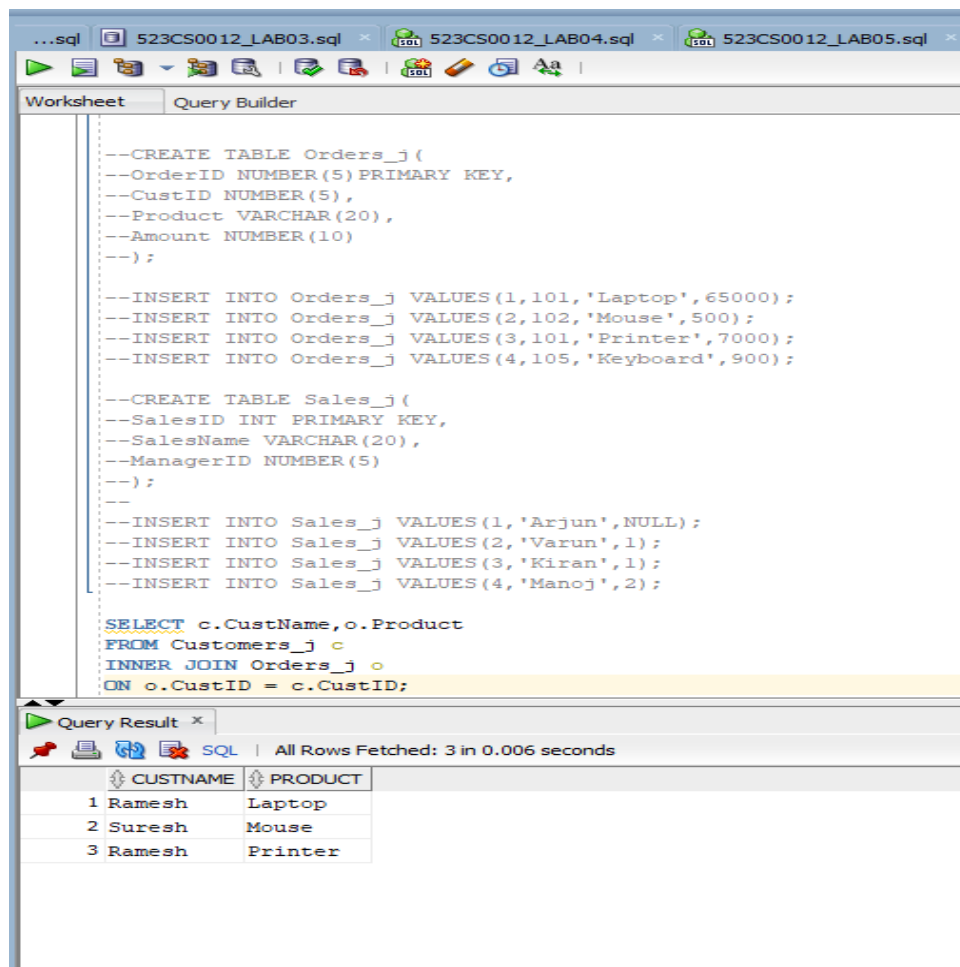


# JOINS AND VIEWS

523cs0012(Gaurav Singh)

Assignment – 1.....

PART A -- (JOINS)



The screenshot shows a SQL IDE window with three tabs: '523CS0012\_LAB03.sql', '523CS0012\_LAB04.sql', and '523CS0012\_LAB05.sql'. The 'Query Builder' tab is active, displaying a SQL script. The script creates two tables, 'Orders\_j' and 'Sales\_j', and inserts data into them. It then executes a query that performs an inner join between 'Customers\_j' and 'Orders\_j' on the 'CustID' column. The query results are displayed in a table below the script.

```
--CREATE TABLE Orders_j(  
--OrderID NUMBER(5) PRIMARY KEY,  
--CustID NUMBER(5),  
--Product VARCHAR(20),  
--Amount NUMBER(10)  
--);  
  
--INSERT INTO Orders_j VALUES(1,101,'Laptop',65000);  
--INSERT INTO Orders_j VALUES(2,102,'Mouse',500);  
--INSERT INTO Orders_j VALUES(3,101,'Printer',7000);  
--INSERT INTO Orders_j VALUES(4,105,'Keyboard',900);  
  
--CREATE TABLE Sales_j(  
--SalesID INT PRIMARY KEY,  
--SalesName VARCHAR(20),  
--ManagerID NUMBER(5)  
--);  
  
--INSERT INTO Sales_j VALUES(1,'Arjun',NULL);  
--INSERT INTO Sales_j VALUES(2,'Varun',1);  
--INSERT INTO Sales_j VALUES(3,'Kiran',1);  
--INSERT INTO Sales_j VALUES(4,'Manoj',2);  
  
SELECT c.CustName,o.Product  
FROM Customers_j c  
INNER JOIN Orders_j o  
ON o.CustID = c.CustID;
```

Query Result x

All Rows Fetched: 3 in 0.006 seconds

	CUSTNAME	PRODUCT
1	Ramesh	Laptop
2	Suresh	Mouse
3	Ramesh	Printer

...sql
523CS0012\_LAB03.sql
523CS0012\_LAB04.sql
523CS0012\_LAB05.sql

Worksheet
Query Builder

```

--Amount NUMBER(10)
--);

--INSERT INTO Orders_j VALUES(1,101,'Laptop',65000);
--INSERT INTO Orders_j VALUES(2,102,'Mouse',500);
--INSERT INTO Orders_j VALUES(3,101,'Printer',7000);
--INSERT INTO Orders_j VALUES(4,105,'Keyboard',900);

--CREATE TABLE Sales_j(
--SalesID INT PRIMARY KEY,
--SalesName VARCHAR(20),
--ManagerID NUMBER(5)
--);

--
--INSERT INTO Sales_j VALUES(1,'Arjun',NULL);
--INSERT INTO Sales_j VALUES(2,'Varun',1);
--INSERT INTO Sales_j VALUES(3,'Kiran',1);
--INSERT INTO Sales_j VALUES(4,'Manoj',2);

--SELECT c.CustName,o.Product
--FROM Customers_j c
--INNER JOIN Orders_j o
--ON o.CustID = c.CustID;

SELECT c.CustName,o.OrderID,o.Product,o.amount
FROM Customers_j c
LEFT OUTER JOIN Orders_j o
ON o.CustID = c.CustID;

```

Query Result x

SQL | All Rows Fetched: 5 in 0.002 seconds

	CUSTNAME	ORDERID	PRODUCT	AMOUNT
1	Ramesh	1	Laptop	65000
2	Suresh	2	Mouse	500
3	Ramesh	3	Printer	7000
4	Naresh	(null)	(null)	(null)
5	Mahesh	(null)	(null)	(null)

...sql 523CS0012\_LAB03.sql x 523CS0012\_LAB04.sql x 523CS0012\_LAB05.sql x

Worksheet Query Builder

```
--INSERT INTO Orders_j VALUES(2,102,'Mouse',900);
--INSERT INTO Orders_j VALUES(3,101,'Printer',7000);
--INSERT INTO Orders_j VALUES(4,105,'Keyboard',900);

--CREATE TABLE Sales_j(
--SalesID INT PRIMARY KEY,
--SalesName VARCHAR(20),
--ManagerID NUMBER(5)
--);
--
--INSERT INTO Sales_j VALUES(1,'Arjun',NULL);
--INSERT INTO Sales_j VALUES(2,'Varun',1);
--INSERT INTO Sales_j VALUES(3,'Kiran',1);
--INSERT INTO Sales_j VALUES(4,'Manoj',2);

--SELECT c.CustName,o.Product
--FROM Customers_j c
--INNER JOIN Orders_j o
--ON o.CustID = c.CustID;

--SELECT c.CustName,o.OrderID,o.Product,o.amount
--FROM Customers_j c
--LEFT OUTER JOIN Orders_j o
--ON o.CustID = c.CustID;

SELECT c.CustName,o.Product
FROM Customers_j c
RIGHT OUTER JOIN Orders_j o
ON o.CustID = c.CustID;
```

Query Result x

SQL | All Rows Fetched: 4 in 0.003 seconds

	CUSTNAME	PRODUCT
1	Ramesh	Laptop
2	Ramesh	Printer
3	Suresh	Mouse
4	(null)	Keyboard

...sql
523CS0012\_LAB03.sql
523CS0012\_LAB04.sql
523CS0012\_LAB05.sql

Worksheet
Query Builder

```

--SalesName VARCHAR(20),
--ManagerID NUMBER(5)
--);
--
--INSERT INTO Sales_j VALUES(1,'Arjun',NULL);
--INSERT INTO Sales_j VALUES(2,'Varun',1);
--INSERT INTO Sales_j VALUES(3,'Kiran',1);
--INSERT INTO Sales_j VALUES(4,'Manoj',2);

--SELECT c.CustName,o.Product
--FROM Customers_j c
--INNER JOIN Orders_j o
--ON o.CustID = c.CustID;

--SELECT c.CustName,o.OrderID,o.Product,o.amount
--FROM Customers_j c
--LEFT OUTER JOIN Orders_j o
--ON o.CustID = c.CustID;

--SELECT c.CustName,o.Product
--FROM Customers_j c
--RIGHT OUTER JOIN Orders_j o
--ON o.CustID = c.CustID;

SELECT c.CustName,c.City,o.Product,o.amount
FROM Customers_j c
FULL OUTER JOIN Orders_j o
ON o.CustID = c.CustID;

```

Query Result x

SQL | All Rows Fetched: 6 in 0.003 seconds

	CUSTNAME	CITY	PRODUCT	AMOUNT
1	Ramesh	Hyderabad	Laptop	65000
2	Suresh	Chennai	Mouse	500
3	Ramesh	Hyderabad	Printer	7000
4	(null)	(null)	Keyboard	900
5	Naresh	Delhi	(null)	(null)
6	Mahesh	Mumbai	(null)	(null)

...sql 523CS0012\_LAB03.sql x 523CS0012\_LAB04.sql x 523CS0012\_LAB05.sql x

0.104 seconds

Worksheet Query Builder

```
--ON o.CustID = c.CustID;

--SELECT c.CustName,o.OrderID,o.Product,o.amount
--FROM Customers_j c
--LEFT OUTER JOIN Orders_j o
--ON o.CustID = c.CustID;

--SELECT c.CustName,o.Product
--FROM Customers_j c
--RIGHT OUTER JOIN Orders_j o
--ON o.CustID = c.CustID;

--SELECT c.CustName,c.City,o.Product,o.amount
--FROM Customers_j c
--FULL OUTER JOIN Orders_j o
--ON o.CustID = c.CustID;






























SELECT c.CustName,o.Product
FROM Customers_j c
CROSS JOIN Orders_j o;
```

Query Result x Script Output x

SQL | All Rows Fetched: 16 in 0.004 seconds

	CUSTNAME	PRODUCT
1	Ramesh	Laptop
2	Suresh	Laptop
3	Mahesh	Laptop
4	Naresh	Laptop
5	Ramesh	Mouse
6	Suresh	Mouse
7	Mahesh	Mouse
8	Naresh	Mouse
9	Ramesh	Printer
10	Suresh	Printer
11	Mahesh	Printer
12	Naresh	Printer
13	Ramesh	Keyboard
14	Suresh	Keyboard
15	Mahesh	Keyboard
16	Naresh	Keyboard

...sql523CS0012\_LAB03.sql523CS0012\_LAB04.sql523CS0012\_LAB05.sql



WorksheetQuery Builder

```
--SELECT c.CustName,o.OrderID,o.Product,o.amount
--FROM Customers_j c
--LEFT OUTER JOIN Orders_j o
--ON o.CustID = c.CustID;






























--SELECT c.CustName,o.Product
--FROM Customers_j c
--RIGHT OUTER JOIN Orders_j o
--ON o.CustID = c.CustID;

--SELECT c.CustName,c.City,o.Product,o.amount
--FROM Customers_j c
--FULL OUTER JOIN Orders_j o
--ON o.CustID = c.CustID;

--SELECT c.CustName,o.Product
--FROM Customers_j c
--CROSS JOIN Orders_j o;

SELECT a.SalesName AS Employee, b.SalesName AS Manager
FROM Sales_j a , Sales_j b
WHERE a.ManagerID = b.SalesID;
```

Script Output xQuery Result x



SQL | All Rows Fetched: 3 in 0.012 seconds

	EMPLOYEE	MANAGER
1	Varun	Arjun
2	Kiran	Arjun
3	Manoj	Varun

## PART B – (VIEWS)

The screenshot shows the SQL Developer interface with the Query Builder tab active. The query editor contains several SQL statements, with the last one highlighted in yellow:

```
--FROM Customers_j c
--RIGHT OUTER JOIN Orders_j o
--ON o.CustID = c.CustID;

--SELECT c.CustName,c.City,o.Product,o.amount
--FROM Customers_j c
--FULL OUTER JOIN Orders_j o
--ON o.CustID = c.CustID;

--SELECT c.CustName,o.Product
--FROM Customers_j c
--CROSS JOIN Orders_j o;

--SELECT a.SalesName AS Employee, b.SalesName AS Manager
--FROM Sales_j a , Sales_j b
--WHERE a.ManagerID = b.SalesID;

CREATE OR REPLACE VIEW HYD_CUST_VIEW AS
SELECT CustName, City
FROM Customers_j
WHERE City = 'Hyderabad';
```

The Query Result pane shows the output of the query, which is a list of 16 rows. The first 10 rows are a list of customer names and their products. The next 5 rows are a list of customer names and their products, with the first row being a header row. The last row is a message indicating that 16 rows were selected.

Mahesh	Laptop
Naresh	Laptop
Ramesh	Mouse
Suresh	Mouse
Mahesh	Mouse
Naresh	Mouse
Ramesh	Printer
Suresh	Printer
Mahesh	Printer
CUSTNAME	PRODUCT
Naresh	Printer
Ramesh	Keyboard
Suresh	Keyboard
Mahesh	Keyboard
Naresh	Keyboard

16 rows selected.

View HYD\_CUST\_VIEW created.

...sql | 523CS0012\_LAB03.sql | 523CS0012\_LAB04.sql | 523CS0012\_LAB05.sql

Worksheet | Query Builder

```
--ON o.CustID = c.CustID;

--SELECT c.CustName,c.City,o.Product,o.amount
--FROM Customers_j c
--FULL OUTER JOIN Orders_j o
--ON o.CustID = c.CustID;

--SELECT c.CustName,o.Product
--FROM Customers_j c
--CROSS JOIN Orders_j o;

--SELECT a.SalesName AS Employee, b.SalesName AS Manager
--FROM Sales_j a , Sales_j b
--WHERE a.ManagerID = b.SalesID;

--
--CREATE OR REPLACE VIEW HYD_CUST_VIEW AS
--SELECT CustName,City
--FROM Customers_j
--WHERE City = 'Hyderabad';

SELECT * FROM HYD_CUST_VIEW;
```

Script Output x | Query Result x

SQL | All Rows Fetched: 1 in 0.003 seconds

	CUSTNAME	CITY
1	Ramesh	Hyderabad



...sql 523CS0012\_LAB03.sql 523CS0012\_LAB04.sql 523CS0012\_LAB05.sql 523CS0012\_LAB06.sql

Worksheet Query Builder

```
--ON o.CustID = c.CustID;

--SELECT c.CustName,o.Product
--FROM Customers_j c
--CROSS JOIN Orders_j o;

--SELECT a.SalesName AS Employee, b.SalesName AS Manager
--FROM Sales_j a , Sales_j b
--WHERE a.ManagerID = b.SalesID;

--
--CREATE OR REPLACE VIEW HYD_CUST_VIEW AS
--SELECT CustName,City
--FROM Customers_j
--WHERE City = 'Hyderabad';

--UPDATE HYD_CUST_VIEW
--SET City = 'Kurnool'
--WHERE CustName = 'Ramesh';

DROP VIEW HYD_CUST_VIEW;
```

Script Output x Query Result x

Task completed in 0.049 seconds

Error at Command Line : 77 Column : 15  
Error report -  
SQL Error: ORA-00942: table or view does not exist

<https://docs.oracle.com/error-help/db/ora-00942/00942.00000> - "table or view does not exist"

\*Cause: The specified table or view did not exist, or a synonym pointed to a table or view that did not exist. To find existing user tables and views, query the ALL\_TABLES and ALL\_VIEWS data dictionary views. Certain privileges may be required to access the table. If an application returned this message, then the table that the application tried to access did not exist in the database, or the application did not have access to it.










\*Action: Check each of the following

- The spelling of the table or view name is correct.
- The referenced table or view name does exist.
- The synonym points to an existing table or view.

More Details :  
<https://docs.oracle.com/error-help/db/ora-00942/>

View HYD\_CUST\_VIEW dropped.

...sql523CS0012\_LAB03.sql523CS0012\_LAB04.sql523CS0012\_LAB05.sql



WorksheetQuery Builder

```
--WHERE a.ManagerID = b.SalesID;

--
--CREATE OR REPLACE VIEW HYD_CUST_VIEW AS
--SELECT CustName, City
--FROM Customers_j
--WHERE City = 'Hyderabad';










--SELECT * FROM HYD_CUST_VIEW;
--UPDATE HYD_CUST_VIEW
--SET City = 'Kurnool'
--WHERE CustName = 'Ramesh';

--DROP VIEW HYD_CUST_VIEW;

--CREATE OR REPLACE VIEW VIEW_2 AS
--SELECT c.CustName, o.Product, o.amount
--FROM Customers_j c
--INNER JOIN Orders_j o
--ON o.CustID = c.CustID;

SELECT * FROM VIEW_2;
```

Script Output xQuery Result x



SQL | All Rows Fetched: 3 in 0.002 seconds

	CUSTNAME	PRODUCT	AMOUNT
1	Ramesh	Laptop	65000
2	Suresh	Mouse	500
3	Ramesh	Printer	7000

## Assignment 2 .....

### PART A (JOINS)

The screenshot shows the SQL Developer interface with a query script in the main editor and its results in the Query Results pane.

**Query Script:**

```
--INSERT INTO Members_j VALUES ('M3','Rohit','Mumbai');
--INSERT INTO Members_j VALUES ('M4','Neha','Delhi');

--CREATE TABLE Issue_Details_j(
--IssueID VARCHAR2(4),
--MemberID VARCHAR2(2),
--BookID VARCHAR2(4),
--IssueDate DATE,
--ReturnDate DATE
--);

--INSERT INTO Issue_Details_j VALUES('I1','M1','B1',TO_DATE('01-SEP-2024','DD-MON-YYYY'),TO_DATE('01-SEP-2024','DD-MON-YYYY'));
--INSERT INTO Issue_Details_j VALUES('I2','M2','B3',TO_DATE('03-SEP-2024','DD-MON-YYYY'),TO_DATE('03-SEP-2024','DD-MON-YYYY'));
--INSERT INTO Issue_Details_j VALUES('I3','M1','B2',TO_DATE('04-SEP-2024','DD-MON-YYYY'),TO_DATE('04-SEP-2024','DD-MON-YYYY'));
--INSERT INTO Issue_Details_j VALUES('I4','M5','B4',TO_DATE('05-SEP-2024','DD-MON-YYYY'),TO_DATE('05-SEP-2024','DD-MON-YYYY'));

--SELECT m.MemberName,b.Title
--FROM   Members_j m
--INNER JOIN Issue_Details_j i on m.MemberID = i.MemberID
--INNER JOIN Books_j b on i.BookID = b.BookID;
```

**Query Results:**

	MEMBERNAME	TITLE
1	Arjun	Database Systems
2	Arjun	Let Us C
3	Sneha	Python Basics

...sql
523CS0012\_LAB03.sql
523CS0012\_LAB04.sql
523CS0012\_LAB05.sql

Worksheet
Query Builder

```

--IssueID VARCHAR2(4),
--MemberID VARCHAR2(2),
--BookID VARCHAR2(4),
--IssueDate DATE,
--ReturnDate DATE
--);

--INSERT INTO Issue_Details_j VALUES('I1','M1','B1',TO_DATE('01-01-2017'));
--INSERT INTO Issue_Details_j VALUES('I2','M2','B3',TO_DATE('03-01-2017'));
--INSERT INTO Issue_Details_j VALUES('I3','M1','B2',TO_DATE('04-01-2017'));
--INSERT INTO Issue_Details_j VALUES('I4','M5','B4',TO_DATE('05-01-2017'));

--SELECT m.MemberName,b.Title
--FROM Members_j m
--INNER JOIN Issue_Details_j i on m.MemberID = i.MemberID
--INNER JOIN Books_j b on i.BookID = b.BookID;

SELECT m.MemberName,b.Title
FROM Members_j m
LEFT JOIN Issue_Details_j i on m.MemberID = i.MemberID
LEFT JOIN Books_j b on i.BookID = b.BookID;

```














Script Output x
Query Result x

All Rows Fetched: 5 in 0.004 seconds

	MEMBERNAME	TITLE
1	Arjun	Database Systems
2	Arjun	Let Us C
3	Sneha	Python Basics
4	Rohit	(null)
5	Neha	(null)



...sql523CS0012\_LAB03.sql523CS0012\_LAB04.sql523CS0012\_LAB05.sql523CS0012\_LAB06.sql



Worksheet

Query Builder

```
--INSERT INTO Issue_Details_j VALUES('I3','M1','B2',TO_DATE('04-SEP-2024','DD-MON-YYYY'),
--INSERT INTO Issue_Details_j VALUES('I4','M5','B4',TO_DATE('05-SEP-2024','DD-MON-YYYY'),

--SELECT m.MemberName,b.Title
--FROM Members_j m
--INNER JOIN Issue_Details_j i on m.MemberID = i.MemberID
--INNER JOIN Books_j b on i.BookID = b.BookID;














--SELECT m.MemberName,b.Title
--FROM Members_j m
--LEFT JOIN Issue_Details_j i on m.MemberID = i.MemberID
--LEFT JOIN Books_j b on i.BookID = b.BookID;

--SELECT m.MemberName,b.Title
--FROM Members_j m
--RIGHT OUTER JOIN Issue_Details_j i on m.MemberID = i.MemberID
--RIGHT OUTER JOIN Books_j b on i.BookID = b.BookID;

SELECT m.MemberName,b.Title
FROM Members_j m
FULL OUTER JOIN Issue_Details_j i on m.MemberID = i.MemberID
FULL OUTER JOIN Books_j b on i.BookID = b.BookID;
```

Script Output x

Query Result x



SQL | All Rows Fetched: 6 in 0.003 seconds

	MEMBERNAME	TITLE
1	Arjun	Database Systems
2	Sneha	Python Basics
3	Arjun	Let Us C
4	(null)	Data Mining
5	Rohit	(null)
6	Neha	(null)

...sql 523CS0012\_LAB03.sql x 523CS0012\_LAB04.sql x 523CS0012\_LAB05.sql x

Worksheet Query Builder

```
--INNER JOIN Issue_Details_j i on m.MemberID = i.MemberID
--INNER JOIN Books_j b on i.BookID = b.BookID;

--SELECT m.MemberName,b.Title
--FROM Members_j m
--LEFT JOIN Issue_Details_j i on m.MemberID = i.MemberID
--LEFT JOIN Books_j b on i.BookID = b.BookID;

--SELECT m.MemberName,b.Title
--FROM Members_j m
--RIGHT OUTER JOIN Issue_Details_j i on m.MemberID = i.MemberID
--RIGHT OUTER JOIN Books_j b on i.BookID = b.BookID;

--SELECT m.MemberName,b.Title
--FROM Members_j m
--FULL OUTER JOIN Issue_Details_j i on m.MemberID = i.MemberID
--FULL OUTER JOIN Books_j b on i.BookID = b.BookID;

SELECT m.MemberName,b.Title
FROM Members_j m
CROSS JOIN Books_j b;
```

Script Output x Query Result x

SQL | All Rows Fetched: 16 in 0.003 seconds

	MEMBERNAME	TITLE
1	Arjun	Database Systems
2	Arjun	Let Us C
3	Arjun	Python Basics
4	Arjun	Data Mining
5	Sneha	Database Systems
6	Sneha	Let Us C
7	Sneha	Python Basics
8	Sneha	Data Mining
9	Rohit	Database Systems
10	Rohit	Let Us C
11	Rohit	Python Basics
12	Rohit	Data Mining
13	Neha	Database Systems
14	Neha	Let Us C
15	Neha	Python Basics
16	Neha	Data Mining

...sql
523CS0012\_LAB03.sql
523CS0012\_LAB04.sql
523CS0012\_LAB05.sql

Worksheet
Query Builder

```

--FULL OUTER JOIN Issue_Details_j i on m.MemberID = i.MemberID
--FULL OUTER JOIN Books_j b on i.BookID = b.BookID;

--SELECT m.MemberName,b.Title
--FROM Members_j m
--CROSS JOIN Books_j b;
--
--CREATE TABLE Staffs_j(
--StaffID VARCHAR(4) PRIMARY KEY,
--StaffName VARCHAR(20),
--SupervisorID VARCHAR(4)
--);

--INSERT INTO Staffs_j VALUES('S1','Meena',NULL);
--INSERT INTO Staffs_j VALUES('S2','Ramesh','S1');
--INSERT INTO Staffs_j VALUES('S3','Kavya','S1');
--INSERT INTO Staffs_j VALUES('S4','Mohan','S2');

SELECT a.StaffName AS Employee, b.StaffName AS Manager
FROM Staffs_j a , Staffs_j b
WHERE a.SupervisorID = b.staffID;

```

Script Output x
Query Result x

SQL | All Rows Fetched: 3 in 0.004 seconds

	EMPLOYEE	MANAGER
1	Ramesh	Meena
2	Kavya	Meena
3	Mohan	Ramesh



## PART B – (VIEWS)

The screenshot shows a SQL IDE with three tabs: 523CS0012\_LAB03.sql, 523CS0012\_LAB04.sql (active), and 523CS0012\_LAB05.sql. The interface includes a toolbar with icons for running queries, saving, and other database operations. The main window is divided into a 'Worksheet' tab and a 'Query Builder' tab. The 'Worksheet' tab contains the following SQL script:

```
--CREATE TABLE Staffs_j(  
--StaffID VARCHAR(4) PRIMARY KEY,  
--StaffName VARCHAR(20),  
--SupervisorID VARCHAR(4)  
--);  
  
--INSERT INTO Staffs_j VALUES('S1','Meena',NULL);  
--INSERT INTO Staffs_j VALUES('S2','Ramesh','S1');  
--INSERT INTO Staffs_j VALUES('S3','Kavya','S1');  
--INSERT INTO Staffs_j VALUES('S4','Mohan','S2');  
  
--SELECT a.StaffName AS Employee, b.StaffName AS Manager  
--FROM Staffs_j a , Staffs_j b  
--WHERE a.SupervisorID = b.staffID;  
  
--CREATE OR REPLACE VIEW PRICE_VIEW AS  
--SELECT Title,Author  
--FROM Books_j  
--WHERE Price > 500;  
  
SELECT * FROM PRICE_VIEW;
```

The 'Query Result' tab is active, showing the output of the last query. It displays a table with two columns: 'TITLE' and 'AUTHOR'. The table contains three rows of data:

	TITLE	AUTHOR
1	Database Systems	Navathe
2	Python Basics	Reema Thareja
3	Data Mining	Han Kamber

Below the table, it indicates 'All Rows Fetched: 3 in 0.004 seconds'.

...sql
523CS0012\_LAB03.sql
523CS0012\_LAB04.sql
523CS0012\_LAB05.sql

Worksheet
Query Builder

```

--SupervisorID VARCHAR(4)
--);

--INSERT INTO Staffs_j VALUES('S1','Meena',NULL);
--INSERT INTO Staffs_j VALUES('S2','Ramesh','S1');
--INSERT INTO Staffs_j VALUES('S3','Kavya','S1');
--INSERT INTO Staffs_j VALUES('S4','Mohan','S2');

--SELECT a.StaffName AS Employee, b.StaffName AS Manager
--FROM Staffs_j a , Staffs_j b
--WHERE a.SupervisorID = b.staffID;

--CREATE OR REPLACE VIEW PRICE_VIEW AS
--SELECT Title,Author,Price
--FROM Books_j
--WHERE Price > 500;

SELECT * FROM PRICE_VIEW;
--UPDATE PRICE_VIEW
--SET Price = 800
--WHERE Title = 'Python Basics';

```

Script Output
Query Result

All Rows Fetched: 3 in 0.001 seconds

	TITLE	AUTHOR	PRICE
1	Database Systems	Navathe	550
2	Python Basics	Reema Thareja	800
3	Data Mining	Han Kamber	700

...sql 523CS0012\_LAB03.sql x 523CS0012\_LAB04.sql x 523CS0012\_LAB05.sql x

Worksheet Query Builder

```
--INSERT INTO Staffs_j VALUES('S1','Meena',NULL);
--INSERT INTO Staffs_j VALUES('S2','Ramesh','S1');
--INSERT INTO Staffs_j VALUES('S3','Kavya','S1');
--INSERT INTO Staffs_j VALUES('S4','Mohan','S2');

--SELECT a.StaffName AS Employee, b.StaffName AS Manager
--FROM Staffs_j a , Staffs_j b
--WHERE a.SupervisorID = b.staffID;

--CREATE OR REPLACE VIEW PRICE_VIEW AS
--SELECT Title,Author,Price
--FROM Books_j
--WHERE Price > 500;

SELECT * FROM PRICE_VIEW;
--UPDATE PRICE_VIEW
--SET Price = 800
--WHERE Title = 'Python Basics';

--DELETE PRICE_VIEW
--WHERE Price > 600;
```

Script Output x Query Result x

SQL | All Rows Fetched: 1 in 0.001 seconds

	TITLE	AUTHOR	PRICE
1	Database Systems	Navathe	550

File
Edit
View
Navigate
Run
Source
Team
Tools
Window
Help

...sql
523CS0012\_LAB03.sql
523CS0012\_LAB04.sql
523CS0012\_LAB05.sql
523CS0012\_LAB06.sql

WorksheetQuery Builder

```

--INSERT INTO Staffs_j VALUES('S4','Mohan','S2');

--SELECT a.StaffName AS Employee, b.StaffName AS Manager
--FROM Staffs_j a , Staffs_j b
--WHERE a.SupervisorID = b.staffID;

--CREATE OR REPLACE VIEW PRICE_VIEW AS
--SELECT Title,Author,Price
--FROM Books_j
--WHERE Price > 500;

--UPDATE PRICE_VIEW
--SET Price = 800
--WHERE Title = 'Python Basics';

--DELETE PRICE_VIEW
--WHERE Price > 600;

--SELECT * FROM PRICE_VIEW;

DROP VIEW PRICE_VIEW;

```

Script Output xQuery Result x

Task completed in 0.029 seconds

- The synonym points to an existing table or view.

View PRICE\_VIEW dropped.

View PRICE\_VIEW created.

1 row updated.

TITLE	AUTHOR	PRICE
Database Systems	Navathe	550
Python Basics	Reema Thareja	800
Data Mining	Han Kamber	700

2 rows deleted.

View PRICE\_VIEW dropped.

...sql 523CS0012\_LAB03.sql x 523CS0012\_LAB04.sql x 523CS0012\_LAB05.sql x

0.23899999 seconds

Worksheet Query Builder

```
--UPDATE PRICE_VIEW
--SET Price = 800
--WHERE Title = 'Python Basics';

--DELETE PRICE_VIEW
--WHERE Price > 600;

--SELECT * FROM PRICE_VIEW;

--DROP VIEW PRICE_VIEW;

--CREATE OR REPLACE VIEW VIEW_02 AS
--SELECT m.MemberName,b.Title,i.IssueDate
--FROM Members_j m
--INNER JOIN Issue_Details_j i on m.MemberID = i.MemberID
--INNER JOIN Books_j b on i.BookID = b.BookID;

--SELECT * FROM VIEW_02;
```

Script Output x Query Result x

Task completed in 0.239 seconds

Table ISSUE\_DETAILS\_J dropped.

Table ISSUE\_DETAILS\_J created.

1 row inserted.

1 row inserted.

1 row inserted.

1 row inserted.

MEMBERNAME	TITLE
Arjun	Database Systems
Arjun	Let Us C