Task No: 05 Writing Join Queries, Equivalent, AND 10R Date: 09/09/25 Recursive Queries

Alm: - To implement and execute join aprelies, equivalent quesies and recursive quesies

Types of joins in SQL:-

1. Inner Join: - Retains records that have matching values in both tables.

Syntax: - Select column. name (5) from table I INNER JOIN table 2 on table 1. column-name;

2. Left Order Join: - Returns all records from the Left table, and the matched records from the right table.

Syntau :- Select column - names from table 1 LEFT JOIN Table? on table 1. column - vame = table 2. Column-name.

3. Right Order Join: - Return all records from the right table, and the matched records from the left table.

Syntax: - Select Column-numers) from table 1 EFT RIGHT JOIN table 2 on table 1. column-name = table 2. column-name.

in either left of right table.

syntax: - Select Column-names) from table 1 FUII acted Toin table 2 ON table 1 · Column-name = table 2 · Column-name;

## 1. Join Queries

Create Tables

create table customer (

CustomerID int primary key, norme valetion (60),

address varchar (100), referred by ID INT NULL,
): Foreign key (referred by ID References. (customer (customer ID)

create table bank-account (

account number int primary try; customer ID int; balance int; category varchar (50).

```
foreign key (customer ID) references customer (customer)
       );
 create table branch (
       branch ID int Primary key,
        branch Name Valchar (50),
       7:
2. Insert Sample data
     in seet into customer (customerID, home, address) values (101, Pam
     Kumos! (chennai);
    insert into customer (customer ID, name, address) values (102, Vijay
      Roo', 'Hyderabad');
    insert into customer (customer DI name, address) values (103, Vasu
     reddy', 'vizag');
    insert into customer (customer ID, name, orderers) Values (104, Robit!
      Chennail.
     insert into customer (customer ID, name, address) value (105, 'Vinay
     Kumosi, 'Odhi');
   insert into bank-account (account-number, customer_ID, balance,
    Category) Values (1001,101,15000, 'Savings');
   insert into bank-account (account-number, Customer-ID balance,
   Category) values (1002, 102, 0, 'current');
   insert into bank-account Caccount-number, customerID, balance,
   Category) values (1003,103, 5000, 'Savings');
   insert into bounk-account Caccount-number, castomerID, balance, Categor
   Values (1004/105, 2000, 1 current 1);
  insect into branch (branch ID , brach Name) values (1, 'chennai Branch'
  insert into branch Cbranch to , branch Name) values (2,144 desabad Bran
  insert into branch (branch ID, branch Name) Values (3, Vizag Branch'
5. Join Quelies: -
a) Inner Join:
Quely: - select coname, braccount, number for customer c in
```

Join bank-account & ONC. Customer. ID = b. customer ID;

## Output:

account-number
1001
1002
5001
1004

b. left join : -

Quely: select ename, b. account-number from customera left join bank-account bon c.customer ID = b. Customer ID;

name	account-number
Ram kumar	1001
Vijay Rao	1002
Vasu Reddy	€ १ ००३
Vinay kumar	1004
Rohit Shaama	NOLL
	Ram kumor Vijay Roo Vasu Reddy Vinay kumor

3. Right Join :-

Query: select a name, biaccount-number from customer a Right Join bank-account 6 ON C. customer ID = b. Customer ID; Output !-

name	account number
Pam Kumar	1001
Vijay Rao	1002
Vasu Redoly	5001
vincy kumon	1004

d. Full Order Join:

Query: Select coname, braccount-number from customer c full order join bank-account 6 ON C. customerID= 6 customerID.

Name	account number
Romkumon	1001
Vijay Rao	1002
Vasu Reddy	5'001
Vinoy kumon	1004
Robit Sharma	NULL

## Equivalent query

a) Using Join

Account number from customer C Join bank-account bonc. customerID.

= b. customer ID:

Output	,
0 0 1 000	

customer Hame	Account Number
Ram kamon	Lool
Vijay Rao	1002
Vasu reddy	1003
Vinay kuman	1004

## b. using sub away

Select coname as customername, (select braccount number From bank-account bankaccount brance brows to be customer ID -cocustomer ID limit 1) As account Number from customer c;

Output

1	Custome? Name	Account Number
Common	Ram kuman	1001
ONE OF THE LAND	Vijoy Rao	1002
	vasa reddy	1003
Constitution, V	Vinay kumar	1004
	Rohit Shoama	NULL

7

5. Recuasive Query:

Query: with Recursive Referred Hierorchy AS (select customer ID, referred ByID from customer where befored ByID is NOT phull UNION select a customerID, a referred ByID From Customera)

Select \* from Referred Hierarachy;

Output: -

Customer ID	referred ID
102	tor
103	102
104	103
/ /	

EX NO

PERFORMANCE (5)

RESULT AND ANALYSIS (5)

VIVA VICE (5)

RECO (5)

Result:

The implementation of SQL Commonds using Joins and

de cursive Queries are enecuted successfully.