

### Assignment No 3

#### Problem Statement : Implementation of SQL- Data Manipulation Language – Joins

Create the following tables and solve the queries:

Member( M\_id, m\_name, m\_address,m\_phno)

Books( Book\_id, bookname, bookauthor,bookpub,bookcost, DOPub,M\_id(FK))

Staff( staff\_id, staff\_name, staff\_address, staff\_gender)

Issue

M_id(FK)	Book_id(FK)	Issue_no	Idate	Rdate	fine

(PK)

#### 1. Display Display Book\_id, bookname,cost,member name and phno.

Select Books.Book\_id, Books.Book\_Name , Books.Book\_cost, Members.Member\_Name, Members.Member\_PhoneNo from Books

INNER JOIN Members

ON Books.Member\_id=Members.Member\_id;

Book_id	Book_Name	Book_cost	Member_Name	Member_PhoneNo
1200	The Subtle Art of Not Giving a Fuck	263	Madhura	8967564567
1201	Harry Potter	370	Ahana	8757979078
1203	The Room on the Roof	210	Pallavi	7867564567
1205	Pathways to Light	221	Aarya	6876890890
1206	Rich Dad Poor Dad	300	Rasika	7879789876
1207	Endless Night	500	Sanika	7867564534
1208	Five Little Pigs	500	Pratham	8756789087
1209	The Story of My Life	600	Rajesh	9898768678
1210	Crooked House	450	Sahil	7654568968

#### 2. Write a query to count the number of books owned by Ahana.

SELECT Books.Book\_Name,COUNT(\*) FROM Books,Members WHERE Books.Member\_id=Members.Member\_id AND Members.Member\_Name='Ahana';

Book_Name	COUNT(*)
Harry Potter	1

#### 3. Write a query to display sum of cost of all the books issued by “Ahana”

SELECT sum(Book\_cost) from Books, Members where Books.Member\_id=Members.Member\_id and Member\_name ='Ahana';

sum(Book_cost)
370

#### 4. Write a query to display the book code, book title of the books which are issued on the date “June 2021” (inner join)

SELECT Books.Book\_id,Books.Book\_Name FROM Books RIGHT JOIN Issues ON  
Books.Book\_id=Issues.Book\_id WHERE monthname(issues.issueDate)='June' AND  
year(issues.issueDate)=2021;

Book_id	Book_Name
1209	The Story of My Life
NULL	NULL
NULL	NULL

5. Write a query to display the member id, member name, address of members with the total fine paid by them with alias name “Fine\_Amount”. (inner join)

Select Members.Member\_id, Members .Member\_name, Members.Member\_address, sum(issues.fine)  
as Fine\_Amount from Members INNER JOIN issues ON

Members.Member\_id = issues.Member\_id group by issues.Member\_id;

	Member_id	Member_name	Member_address	Fine_Amount
▶	101	Ahana	VimanNagar	100
	102	Sneha	Baner	0
	103	Pallavi	KalyaniNagar	0
	104	Purva	Araneshwar Park	25
	105	Aarya	Surat	10

6. Write a query to display list of all members who have issued the book “The Room On The Roof”.

SELECT Members.Member\_id,Members.Member\_Name FROM Members INNER JOIN Books ON  
Books.Member\_id =Members.member\_id WHERE Book\_Name='The Room On The Roof';

Member_id	Member_Name
103	Pallavi

7. List the member details along with the book names they have issued.

SELECT Members.Member\_id,Members.Member\_Name,  
Members.Member\_PhoneNo,Members.Member\_Address,Books.Book\_Name FROM Members  
INNER JOIN Books ON Members.Member\_id=Books.Member\_id;

Member_id	Member_Name	Member_PhoneNo	Member_Address	Book_Name
101	Ahana	8757979078	VimanNagar	Harry Potter
107	Sanika	7867564534	Sahakar Nagar	Endless Night
108	Pratham	8756789087	Pawai	Five Little Pigs
100	Madhura	8967564567	Padmavati	The Subtle Art of Not Giving a Fuck
110	Sahil	7654568968	Navi Mumbai	Crooked House
103	Pallavi	7867564567	KalyaniNagar	The Room on the Roof
106	Rasika	7879789876	FCRoad	Rich Dad Poor Dad
105	Aarya	6876890890	Delhi	Pathways to Light
109	Rajesh	9898768678	Dadar	The Story of My Life

**8.Display all members and any books they might have have issued (use left and right join)**

SELECT \* FROM Members LEFT JOIN Books ON Members.Member\_id=Books.Member\_id;

Member_id	Member_Name	Member_Address	Member_PhoneNo	Book_id	Book_Name	Book_Author	Book_Publication	Book_cost	DOPublication	Member_id
100	Madhura	Padmavati	8967564567	1200	The Subtle Art of Not Giving a Fuck	Mark Manson	HarperCollins Publishers	263	2017-05-10	100
101	Ahana	VimanNagar	8757979078	1201	Harry Potter	J.K.Rowling	Penguin Publication	370	2001-07-23	101
103	Pallavi	KalyaniNagar	7867564567	1203	The Room on the Roof	Ruskin Bond	HarperCollins	210	2000-08-13	103
105	Aarya	Delhi	6876890890	1205	Pathways to Light	Dr Prakash Amte	Penguin Publication	221	2000-03-08	105
106	Rasika	FCRoad	7879789876	1206	Rich Dad Poor Dad	Robert T. Kiyosaki	Penguin Publication	300	2008-06-03	106
107	Sanika	Sahakar Nagar	7867564534	1207	Endless Night	Agatha Christi	HarperCollins Publication	500	2009-05-09	107
108	Pratham	Pawai	8756789087	1208	Five Little Pigs	Agatha Christi	HarperCollins	500	2007-06-19	108
109	Rajesh	Dadar	9898768678	1209	The Story of My Life	Hellen Keller	Simon & Schuster	600	1976-06-19	109
110	Sahil	Navi Mumbai	7654568968	1210	Crooked House	Agatha Christi	HarperCollins	450	2004-06-19	110
111	Rudr	ShivajiNagar	9867897867	NULL	NULL	NULL	NULL	NULL	NULL	NULL
112	Ridhi	Katraj	9865345678	NULL	NULL	NULL	NULL	NULL	NULL	NULL

**9. Write a query to display the total number of books available in the library with alias name "NO\_OF\_BOOKS\_AVAILABLE"**

SELECT COUNT(\*) AS "No. Of Books Available" FROM Books;

No. Of Books Available
11

**10. Find those books whose cost exists between 500 and 2000. Return bookid, book name,member,name,mem id.**

SELECT Book\_id, Book\_Name, Members.Member\_id, Members.Member\_name from Books left join Members ON

books.Member\_id=Members.member\_id where book\_cost between 500 AND 2000;

Book_id	Book_Name	Member_id	Member_name
1207	Endless Night	107	Sanika
1208	Five Little Pigs	108	Pratham
1209	The Story of My Life	109	Rajesh

**11. Write a SQL statement to make a join on the tables books, members and issue in such a form that the same column of each table will appear once and only the relational rows will come.(natural join)**

SELECT \* FROM Books NATURAL JOIN Members NATURAL JOIN Issues;

Member_id	Book_id	Book_Name	Book_Author	Book_Publication	Book_cost	DOPublication	Member_Name	Member_Address	Member_PhoneNo	staff_id	IssueDate	ReturnDate	fine
101	1201	Harry Potter	J.K.Rowling	Penguin Publication	370	2001-07-23	Ahana	VimanNagar	8757979078	1	2020-02-25	2020-06-30	0
109	1209	The Story of My Life	Hellen Keller	Simon & Schuster	600	1976-06-19	Rajesh	Dadar	9898768678	2	2021-06-05	2021-07-07	0

**12. Create a view to access m\_id,m\_name,m\_address,m\_phno. Find out whether the view is an updatable view.**

Create View V As

Select Member\_id,Member\_name,Member\_address,Member\_phoneNo

From Members;

Select \* from V;

Member_id	Member_name	Member_address	Member_phoneNo
100	Madhura	Padmavati	8967564567
101	Ahana	VimanNagar	8757979078
103	Pallavi	KalyaniNagar	7867564567
105	Aarya	Delhi	6876890890
106	Rasika	FCRoad	7879789876
107	Sanika	Sahakar Nagar	7867564534
108	Pratham	Pawai	8756789087
109	Rajesh	Dadar	9898768678
110	Sahil	Navi Mumbai	7654568968
111	Rudr	ShivajiNagar	9867897867
112	Ridhi	Katraj	9865345678

**13. Create a view to access m\_name,book\_id and bookpub. Find out whether the view is an updatable view.**

Create View Pview As

Select Members.Member\_name,books.book\_id ,books.book\_Publication

From Members,books where books.Member\_id=Members.Member\_id;

SELECT \* FROM Pview;

	Member_name	book_id	book_Publication
►	Madhura	1200	HarperCollins Publishers
	Ahana	1201	Penguin Publication
	Sneha	1202	Malgudi Publishers
	Pallavi	1203	HarperCollins
	Purva	1204	Penguin Random House
	Aarya	1205	Penguin Publication
	Rasika	1206	Penguin Publication
	Sanika	1207	HarperCollins

UPDATE View Pview SET Member\_name='Aditi' where Member\_name='Sneha';

**The view is an updatable view.**

#### 14. Create suitable index on books table and find the performance improvement

**CREATE UNIQUE INDEX perf ON books( book\_id);**

The results show up in 0 seconds which is faster than earlier. Hence, there is performance improvement after indexing.

**show indexes from books;**

	Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expression
▶	books	0	PRIMARY	1	Book_id	A	11	NULL	NULL		BTREE			YES	NULL
	books	0	perf	1	Book_id	A	11	NULL	NULL		BTREE			YES	NULL
	books	1	Member_id	1	Member_id	A	11	NULL	NULL	YES	BTREE			YES	NULL
	books	1	performance	1	Book_Publication	A	6	NULL	NULL		BTREE			YES	NULL

#### 15. Display list of all members who live in the same city as “Rasika”.(subquery)

Select Member\_name from Members

Where Member\_address=

(Select Member\_address from members where Member\_name='Rasika');

Member_name
Aarya
Rasika

#### 16. Write a query that uses sub query to obtain all the books owned by “Ahana”.

Select \* from Books

Where Member\_id=

(Select Member\_id from members where Member\_name='Ahana');

Book_id	Book_Name	Book_Author	Book_Publication	Book_cost	DOPublication	Member_id
1201	Harry Potter	J.K.Rowling	Penguin Publication	370	2001-07-23	101
1213	Fault In our Stars	John Green	Dutton Books	600	2010-04-08	101
NULL	NULL	NULL	NULL	NULL	NULL	NULL

#### 17. Display the name of member who has paid highest fine for the book.(subquery)

Select Member\_name from Members,Issues Where issues.Member\_id=Members.Member\_id AND Issues.fine= (Select MAX(fine) from issues);

#### 18. Display members who have issued books (subquery-use IN)

SELECT \* FROM Members WHERE Member\_id in (SELECT Member\_id FROM Issues);

Member_id	Member_Name	Member_Address	Member_PhoneNo
101	Ahana	VimanNagar	8757979078
103	Pallavi	KalyaniNagar	7867564567
105	Aarya	Delhi	6876890890
106	Rasika	Delhi	7879789876
108	Pratham	Pawai	8756789087
109	Rajesh	Dadar	9898768678
110	Sahil	Navi Mumbai	7654568968
NULL	NULL	NULL	NULL

**19. Display the list of members id who are members but they have not issued any book. (subquery-NOT IN)**

SELECT \* FROM Members WHERE Member\_id not in (SELECT Member\_id FROM Issues);

Member_id	Member_Name	Member_Address	Member_PhoneNo
100	Madhura	Padmavati	8967564567
107	Sanika	Sahakar Nagar	7867564534
111	Rudr	ShivajiNagar	9867897867
112	Ridhi	Katraj	9865345678
NULL	NULL	NULL	NULL