Create database lms\_db;

USE [lms\_db]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[LMS\_BOOK\_DETAILS](

[BOOK\_CODE] [varchar](10) NOT NULL,

[BOOK\_TITLE] [varchar](50) NOT NULL,

[CATEGORY] [varchar](15) NOT NULL,

[AUTHOR] [varchar](30) NOT NULL,

[PUBLICATION] [varchar](30) NULL,

[PUBLISH\_DATE] [date] NULL,

[BOOK\_EDITION] [int] NULL,

[PRICE] [decimal](8, 2) NULL,

[RACK\_NUM] [varchar](3) NULL,

[DATE\_ARRIVAL] [date] NOT NULL,

[SUPPLIER\_ID] [varchar](3) NULL,

CONSTRAINT [LMS\_cts4] PRIMARY KEY CLUSTERED

(

[BOOK\_CODE] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[LMS\_BOOK\_ISSUE](

[BOOK\_ISSUE\_NO] [int] NOT NULL,

[MEMBER\_ID] [varchar](10) NOT NULL,

[BOOK\_CODE] [varchar](10) NOT NULL,

[DATE\_ISSUE] [date] NOT NULL,

[DATE\_RETURN] [date] NOT NULL,

[DATE\_RETURNED] [date] NULL,

[FINE\_RANGE] [varchar](3) NULL,

CONSTRAINT [LMS\_cts5] PRIMARY KEY CLUSTERED

(

[BOOK\_ISSUE\_NO] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[LMS\_FINE\_DETAILS] Script Date: 01-02-2023 15:17:36 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[LMS\_FINE\_DETAILS](

[FINE\_RANGE] [varchar](3) NOT NULL,

[FINE\_AMOUNT] [decimal](10, 2) NOT NULL,

CONSTRAINT [LMS\_cts3] PRIMARY KEY CLUSTERED

(

[FINE\_RANGE] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[LMS\_MEMBERS] Script Date: 01-02-2023 15:17:36 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[LMS\_MEMBERS](

[MEMBER\_ID] [varchar](10) NOT NULL,

[MEMBER\_NAME] [varchar](30) NOT NULL,

[CITY] [varchar](20) NULL,

[DATE\_REGISTER] [date] NOT NULL,

[DATE\_EXPIRE] [date] NULL,

[MEMBERSHIP\_STATUS] [varchar](15) NOT NULL,

CONSTRAINT [LMS\_cts1] PRIMARY KEY CLUSTERED

(

[MEMBER\_ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[LMS\_SUPPLIERS\_DETAILS] Script Date: 01-02-2023 15:17:36 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[LMS\_SUPPLIERS\_DETAILS](

[SUPPLIER\_ID] [varchar](3) NOT NULL,

[SUPPLIER\_NAME] [varchar](30) NOT NULL,

[ADDRESS] [varchar](50) NULL,

[CONTACT] [bigint] NOT NULL,

[EMAIL] [varchar](15) NOT NULL,

CONSTRAINT [LMS\_cts2] PRIMARY KEY CLUSTERED

(

[SUPPLIER\_ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON, OPTIMIZE\_FOR\_SEQUENTIAL\_KEY = OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

INSERT [dbo].[LMS\_BOOK\_DETAILS] ([BOOK\_CODE], [BOOK\_TITLE], [CATEGORY], [AUTHOR], [PUBLICATION], [PUBLISH\_DATE], [BOOK\_EDITION], [PRICE], [RACK\_NUM], [DATE\_ARRIVAL], [SUPPLIER\_ID]) VALUES (N'BL000001', N'Java How To Do Program', N'JAVA', N'Paul J. Deitel', N'Prentice Hall', CAST(N'1999-12-10' AS Date), 6, CAST(600.00 AS Decimal(8, 2)), N'A1', CAST(N'2011-05-10' AS Date), N'S01')

INSERT [dbo].[LMS\_BOOK\_DETAILS] ([BOOK\_CODE], [BOOK\_TITLE], [CATEGORY], [AUTHOR], [PUBLICATION], [PUBLISH\_DATE], [BOOK\_EDITION], [PRICE], [RACK\_NUM], [DATE\_ARRIVAL], [SUPPLIER\_ID]) VALUES (N'BL000002', N'Java: The Complete Reference ', N'JAVA', N'Herbert Schildt', N'Tata Mcgraw Hill ', CAST(N'2011-10-10' AS Date), 5, CAST(750.00 AS Decimal(8, 2)), N'A1', CAST(N'2011-05-10' AS Date), N'S03')

INSERT [dbo].[LMS\_BOOK\_DETAILS] ([BOOK\_CODE], [BOOK\_TITLE], [CATEGORY], [AUTHOR], [PUBLICATION], [PUBLISH\_DATE], [BOOK\_EDITION], [PRICE], [RACK\_NUM], [DATE\_ARRIVAL], [SUPPLIER\_ID]) VALUES (N'BL000003', N'Java How To Do Program', N'JAVA', N'Paul J. Deitel', N'Prentice Hall', CAST(N'1999-05-10' AS Date), 6, CAST(600.00 AS Decimal(8, 2)), N'A1', CAST(N'2012-05-10' AS Date), N'S01')

INSERT [dbo].[LMS\_BOOK\_DETAILS] ([BOOK\_CODE], [BOOK\_TITLE], [CATEGORY], [AUTHOR], [PUBLICATION], [PUBLISH\_DATE], [BOOK\_EDITION], [PRICE], [RACK\_NUM], [DATE\_ARRIVAL], [SUPPLIER\_ID]) VALUES (N'BL000004', N'Java: The Complete Reference ', N'JAVA', N'Herbert Schildt', N'Tata Mcgraw Hill ', CAST(N'2011-10-10' AS Date), 5, CAST(750.00 AS Decimal(8, 2)), N'A1', CAST(N'2012-05-11' AS Date), N'S01')

INSERT [dbo].[LMS\_BOOK\_DETAILS] ([BOOK\_CODE], [BOOK\_TITLE], [CATEGORY], [AUTHOR], [PUBLICATION], [PUBLISH\_DATE], [BOOK\_EDITION], [PRICE], [RACK\_NUM], [DATE\_ARRIVAL], [SUPPLIER\_ID]) VALUES (N'BL000005', N'Java How To Do Program', N'JAVA', N'Paul J. Deitel', N'Prentice Hall', CAST(N'1999-12-10' AS Date), 6, CAST(600.00 AS Decimal(8, 2)), N'A1', CAST(N'2012-05-11' AS Date), N'S01')

INSERT [dbo].[LMS\_BOOK\_DETAILS] ([BOOK\_CODE], [BOOK\_TITLE], [CATEGORY], [AUTHOR], [PUBLICATION], [PUBLISH\_DATE], [BOOK\_EDITION], [PRICE], [RACK\_NUM], [DATE\_ARRIVAL], [SUPPLIER\_ID]) VALUES (N'BL000006', N'Java: The Complete Reference ', N'JAVA', N'Herbert Schildt Paul', N'Tata Mcgraw Hill ', CAST(N'2011-10-10' AS Date), 5, CAST(750.00 AS Decimal(8, 2)), N'A1', CAST(N'2012-05-12' AS Date), N'S03')

INSERT [dbo].[LMS\_BOOK\_DETAILS] ([BOOK\_CODE], [BOOK\_TITLE], [CATEGORY], [AUTHOR], [PUBLICATION], [PUBLISH\_DATE], [BOOK\_EDITION], [PRICE], [RACK\_NUM], [DATE\_ARRIVAL], [SUPPLIER\_ID]) VALUES (N'BL000007', N'Let Us C', N'C', N'Yashavant Kanetkar ', N'BPB Publications', CAST(N'2010-12-11' AS Date), 9, CAST(500.00 AS Decimal(8, 2)), N'A3', CAST(N'2010-11-03' AS Date), N'S03')

INSERT [dbo].[LMS\_BOOK\_DETAILS] ([BOOK\_CODE], [BOOK\_TITLE], [CATEGORY], [AUTHOR], [PUBLICATION], [PUBLISH\_DATE], [BOOK\_EDITION], [PRICE], [RACK\_NUM], [DATE\_ARRIVAL], [SUPPLIER\_ID]) VALUES (N'BL000008', N'Let Us SQL', N'SQL', N'Yashavant Kanetkar ', N'BPB Publications', CAST(N'2010-05-12' AS Date), 9, NULL, NULL, CAST(N'2011-08-09' AS Date), N'S04')

INSERT [dbo].[LMS\_BOOK\_DETAILS] ([BOOK\_CODE], [BOOK\_TITLE], [CATEGORY], [AUTHOR], [PUBLICATION], [PUBLISH\_DATE], [BOOK\_EDITION], [PRICE], [RACK\_NUM], [DATE\_ARRIVAL], [SUPPLIER\_ID]) VALUES (N'BL000009', N'.Net Core', N'.Net', N'Dhivya', N'CloudKampus Publications', CAST(N'2022-07-10' AS Date), 1, CAST(500.00 AS Decimal(8, 2)), N'A2', CAST(N'2022-07-15' AS Date), NULL)

GO

INSERT [dbo].[LMS\_BOOK\_ISSUE] ([BOOK\_ISSUE\_NO], [MEMBER\_ID], [BOOK\_CODE], [DATE\_ISSUE], [DATE\_RETURN], [DATE\_RETURNED], [FINE\_RANGE]) VALUES (1, N'LM001', N'BL000001', CAST(N'2012-05-01' AS Date), CAST(N'2012-05-16' AS Date), CAST(N'2012-05-16' AS Date), N'R0')

INSERT [dbo].[LMS\_BOOK\_ISSUE] ([BOOK\_ISSUE\_NO], [MEMBER\_ID], [BOOK\_CODE], [DATE\_ISSUE], [DATE\_RETURN], [DATE\_RETURNED], [FINE\_RANGE]) VALUES (2, N'LM002', N'BL000002', CAST(N'2012-05-01' AS Date), CAST(N'2012-05-06' AS Date), CAST(N'2012-05-16' AS Date), N'R2')

INSERT [dbo].[LMS\_BOOK\_ISSUE] ([BOOK\_ISSUE\_NO], [MEMBER\_ID], [BOOK\_CODE], [DATE\_ISSUE], [DATE\_RETURN], [DATE\_RETURNED], [FINE\_RANGE]) VALUES (3, N'LM003', N'BL000007', CAST(N'2012-04-01' AS Date), CAST(N'2012-04-16' AS Date), CAST(N'2012-04-20' AS Date), N'R1')

INSERT [dbo].[LMS\_BOOK\_ISSUE] ([BOOK\_ISSUE\_NO], [MEMBER\_ID], [BOOK\_CODE], [DATE\_ISSUE], [DATE\_RETURN], [DATE\_RETURNED], [FINE\_RANGE]) VALUES (4, N'LM004', N'BL000005', CAST(N'2012-04-01' AS Date), CAST(N'2012-04-16' AS Date), CAST(N'2012-04-20' AS Date), N'R1')

INSERT [dbo].[LMS\_BOOK\_ISSUE] ([BOOK\_ISSUE\_NO], [MEMBER\_ID], [BOOK\_CODE], [DATE\_ISSUE], [DATE\_RETURN], [DATE\_RETURNED], [FINE\_RANGE]) VALUES (5, N'LM005', N'BL000008', CAST(N'2012-03-30' AS Date), CAST(N'2012-04-15' AS Date), CAST(N'2012-04-20' AS Date), N'R1')

INSERT [dbo].[LMS\_BOOK\_ISSUE] ([BOOK\_ISSUE\_NO], [MEMBER\_ID], [BOOK\_CODE], [DATE\_ISSUE], [DATE\_RETURN], [DATE\_RETURNED], [FINE\_RANGE]) VALUES (6, N'LM005', N'BL000008', CAST(N'2012-04-20' AS Date), CAST(N'2012-05-05' AS Date), CAST(N'2012-05-05' AS Date), N'R0')

INSERT [dbo].[LMS\_BOOK\_ISSUE] ([BOOK\_ISSUE\_NO], [MEMBER\_ID], [BOOK\_CODE], [DATE\_ISSUE], [DATE\_RETURN], [DATE\_RETURNED], [FINE\_RANGE]) VALUES (7, N'LM003', N'BL000007', CAST(N'2012-04-22' AS Date), CAST(N'2012-05-07' AS Date), CAST(N'2012-05-25' AS Date), N'R4')

GO

INSERT [dbo].[LMS\_FINE\_DETAILS] ([FINE\_RANGE], [FINE\_AMOUNT]) VALUES (N'R0', CAST(0.00 AS Decimal(10, 2)))

INSERT [dbo].[LMS\_FINE\_DETAILS] ([FINE\_RANGE], [FINE\_AMOUNT]) VALUES (N'R1', CAST(20.00 AS Decimal(10, 2)))

INSERT [dbo].[LMS\_FINE\_DETAILS] ([FINE\_RANGE], [FINE\_AMOUNT]) VALUES (N'R2', CAST(50.00 AS Decimal(10, 2)))

INSERT [dbo].[LMS\_FINE\_DETAILS] ([FINE\_RANGE], [FINE\_AMOUNT]) VALUES (N'R3', CAST(75.00 AS Decimal(10, 2)))

INSERT [dbo].[LMS\_FINE\_DETAILS] ([FINE\_RANGE], [FINE\_AMOUNT]) VALUES (N'R4', CAST(100.00 AS Decimal(10, 2)))

INSERT [dbo].[LMS\_FINE\_DETAILS] ([FINE\_RANGE], [FINE\_AMOUNT]) VALUES (N'R5', CAST(150.00 AS Decimal(10, 2)))

INSERT [dbo].[LMS\_FINE\_DETAILS] ([FINE\_RANGE], [FINE\_AMOUNT]) VALUES (N'R6', CAST(200.00 AS Decimal(10, 2)))

INSERT [dbo].[LMS\_FINE\_DETAILS] ([FINE\_RANGE], [FINE\_AMOUNT]) VALUES (N'R7 ', CAST(225.00 AS Decimal(10, 2)))

INSERT [dbo].[LMS\_FINE\_DETAILS] ([FINE\_RANGE], [FINE\_AMOUNT]) VALUES (N'R8 ', CAST(250.00 AS Decimal(10, 2)))

GO

INSERT [dbo].[LMS\_MEMBERS] ([MEMBER\_ID], [MEMBER\_NAME], [CITY], [DATE\_REGISTER], [DATE\_EXPIRE], [MEMBERSHIP\_STATUS]) VALUES (N'LM001', N'AMIT', N'CHENNAI', CAST(N'2012-02-12' AS Date), CAST(N'2013-02-11' AS Date), N'Temporary')

INSERT [dbo].[LMS\_MEMBERS] ([MEMBER\_ID], [MEMBER\_NAME], [CITY], [DATE\_REGISTER], [DATE\_EXPIRE], [MEMBERSHIP\_STATUS]) VALUES (N'LM002', N'ABDHUL', N'DELHI', CAST(N'2012-04-10' AS Date), CAST(N'2013-04-09' AS Date), N'Temporary')

INSERT [dbo].[LMS\_MEMBERS] ([MEMBER\_ID], [MEMBER\_NAME], [CITY], [DATE\_REGISTER], [DATE\_EXPIRE], [MEMBERSHIP\_STATUS]) VALUES (N'LM003', N'GAYAN', N'CHENNAI', CAST(N'2012-05-13' AS Date), CAST(N'2013-05-12' AS Date), N'Permanent')

INSERT [dbo].[LMS\_MEMBERS] ([MEMBER\_ID], [MEMBER\_NAME], [CITY], [DATE\_REGISTER], [DATE\_EXPIRE], [MEMBERSHIP\_STATUS]) VALUES (N'LM004', N'RADHA', N'CHENNAI', CAST(N'2012-04-22' AS Date), CAST(N'2013-04-21' AS Date), N'Temporary')

INSERT [dbo].[LMS\_MEMBERS] ([MEMBER\_ID], [MEMBER\_NAME], [CITY], [DATE\_REGISTER], [DATE\_EXPIRE], [MEMBERSHIP\_STATUS]) VALUES (N'LM005', N'GURU', N'BANGALORE', CAST(N'2012-03-30' AS Date), CAST(N'2013-05-16' AS Date), N'Temporary')

INSERT [dbo].[LMS\_MEMBERS] ([MEMBER\_ID], [MEMBER\_NAME], [CITY], [DATE\_REGISTER], [DATE\_EXPIRE], [MEMBERSHIP\_STATUS]) VALUES (N'LM006', N'MOHAN', N'CHENNAI', CAST(N'2012-04-12' AS Date), CAST(N'2013-05-16' AS Date), N'Temporary')

INSERT [dbo].[LMS\_MEMBERS] ([MEMBER\_ID], [MEMBER\_NAME], [CITY], [DATE\_REGISTER], [DATE\_EXPIRE], [MEMBERSHIP\_STATUS]) VALUES (N'LM007', N'Vishnu', N'Hyderbad', CAST(N'2001-01-23' AS Date), CAST(N'2005-01-23' AS Date), N'Permanent')

INSERT [dbo].[LMS\_MEMBERS] ([MEMBER\_ID], [MEMBER\_NAME], [CITY], [DATE\_REGISTER], [DATE\_EXPIRE], [MEMBERSHIP\_STATUS]) VALUES (N'LM008', N'Barga', N'Pune', CAST(N'2016-10-20' AS Date), CAST(N'2017-10-20' AS Date), N'Permanent')

GO

INSERT [dbo].[LMS\_SUPPLIERS\_DETAILS] ([SUPPLIER\_ID], [SUPPLIER\_NAME], [ADDRESS], [CONTACT], [EMAIL]) VALUES (N'S01', N'SINGAPORE SHOPPEE', N'CHENNAI', 9894123555, N'sing@gmail.com')

INSERT [dbo].[LMS\_SUPPLIERS\_DETAILS] ([SUPPLIER\_ID], [SUPPLIER\_NAME], [ADDRESS], [CONTACT], [EMAIL]) VALUES (N'S02', N'JK Stores', N'MUMBAI', 9940123450, N'jks@yahoo.com')

INSERT [dbo].[LMS\_SUPPLIERS\_DETAILS] ([SUPPLIER\_ID], [SUPPLIER\_NAME], [ADDRESS], [CONTACT], [EMAIL]) VALUES (N'S03', N'ROSE BOOK STORE', N'TRIVANDRUM', 9444411222, N'rose@gmail.com')

INSERT [dbo].[LMS\_SUPPLIERS\_DETAILS] ([SUPPLIER\_ID], [SUPPLIER\_NAME], [ADDRESS], [CONTACT], [EMAIL]) VALUES (N'S04', N'KAVARI STORE', N'DELHI', 8630001452, N'kavi@redif.com')

INSERT [dbo].[LMS\_SUPPLIERS\_DETAILS] ([SUPPLIER\_ID], [SUPPLIER\_NAME], [ADDRESS], [CONTACT], [EMAIL]) VALUES (N'S05', N'EINSTEN BOOK GALLARY', N'US', 9542000001, N'eingal@aol.com')

INSERT [dbo].[LMS\_SUPPLIERS\_DETAILS] ([SUPPLIER\_ID], [SUPPLIER\_NAME], [ADDRESS], [CONTACT], [EMAIL]) VALUES (N'S06', N'AKBAR STORE', N'MUMBAI', 7855623100, N'akbakst@aol.com')

GO

/\*ALTER TABLE [dbo].[LMS\_BOOK\_DETAILS] WITH CHECK ADD CONSTRAINT [LMS\_cts41] FOREIGN KEY([SUPPLIER\_ID])

REFERENCES [dbo].[LMS\_SUPPLIERS\_DETAILS] ([SUPPLIER\_ID])

GO

ALTER TABLE [dbo].[LMS\_BOOK\_DETAILS] CHECK CONSTRAINT [LMS\_cts41]

GO

ALTER TABLE [dbo].[LMS\_BOOK\_ISSUE] WITH CHECK ADD CONSTRAINT [LMS\_BookDetail] FOREIGN KEY([BOOK\_CODE])

REFERENCES [dbo].[LMS\_BOOK\_DETAILS] ([BOOK\_CODE])

GO

ALTER TABLE [dbo].[LMS\_BOOK\_ISSUE] CHECK CONSTRAINT [LMS\_BookDetail]

GO

ALTER TABLE [dbo].[LMS\_BOOK\_ISSUE] WITH CHECK ADD CONSTRAINT [LMS\_FineDetail] FOREIGN KEY([FINE\_RANGE])

REFERENCES [dbo].[LMS\_FINE\_DETAILS] ([FINE\_RANGE])

GO

ALTER TABLE [dbo].[LMS\_BOOK\_ISSUE] CHECK CONSTRAINT [LMS\_FineDetail]

GO

ALTER TABLE [dbo].[LMS\_BOOK\_ISSUE] WITH CHECK ADD CONSTRAINT [LMS\_Mem] FOREIGN KEY([MEMBER\_ID])

REFERENCES [dbo].[LMS\_MEMBERS] ([MEMBER\_ID])

GO

ALTER TABLE [dbo].[LMS\_BOOK\_ISSUE] CHECK CONSTRAINT [LMS\_Mem]

GO

\*/

select \*from LMS\_BOOK\_DETAILS;

SELECT \* FROM LMS\_SUPPLIERS\_DETAILS;

---display bookcode,bookname and supplier name of the books in library

--INNER JOIN

select BOOK\_CODE,BOOK\_TITLE,SUPPLIER\_NAME

FROM LMS\_BOOK\_DETAILS JOIN LMS\_SUPPLIERS\_DETAILS ON

LMS\_BOOK\_DETAILS.SUPPLIER\_ID=LMS\_SUPPLIERS\_DETAILS.SUPPLIER\_ID;

---USE AS FOR TABLE

select L1.BOOK\_CODE,L1.BOOK\_TITLE,L2.SUPPLIER\_NAME

FROM LMS\_BOOK\_DETAILS AS L1 JOIN LMS\_SUPPLIERS\_DETAILS AS L2 ON

L1.SUPPLIER\_ID=L2.SUPPLIER\_ID;

----N TABLES NEED N-1 JOINS

--BOOK\_TITLE AND THE MEMBER\_ID WHO HAS TAKEN THE BOOKS FROM LIBRARY

SELECT \* FROM LMS\_BOOK\_DETAILS;

SELECT \* FROM LMS\_MEMBERS;

SELECT \* FROM LMS\_BOOK\_ISSUE;

SELECT DISTINCT BOOK\_TITLE,MEMBER\_ID FROM LMS\_BOOK\_DETAILS JOIN LMS\_BOOK\_ISSUE

ON LMS\_BOOK\_DETAILS.BOOK\_CODE=LMS\_BOOK\_ISSUE.BOOK\_CODE;

--DISPLAY THE SIPPLIER\_ID AND NUMBER\_OF BOOKS SUPPLIED BY EACH SUPPLIER

SELECT \* FROM LMS\_SUPPLIERS\_DETAILS;

SELECT \* FROM LMS\_BOOK\_DETAILS;

SELECT SUPPLIER\_ID,COUNT(BOOK\_CODE) FROM LMS\_BOOK\_DETAILS WHERE SUPPLIER\_ID IS NOT NULL

GROUP BY SUPPLIER\_ID;

--DISPLAY THE SUPPLIER\_NAME AND NUMBER OF BOOK SUPPLIED BT EACH SUPPLIER

SELECT L2.SUPPLIER\_NAME ,COUNT(L1.BOOK\_CODE) AS COUNTOFBOOKS FROM LMS\_BOOK\_DETAILS AS L1 JOIN

LMS\_SUPPLIERS\_DETAILS AS L2 ON L1.SUPPLIER\_ID=L2.SUPPLIER\_ID GROUP BY

L2.SUPPLIER\_NAME;

SELECT L2.SUPPLIER\_NAME,L2.SUPPLIER\_ID,COUNT(BOOK\_CODE) AS COUNT FROM

LMS\_BOOK\_DETAILS AS L1 JOIN LMS\_SUPPLIERS\_DETAILS AS L2

ON L1.SUPPLIER\_ID=L2.SUPPLIER\_ID GROUP BY L2.SUPPLIER\_NAME,

L2.SUPPLIER\_ID;

-----------PRACTICE ONE --------------------------------------

--Write a query to display the book title,

--supplier name who has supplied books to library.

SELECT BOOK\_TITLE,SUPPLIER\_NAME FROM LMS\_BOOK\_DETAILS AS D JOIN

LMS\_SUPPLIERS\_DETAILS AS S ON S.SUPPLIER\_ID=D.SUPPLIER\_ID;

--Write a query to display member name and book code

--for the book he/she taken from the library.

SELECT \* FROM LMS\_MEMBERS;

SELECT \* FROM LMS\_BOOK\_ISSUE;

SELECT MEMBER\_NAME,BOOK\_CODE FROM LMS\_MEMBERS AS M JOIN LMS\_BOOK\_ISSUE AS

B ON M.MEMBER\_ID=B.MEMBER\_ID;

--Write a query to display the book title and date when the book

--was taken from the library.

SELECT BOOK\_TITLE,DATE\_ISSUE FROM LMS\_BOOK\_DETAILS AS D JOIN LMS\_BOOK\_ISSUE AS

I ON D.BOOK\_CODE=I.BOOK\_CODE;

--Write a query to display the member id and

--the fine amount paid by the member in the library.

SELECT \* FROM LMS\_BOOK\_ISSUE;

SELECT \* FROM LMS\_FINE\_DETAILS;

SELECT DISTINCT MEMBER\_ID,FINE\_AMOUNT FROM LMS\_BOOK\_ISSUE AS I JOIN LMS\_FINE\_DETAILS

AS D ON I.FINE\_RANGE=D.FINE\_RANGE;

--Write a query to display the member name and

-- the date difference between the date return and date returned.

SELECT M.MEMBER\_NAME, DATEDIFF(DD,I.DATE\_RETURN,I.DATE\_RETURNED) AS DIFFER

FROM LMS\_MEMBERS AS M JOIN LMS\_BOOK\_ISSUE AS I ON M.MEMBER\_ID=I.MEMBER\_ID;

SELECT \* FROM LMS\_BOOK\_ISSUE;

--Write a query to display the book title and the member

-- name of member who has taken book from library.

SELECT \* FROM LMS\_BOOK\_DETAILS;

SELECT \* FROM LMS\_MEMBERS;

SELECT \* FROM LMS\_BOOK\_ISSUE;

SELECT L1.BOOK\_TITLE,L3.MEMBER\_NAME FROM LMS\_BOOK\_DETAILS AS L1

JOIN LMS\_BOOK\_ISSUE AS L2 ON L1.BOOK\_CODE=L2.BOOK\_CODE

JOIN LMS\_MEMBERS AS L3 ON L2.MEMBER\_ID=L3.MEMBER\_ID;

--Write a query to display the book title, member name and

--the supplier name for the books taken from the library.

SELECT BOOK\_TITLE,MEMBER\_NAME,SUPPLIER\_NAME FROM LMS\_BOOK\_DETAILS AS L1

JOIN LMS\_SUPPLIERS\_DETAILS AS L2 ON L1.SUPPLIER\_ID=L2.SUPPLIER\_ID JOIN

LMS\_BOOK\_ISSUE AS L3 ON L1.BOOK\_CODE=L3.BOOK\_CODE JOIN LMS\_MEMBERS AS L4

ON L3.MEMBER\_ID=L4.MEMBER\_ID;

--Write a query to display the book title and

--the difference fine amount paid for the book in library.

SELECT L1.BOOK\_TITLE, DIFFERENCE(L3.FINE\_RANGE,L3.FINE\_AMOUNT) AS FINE\_DIFFER

FROM

LMS\_BOOK\_DETAILS AS L1 JOIN LMS\_BOOK\_ISSUE AS L2 ON

L1.BOOK\_CODE=L2.BOOK\_CODE JOIN LMS\_FINE\_DETAILS AS L3 ON

L2.FINE\_RANGE=L3.FINE\_RANGE;

SELECT BOOK\_TITLE,FINE\_AMOUNT FROM LMS\_BOOK\_DETAILS AS L1 JOIN LMS\_BOOK\_ISSUE

AS L2 ON L1.BOOK\_CODE=L2.BOOK\_CODE JOIN LMS\_FINE\_DETAILS AS L3

ON L2.FINE\_RANGE=L3.FINE\_RANGE;

SELECT \* FROM LMS\_BOOK\_DETAILS;

SELECT \* FROM LMS\_BOOK\_ISSUE;

SELECT \*FROM LMS\_FINE\_DETAILS;

------------AVG PROBLEMS------------------------------------------------------

----------- 3\_FEB

--having clause

--find supplier details who has supplied max no of books

select \* from LMS\_SUPPLIERS\_DETAILS;

select \* from LMS\_BOOK\_DETAILS;

--select supplier\_id,max(count(supplier\_id)) from

--lms\_book\_details group by supplier\_id;

select SUPPLIER\_ID, count(supplier\_id) from lms\_book\_details group by supplier\_id

HAVING COUNT(SUPPLIER\_ID) > 2;

--DISPLAY MEM ID WHO HAVE TAKEN LESS THAN 2 BOOKS

SELECT MEMBER\_ID,COUNT(BOOK\_CODE) FROM LMS\_BOOK\_ISSUE GROUP BY MEMBER\_ID HAVING

COUNT(BOOK\_CODE)<2;

SELECT \* FROM LMS\_MEMBERS;

SELECT MEMBER\_NAME, COUNT(BOOK\_CODE) FROM LMS\_BOOK\_DETAILS AS L1 JOIN

LMS\_MEMBERS AS L2 ON L1.MEMBER\_ID=L2.MEMBER\_ID GROUP BY MEMBER\_NAME HAVING

COUNT(BOOK\_CODE)<2;

SELECT MEMBER\_NAME, COUNT(BOOK\_CODE) AS NOOFBOOKS FROM LMS\_BOOK\_ISSUE AS L1

JOIN LMS\_MEMBERS AS L2 ON L1.MEMBER\_ID=L2.MEMBER\_ID WHERE MEMBER\_NAME LIKE 'A%'

GROUP BY MEMBER\_NAME HAVING COUNT(BOOK\_CODE)<2 ORDER BY MEMBER\_NAME DESC;

--\*\*\* DENSE\_RANK() FROM EMPLOYEE \*\*\*---------------------------------

SELECT \* FROM (SELECT FIRST\_NAME,SALARY,DENSE\_RANK() OVER(ORDER BY SALARY DESC)

R FROM EMPLOYEE) AS E WHERE R=5;

=============================================================================

--Write a query to display the member id, member name of the members,

-- book code and book title of the books taken by them

SELECT L1.MEMBER\_ID,L1.MEMBER\_NAME ,L3.BOOK\_CODE,L3.BOOK\_TITLE FROM

LMS\_MEMBERS AS L1

JOIN LMS\_BOOK\_ISSUE AS L2 ON L1.MEMBER\_ID=L2.MEMBER\_ID JOIN LMS\_BOOK\_DETAILS

AS L3 ON L2.BOOK\_CODE=L3.BOOK\_CODE;

--2Write a query to display the total number of books available in the

-- library with alias name “NO\_OF\_BOOKS\_AVAILABLE” (Which is not issued).

--Hint: The issued books details are available in the LMS\_BOOK\_ISSUE table.

SELECT COUNT(L1.BOOK\_CODE) AS NO\_OF\_BOOKS\_AVAILABLE FROM LMS\_BOOK\_DETAILS AS L1 JOIN

LMS\_BOOK\_ISSUE AS L2 ON L1.BOOK\_CODE=L2.BOOK\_CODE WHERE L2.BOOK\_CODE; IS NULL;

SELECT COUNT(t1.BOOK\_CODE)AS NO\_OF\_BOOKS\_AVAILABLE FROM LMS\_BOOK\_DETAILS t1

LEFT JOIN LMS\_BOOK\_ISSUE t2 ON t1.BOOK\_CODE = t2.BOOK\_CODE WHERE t2.BOOK\_CODE IS NULL

--3Write a query to display the member id, member name, fine range and

--fine amount of the members whose fine amount is less than 100.

--SELECT L1.MEMBER\_ID,L1.MEMBER\_NAME,L3.FINE\_RANGE,L3.FINE\_AMOUNT FROM

--LMS\_MEMBERS AS L1

--JOIN LMS\_BOOK\_ISSUE AS L2 ON L1.MEMBER\_ID=L2.MEMBER\_ID JOIN LMS\_FINE\_DETAILS

--AS L3 ON L2.FINE\_RANGE=L3.FINE\_RANGE GROUP BY L1.MEMBER\_ID,L1.MEMBER\_NAME,L3.FINE\_RANGE

--L3.FINE\_AMOUNT HAVING L3.FINE\_AMOUNT<100;

SELECT L1.MEMBER\_ID,L1.MEMBER\_NAME,L3.FINE\_RANGE,

L3.FINE\_AMOUNT

FROM

LMS\_MEMBERS AS L1

JOIN LMS\_BOOK\_ISSUE AS L2 ON L1.MEMBER\_ID=L2.MEMBER\_ID JOIN LMS\_FINE\_DETAILS

AS L3 ON L2.FINE\_RANGE=L3.FINE\_RANGE GROUP BY L3.FINE\_AMOUNT,l1.MEMBER\_ID,l1.MEMBER\_NAME,l3.FINE\_RANGE

HAVING L3.FINE\_AMOUNT<100;

--4Write a query to display the book code, book title, publisher,

--edition, price and year of publication

--and sort based on year of publication, publisher and edition.

--SELECT BOOK\_CODE,BOOK\_TITLE,AUTHOR,book\_edition,price,publiSH\_DATE

--FROM LMS\_BOOK\_DETAILS ORDER BY PUBLISH\_DATE,AUTHOR,BOOK\_EDITION;

SELECT BOOK\_CODE,BOOK\_TITLE,PUBLICATION,BOOK\_EDITION,PRICE,YEAR(PUBLISH\_DATE)

AS YEAR\_OF\_PUBLICATION FROM LMS\_BOOK\_DETAILS ORDER BY YEAR\_OF\_PUBLICATION;

--5Write a query to display the book code, book title and rack number of

-- the books

-- which are placed in rack 'A1' and sort by book title in ascending order.

SELECT BOOK\_CODE,BOOK\_TITLE,RACK\_NUM FROM LMS\_BOOK\_DETAILS WHERE RACK\_NUM='A1' ORDER BY

BOOK\_TITLE ASC;

SELECT \* FROM lMS\_BOOK\_DETAILS;

--6Write a query to display the member id, member name, due date and date returned

--of the members who has returned the books after the due date.

--Hint: Date\_return is due date and Date\_returned is actual book return date.

SELECT MEMBER\_ID,MEMBER\_NAME,DATE\_RETURN,DATE\_RETURNED FROM LMS\_BOOK\_ISSUE