

NAME : CHETAN PRAKASH

UID : 23BCS13776

SECTION : KRG – 1 B

Question 1: Author-Book Relationship Using Joins and Basic SQL Operations

Answer :

```
CREATE TABLE TBL_AUTHOR (  
    AUTHOR_ID INT PRIMARY KEY,  
    AUTHOR_NAME VARCHAR(10),  
    COUNTRY VARCHAR(10)  
);
```

```
CREATE TABLE TBL_BOOK (  
    BOOK_ID INT PRIMARY KEY,  
    BOOK_TITLE VARCHAR(10),  
    AUHTORID INT,  
    FOREIGN KEY (AUHTORID) REFERENCES  
    TBL_AUTHOR(AUTHOR_ID)  
);
```

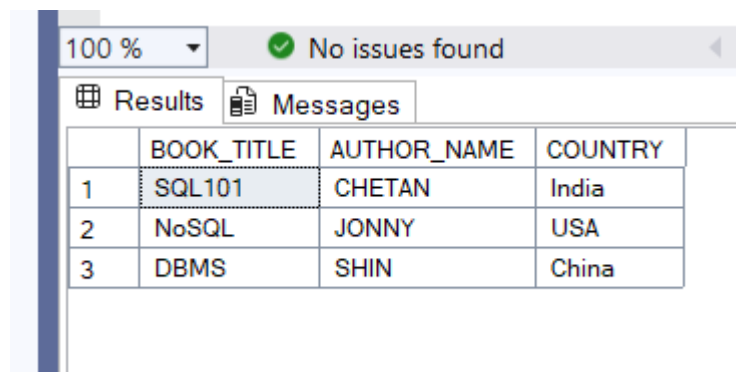
```
INSERT INTO TBL_AUTHOR VALUES  
(1, 'CHETAN', 'India'),  
(2, 'JONNY', 'USA'),  
(3, 'SHIN', 'China');
```

```
INSERT INTO TBL_BOOK VALUES
```

```
(101, 'SQL101', 1),  
(102, 'NoSQL', 2),  
(103, 'DBMS', 3);
```

```
SELECT B.BOOK_TITLE, A.AUTHOR_NAME, A.COUNTRY  
FROM TBL_BOOK AS B  
INNER JOIN TBL_AUTHOR AS A  
ON B.AUHTORID = A.AUTHOR_ID;
```

OUTPUT:



	BOOK_TITLE	AUTHOR_NAME	COUNTRY
1	SQL101	CHETAN	India
2	NoSQL	JONNY	USA
3	DBMS	SHIN	China

Question 2: Department-Course Subquery and Access Control

Answer :

```
CREATE TABLE Department (  
    DeptID INT PRIMARY KEY,  
    DeptName VARCHAR(100)  
);
```

```
CREATE TABLE Course (  
    CourseID INT PRIMARY KEY,  
    CourseName VARCHAR(100),
```

```
DeptID INT,  
FOREIGN KEY (DeptID) REFERENCES Department(DeptID)  
);
```

```
INSERT INTO Department VALUES
```

```
(1, 'Computer Science'),  
(2, 'Physics'),  
(3, 'Mathematics'),  
(4, 'Chemistry'),  
(5, 'Biology');
```

```
INSERT INTO Course VALUES
```

```
(101, 'Data Structures', 1),  
(102, 'Operating Systems', 1),  
(103, 'Quantum Mechanics', 2),  
(104, 'Electromagnetism', 2),  
(105, 'Linear Algebra', 3),  
(106, 'Calculus', 3),  
(107, 'Organic Chemistry', 4),  
(108, 'Physical Chemistry', 4),  
(109, 'Genetics', 5),  
(110, 'Molecular Biology', 5);
```

```
SELECT DeptName  
FROM Department  
WHERE DeptID IN (  
    SELECT DeptID  
    FROM Course  
    GROUP BY DeptID
```

```

HAVING COUNT(*) >= 2

);

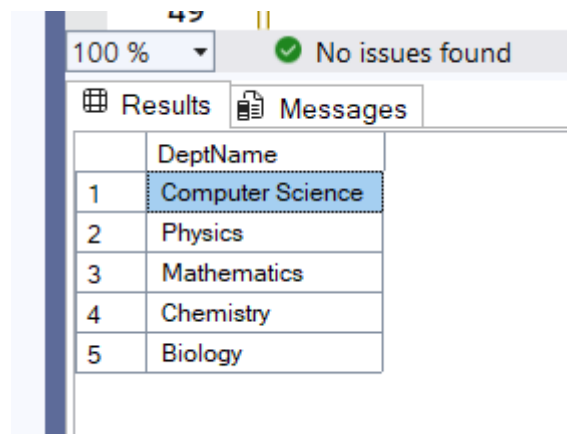
CREATE LOGIN CHETAN
WITH PASSWORD = 'CHETAN@27';

CREATE USER CHETAN_27
FOR LOGIN CHETAN;

GRANT SELECT ON Course TO CHETAN_27;

```

OUTPUT:



The screenshot shows a SQL Server interface with a query window. The top bar indicates '100 %' zoom and 'No issues found'. Below the toolbar, there are tabs for 'Results' and 'Messages'. The 'Results' tab is active, displaying a table with two columns: an implicit index column and 'DeptName'. The table contains five rows of data.

	DeptName
1	Computer Science
2	Physics
3	Mathematics
4	Chemistry
5	Biology

Question 3 : Employee-Manager Reporting Relationship using Self Join

Answer :

```

CREATE TABLE Employee (
    EmpID INT PRIMARY KEY,
    EmpName VARCHAR(50) NOT NULL,
    Department VARCHAR(50) NOT NULL,
    ManagerID INT NULL
);

```

```
ALTER TABLE Employee
```

```
ADD CONSTRAINT FK_Employee FOREIGN KEY (ManagerID)
```

```
REFERENCES Employee(EmpID);
```

```
INSERT INTO Employee (EmpID, EmpName, Department,  
ManagerID) VALUES
```

```
(1, 'Mohan', 'Admin', NULL),
```

```
(2, 'Ankit', 'Sales', 1),
```

```
(3, 'Nisha', 'HR', 2),
```

```
(4, 'Meena', 'Sales', 2),
```

```
(5, 'Rohit', 'Admin', 1),
```

```
(6, 'Komal', 'HR', 3),
```

```
(7, 'Sameer', 'IT', 2);
```

```
SELECT
```

```
E1.EmpName AS [EMPLOYEE NAME],
```

```
E2.EmpName AS [MANAGER NAME],
```

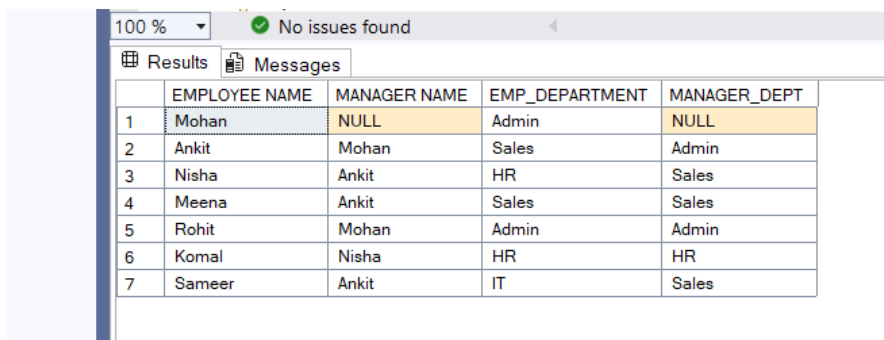
```
E1.Department AS [EMP_DEPARTMENT],
```

```
E2.Department AS [MANAGER_DEPT]
```

```
FROM Employee AS E1
```

```
LEFT OUTER JOIN Employee AS E2
```

```
ON E1.ManagerID = E2.EmpID;
```



	EMPLOYEE NAME	MANAGER NAME	EMP_DEPARTMENT	MANAGER_DEPT
1	Mohan	NULL	Admin	NULL
2	Ankit	Mohan	Sales	Admin
3	Nisha	Ankit	HR	Sales
4	Meena	Ankit	Sales	Sales
5	Rohit	Mohan	Admin	Admin
6	Komal	Nisha	HR	HR
7	Sameer	Ankit	IT	Sales

Question 4 : Yearly NPV Lookup Using LEFT JOIN

ANSWER :

```
CREATE TABLE Queries (  
    ID INT,  
    YEAR INT  
);
```

```
INSERT INTO Queries (ID, YEAR) VALUES  
(1, 2019),  
(13, 2019),  
(15, 2018),  
(15, 2019),  
(6, 2018),  
(3, 2020),  
(7, 2018);
```

```
CREATE TABLE Year_tbl (  
    ID INT,  
    YEAR INT,  
    NPV INT  
);
```

```
INSERT INTO Year_tbl (ID, YEAR, NPV) VALUES  
(1, 2019, 113),  
(3, 2019, 0),  
(3, 2020, 51),  
(7, 2019, 0),  
(7, 2020, 100),
```

(13, 2019, 40),

(15, 2019, 50);

SELECT

Q.ID,

Q.YEAR,

ISNULL(Y.NPV, 0) AS NPV

FROM

Queries AS Q

LEFT OUTER JOIN

Year_tbl AS Y

ON

Q.ID = Y.ID AND Q.YEAR = Y.YEAR;

Results		Messages	
	ID	YEAR	NPV
1	1	2019	113
2	13	2019	40
3	15	2018	0
4	15	2019	50
5	6	2018	0
6	3	2020	51
7	7	2018	0