECT AVG(number_of_beds) FROM T2_Rooms GROUP BY Room_location HAVING Room_location LIKE 'block%';

SELECT COUNT(Customer_ID) FROM T2_Reservation GROUP BY Check_in_date HAVING Check_in_date >= '1992-02-03';

SELECT MUN(Salary) FROM onp_info GROUP BY age HAVING age > 25;

SHIECT MUN(Salary) FROM onp_info GROUP BY LEFT(Payment_date,7) HAVING LEFT(Payment_date,7) LIKE '2021-%';

SHIECT MUN(Service_cost) FROM T2_SERVICES GROUP BY Service_cost HAVING Se
```

Q5) Write at least 3 nested queries using order by, group by and having clause.

## QUERY:

```
ESCLECT Customer_Name_COUNT(*) FROM T2_Customer

NMERE Customer_ID = ANY(

SELECT Reservation

MERE Reservation_number = ANY(

SELECT Reservation_number FROM T2_SERVICES

Metre Service_cost >= 4000

)

GROUP BY Customer_Name HAVING Customer_Name LIKE "Max"

ONDER BY Customer_Name desc;

Results g Messages

Customer_Name (No courne name)

1 Findin 1 1
```

# Q6) Illustrate the Usage of Except, Exists, Not Exists, Union, Intersection EXCEPT():

```
ESELECT Customer_ID FROM T2_Roservation

EXCEPT
SELECT Customer_ID FROM T2_Rosers

WHIRE CUSTS

(SELECT Customer_ID FROM T2_Roser)

WHIRE CUSTS

(SELECT Customer_ID FROM T2_Roservation);

SELECT Customer_ID FROM T2_Roservation);

SELECT Customer_ID FROM T2_Roservation);

SELECT Customer_ID FROM T2_Customer

WHIRE NOT EXISTS

(SELECT Customer_ID FROM T2_Customer)

SELECT City FROM T2_Customer_ID FROM T2_Roservation);

SELECT City FROM T2_Customer_ID FROM T2_Roservation);

SELECT City FROM T2_Customer_ID FROM T2_SELECT ROservation_ID FROM T2_
```

## EXISTS():

## NOT EXISTS():

# UNION():

```
-----EXCEPT-----
   □SELECT Customer_ID FROM TZ_Customer
   EXCEPT
    SELECT Customer_ID FROM T2_Reservation;
    -----EXISTS-----
   BSELECT Customer_ID FROM T2_Rooms
   WHERE EXISTS
    (SELECT Customer_ID FROM T2_Billing)
ORDER BY Customer_ID ASC;
   HSELECT * FROM T2_Customer
WHERE NOT EXISTS
    (SELECT Customer_ID FROM T2_Reservation);
    SELECT City FROM T2_CUSTOMER_ADDRESS
    UNION
SELECT City FROM TZ_Customer;
   SELECT Service_cost FROM T2_SERVICES;
00% + <
⊞ Results 🖫 Messages
City

Bengaluru

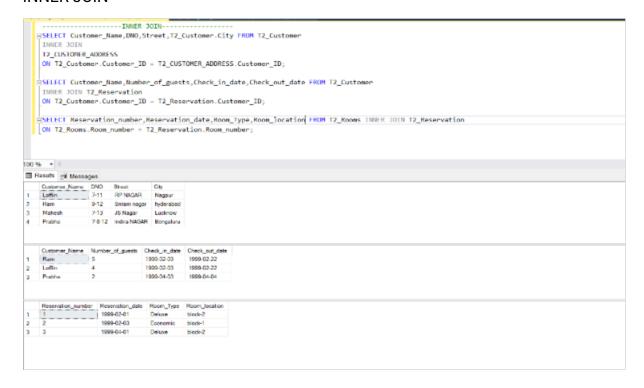
Inyderabad

Lucknow
```

### **INTERSECT:**

```
-----EXCEPT-----
  ☐SELECT Customer_ID FROM T2_Customer
   EXCEPT
   SELECT Customer_ID FROM T2_Reservation;
    ----EXISTS-----
   SELECT Customer_ID FROM T2_Rooms
   WHERE EXISTS
    (SELECT Customer_ID FROM T2_Billing)
   ORDER BY Customer_ID ASC;
   SELECT * FROM T2_Customer
WHERE NOT EXISTS
    (SELECT Customer_ID FROM T2_Reservation);
     ------UNION-
   SELECT City FROM T2_CUSTOMER_ADDRESS
    SELECT City FROM T2_Customer;
   INTERSECT
SELECT Service_cost FROM T2_SERVICES;
100 % *
⊞ Results ≝W Messages
Room charge
1 3000
```

**Q7**) INNER JOIN, LEFT OUTER JOIN, RIGHT OUTER JOIN- 3 queries for each instance INNER JOIN



### **LEFT OUTER JOIN**

```
SELECT * FROM T2_Customer
LEFT OUTER JOIN T2_Rooms
       ON T2_Customer.Customer_ID = T2_Rooms.Customer_ID;
       SELECT * FROM T2_CUSTOMER_ADDRESS
       LEFT OUTER JOIN T2_Reservation
ON T2_Reservation.Customer_ID = T2_CUSTOMER_ADDRESS.Customer_ID;
       SELECT * FROM emp_info LEFT OUTER JOIN
       T2_SERVICES ON
      T2_SERVICES.Service_ID = emp_info.empid;
-----RIGHT_OUTER_JOIN-----
     SELECT * FROM T2_Rooms
RIGHT OUTER JOIN T2 Customer
■ Results 🖫 Messages

        Customer_ID
        Customer_Name
        Phone_number
        City

        1
        Loffin
        8688543748
        Nagpur

        2
        Ram
        9588543744
        hyderahad

                                                           Nagpur MP
hyderabad TN
                      Mahash
                                          8688543746
                                                           Lucknow UP
                                                                                                                                                   NULL NULL NULL
                                  8688543766
                                                           Bengaluru Karnataka 534201 prab@gmail.com 3
                                                                                                                                                   block-2
                     Prabha
                                                                                                                                    Deluxe

        Street
        DNO

        RP NAGAR
        7-11

        Sriram nagar
        9-12

        JS Nagar
        7-13

                                                            State Reservation_number
Madhya pradash 2
     Customer_ID Street
1 RP NAGAR
                                               City
Nagpur
                                               hyderabad Telengana
Lucknow Uter pradesh
                                                                                                      1999-02-03
                                                                                                                       1999-02-22
                                                                                                                                                                1999-02-01
                                                                                                                                   2 5 1999-0
NULL NULL
                                                                                                                                                                           NULL NULL
                                                                                                      NULL
                                                                                                                       NULL
                      Indira NAGAR 7-8-12 Bengaluru Kamataka
                                                                                                      1999-04-03
                                                                                                                      1999-04-04
                                                                                                                                                                1999-04-01
                                                Salary Service_ID Service
                                                                               nai P. S. C. Cook

        empid
        empname
        dob

        1
        Max
        1999

        2
        Jax
        1966

                                                                                                       Reservation_number
                             dob age
1999-03-21 22
                            1959-04-05 62 10000 2
                            1992-07-07 29 5000 3 Hoom
2001-12-11 29 6000 NULL NULL
1976-05-30 45 10000 NULL NULL
               Nax
                                                                                                       NULL
                                                                                 NULL
               Chuck
                                                                                                      NULL
```

#### **RIGHT OUTER JOIN**

