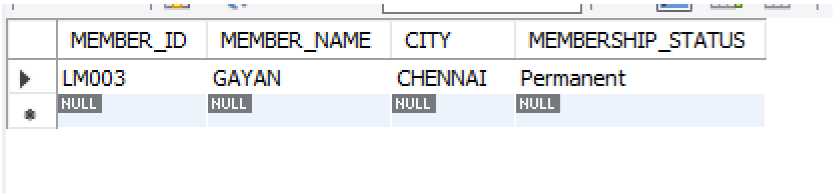
WEEKLY TASK

NAME : SANJEEV
DATE : 14-07-2025

SIMPLE PROBLEMS

1. **Membership status who are all having life time membership**

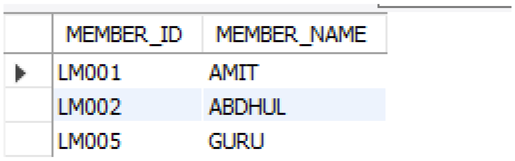
```
SELECT MEMBER_ID, MEMBER_NAME, CITY, MEMBERSHIP_STATUS
FROM LMS_MEMBERS
WHERE MEMBERSHIP_STATUS = 'Permanent';
```



MEMBER_ID	MEMBER_NAME	CITY	MEMBERSHIP_STATUS
LM003	GAYAN	CHENNAI	Permanent
NULL	NULL	NULL	NULL

2. **Member name who have not returned the books**

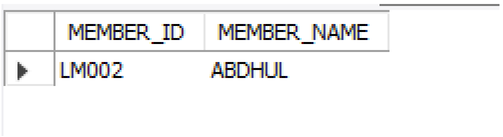
```
SELECT DISTINCT M.MEMBER_ID, M.MEMBER_NAME
FROM LMS_MEMBERS M
JOIN LMS_BOOK_ISSUE B ON M.MEMBER_ID = B.MEMBER_ID
WHERE B.BOOK_ISSUE_STATUS = 'N';
```



MEMBER_ID	MEMBER_NAME
LM001	AMIT
LM002	ABDHUL
LM005	GURU

3. **Members who took book code 'BL000002'**

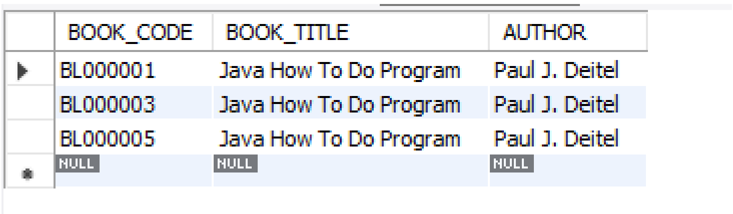
```
SELECT DISTINCT M.MEMBER_ID, M.MEMBER_NAME
FROM LMS_MEMBERS M
JOIN LMS_BOOK_ISSUE B ON M.MEMBER_ID = B.MEMBER_ID
WHERE B.BOOK_CODE = 'BL000002';
```



	MEMBER_ID	MEMBER_NAME
▶	LM002	ABDHUL

4. **Books with author starting with 'P'**

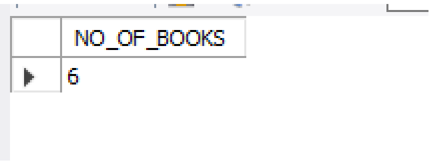
```
SELECT BOOK_CODE, BOOK_TITLE, AUTHOR
FROM LMS_BOOK_DETAILS
WHERE AUTHOR LIKE 'P%';
```



	BOOK_CODE	BOOK_TITLE	AUTHOR
▶	BL000001	Java How To Do Program	Paul J. Deitel
	BL000003	Java How To Do Program	Paul J. Deitel
	BL000005	Java How To Do Program	Paul J. Deitel
*	NULL	NULL	NULL

5. **Total number of JAVA books**

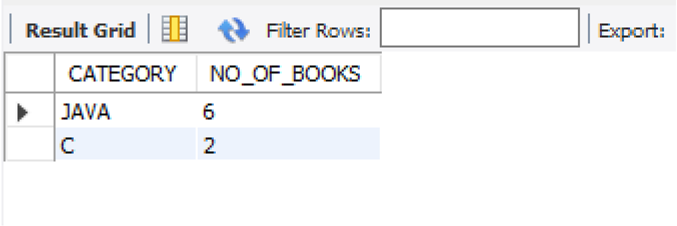
```
SELECT COUNT(*) AS NO_OF_BOOKS
FROM LMS_BOOK_DETAILS
WHERE CATEGORY = 'JAVA';
```



	NO_OF_BOOKS
▶	6

6. **Number of books in each category**

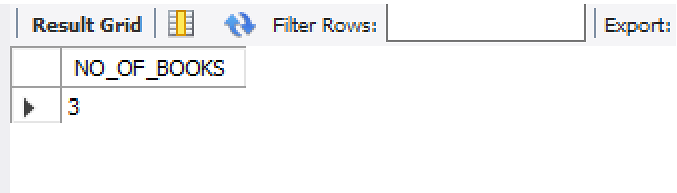
```
SELECT CATEGORY, COUNT(*) AS NO_OF_BOOKS
FROM LMS_BOOK_DETAILS
GROUP BY CATEGORY;
```



CATEGORY	NO_OF_BOOKS
JAVA	6
C	2

7. Number of books published by 'Prentice Hall'

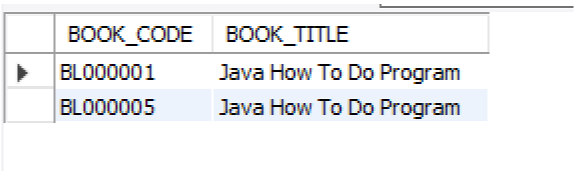
```
SELECT COUNT(*) AS NO_OF_BOOKS
FROM LMS_BOOK_DETAILS
WHERE PUBLICATION = 'Prentice Hall';
```



NO_OF_BOOKS
3

8. Books issued on '2012-04-01'

```
SELECT BOOK_CODE, BOOK_TITLE
FROM LMS_BOOK_DETAILS
WHERE BOOK_CODE IN (
SELECT BOOK_CODE
FROM LMS_BOOK_ISSUE
WHERE DATE_ISSUE = '2012-04-01');
```

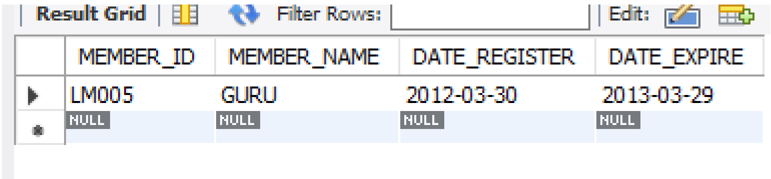


BOOK_CODE	BOOK_TITLE
BL000001	Java How To Do Program
BL000005	Java How To Do Program

9. **Members with membership expiry before April 2013**

```
SELECT MEMBER_ID, MEMBER_NAME, DATE_REGISTER, DATE_EXPIRE
FROM LMS_MEMBERS
```

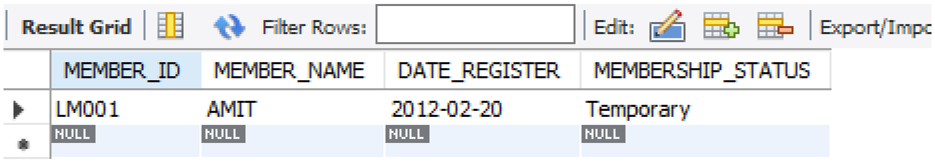
```
WHERE DATE_EXPIRE < '2013-04-01';
```



	MEMBER_ID	MEMBER_NAME	DATE_REGISTER	DATE_EXPIRE
▶	LM005	GURU	2012-03-30	2013-03-29
*	NULL	NULL	NULL	NULL

10. Members registered before March 2012 and status is Temporary

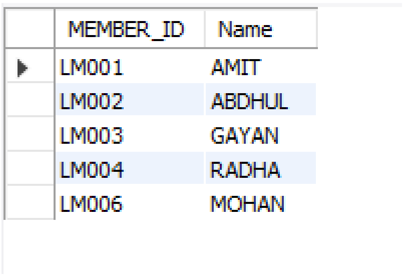
```
SELECT MEMBER_ID, MEMBER_NAME, DATE_REGISTER, MEMBERSHIP_STATUS
FROM LMS_MEMBERS
WHERE DATE_REGISTER < '2012-03-01'
  AND MEMBERSHIP_STATUS = 'Temporary';
```



	MEMBER_ID	MEMBER_NAME	DATE_REGISTER	MEMBERSHIP_STATUS
▶	LM001	AMIT	2012-02-20	Temporary
*	NULL	NULL	NULL	NULL

11. Members from Chennai or Delhi, name in title case

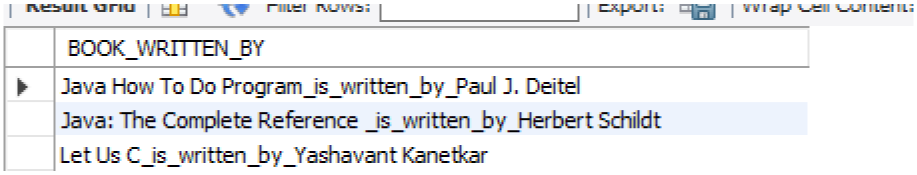
```
SELECT MEMBER_ID,
CONCAT(UPPER(LEFT(MEMBER_NAME,1)),
LOWER(SUBSTRING(MEMBER_NAME,2))) AS Name
FROM LMS_MEMBERS
WHERE CITY IN ('CHENNAI', 'DELHI');
```



	MEMBER_ID	Name
▶	LM001	AMIT
	LM002	ABDHUL
	LM003	GAYAN
	LM004	RADHA
	LM006	MOHAN

12. Book title concatenated with author

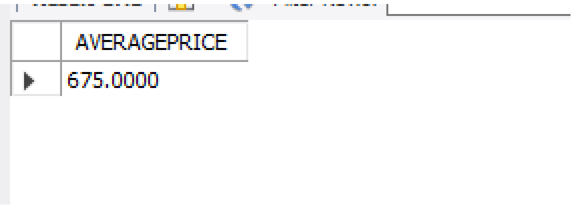
SELECT DISTINCT CONCAT(BOOK_TITLE, '_is_written_by_', AUTHOR) AS BOOK_WRITTEN_BY
FROM LMS_BOOK_DETAILS;



BOOK_WRITTEN_BY
Java How To Do Program_is_written_by_Paul J. Deitel
Java: The Complete Reference _is_written_by_Herbert Schildt
Let Us C_is_written_by_Yashavant Kanetkar

13. Average price of JAVA books

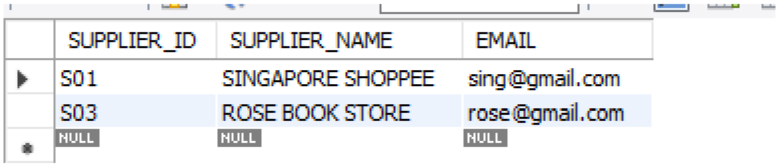
```
SELECT AVG(PRICE) AS AVERAGEPRICE
FROM LMS_BOOK_DETAILS
WHERE CATEGORY = 'JAVA';
```



AVERAGEPRICE
675.0000

14. Suppliers with gmail account

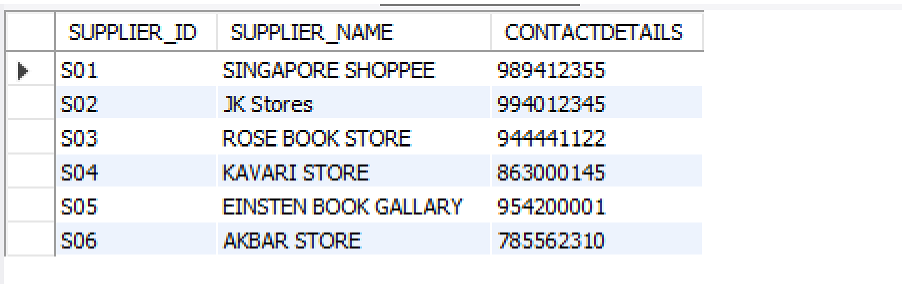
```
SELECT SUPPLIER_ID, SUPPLIER_NAME, EMAIL
FROM LMS_SUPPLIERS_DETAILS
WHERE EMAIL LIKE '%@gmail.com';
```



SUPPLIER_ID	SUPPLIER_NAME	EMAIL
S01	SINGAPORE SHOPPEE	sing@gmail.com
S03	ROSE BOOK STORE	rose@gmail.com
NULL	NULL	NULL

15. Supplier contact details using COALESCE

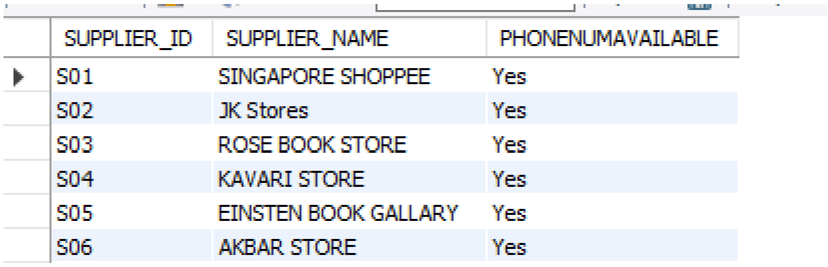
```
SELECT SUPPLIER_ID, SUPPLIER_NAME,
COALESCE(PHONE, EMAIL, ADDRESS)
AS CONTACTDETAILS
FROM LMS_SUPPLIERS_DETAILS;
```



SUPPLIER_ID	SUPPLIER_NAME	CONTACTDETAILS
S01	SINGAPORE SHOPPEE	989412355
S02	JK Stores	994012345
S03	ROSE BOOK STORE	944441122
S04	KAVARI STORE	863000145
S05	EINSTEN BOOK GALLARY	954200001
S06	AKBAR STORE	785562310

16. Supplier phone availability using IFNULL

```
SELECT SUPPLIER_ID, SUPPLIER_NAME,
IF(PHONE IS NULL, 'No', 'Yes')
AS PHONENUMAVAILABLE
FROM LMS_SUPPLIERS_DETAILS;
```

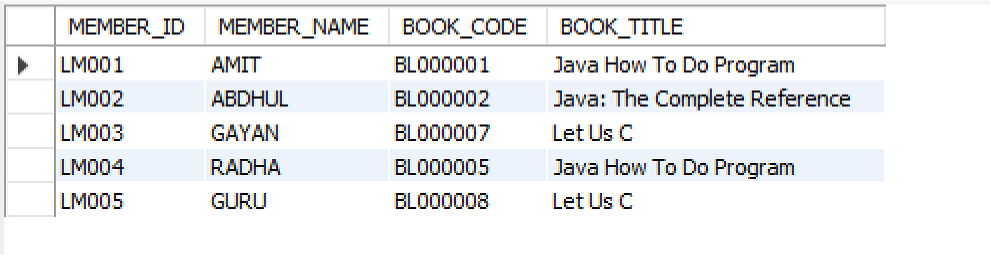


SUPPLIER_ID	SUPPLIER_NAME	PHONENUMAVAILABLE
S01	SINGAPORE SHOPPEE	Yes
S02	JK Stores	Yes
S03	ROSE BOOK STORE	Yes
S04	KAVARI STORE	Yes
S05	EINSTEN BOOK GALLARY	Yes
S06	AKBAR STORE	Yes

AVERAGE PROBLEMS

1. **Members with books they have taken**

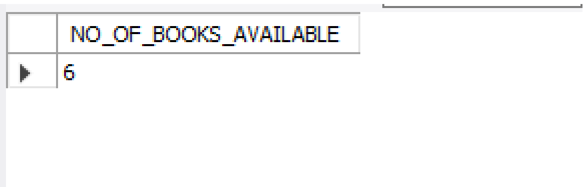
```
SELECT M.MEMBER_ID, M.MEMBER_NAME, B.BOOK_CODE, D.BOOK_TITLE
FROM LMS_MEMBERS M
JOIN LMS_BOOK_ISSUE B ON M.MEMBER_ID = B.MEMBER_ID
JOIN LMS_BOOK_DETAILS D ON B.BOOK_CODE = D.BOOK_CODE;
```



	MEMBER_ID	MEMBER_NAME	BOOK_CODE	BOOK_TITLE
▶	LM001	AMIT	BL000001	Java How To Do Program
	LM002	ABDHUL	BL000002	Java: The Complete Reference
	LM003	GAYAN	BL000007	Let Us C
	LM004	RADHA	BL000005	Java How To Do Program
	LM005	GURU	BL000008	Let Us C

2. **Total number of books available**

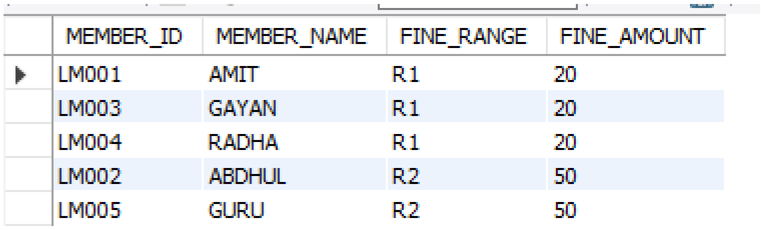
```
SELECT COUNT(*) AS NO_OF_BOOKS_AVAILABLE
FROM LMS_BOOK_DETAILS
WHERE BOOK_CODE NOT IN (
  SELECT BOOK_CODE
  FROM LMS_BOOK_ISSUE
  WHERE BOOK_ISSUE_STATUS = 'Y'
);
```



	NO_OF_BOOKS_AVAILABLE
▶	6

3. **Members with fine amount less than 100**

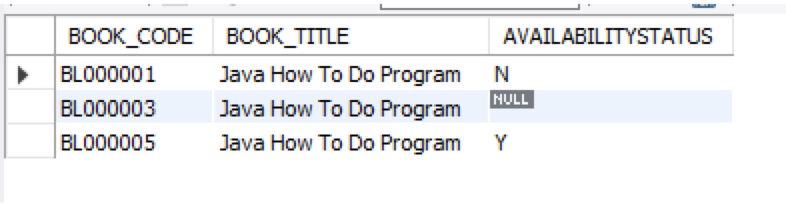
```
SELECT M.MEMBER_ID, M.MEMBER_NAME, F.FINE_RANGE, F.FINE_AMOUNT
FROM LMS_BOOK_ISSUE B
JOIN LMS_MEMBERS M ON B.MEMBER_ID = M.MEMBER_ID
JOIN LMS_FINE_DETAILS F ON B.FINE_RANGE = F.FINE_RANGE
WHERE F.FINE_AMOUNT < 100;
```



MEMBER_ID	MEMBER_NAME	FINE_RANGE	FINE_AMOUNT
LM001	AMIT	R1	20
LM003	GAYAN	R1	20
LM004	RADHA	R1	20
LM002	ABDHUL	R2	50
LM005	GURU	R2	50

4. **JAVA books edition 6 with availability**

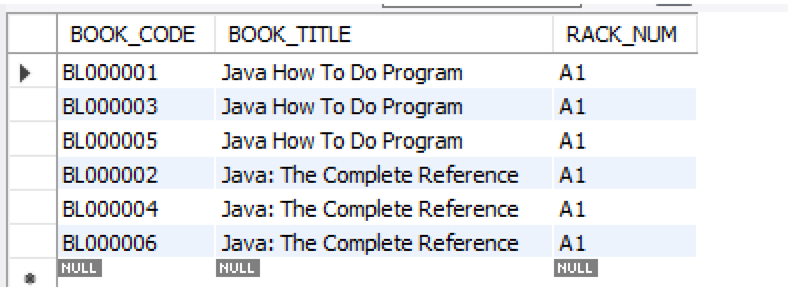
    SELECT D.BOOK_CODE, D.BOOK_TITLE, B.BOOK_ISSUE_STATUS AS AVAILABILITYSTATUS
    FROM LMS_BOOK_DETAILS D
    LEFT JOIN LMS_BOOK_ISSUE B ON D.BOOK_CODE = B.BOOK_CODE
    WHERE D.CATEGORY = 'JAVA' AND D.BOOK_EDITION = 6;



BOOK_CODE	BOOK_TITLE	AVAILABILITYSTATUS
BL000001	Java How To Do Program	N
BL000003	Java How To Do Program	NULL
BL000005	Java How To Do Program	Y

5. **Books in rack A1 sorted by title**

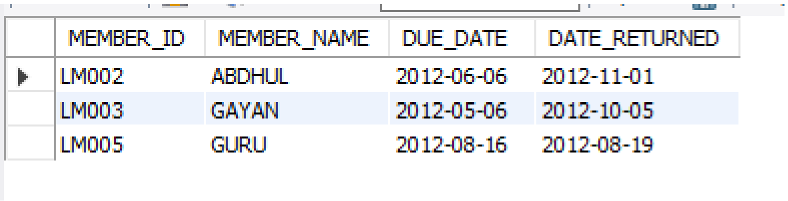
    SELECT BOOK_CODE, BOOK_TITLE, RACK_NUM
    FROM LMS_BOOK_DETAILS
    WHERE RACK_NUM = 'A1'
    ORDER BY BOOK_TITLE ASC;



BOOK_CODE	BOOK_TITLE	RACK_NUM
BL000001	Java How To Do Program	A1
BL000003	Java How To Do Program	A1
BL000005	Java How To Do Program	A1
BL000002	Java: The Complete Reference	A1
BL000004	Java: The Complete Reference	A1
BL000006	Java: The Complete Reference	A1
NULL	NULL	NULL

6. Members who returned books after due date

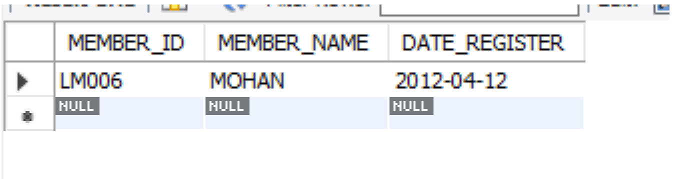
```
SELECT M.MEMBER_ID, M.MEMBER_NAME, B.DATE_RETURN
AS DUE_DATE, B.DATE_RETURNED
FROM LMS_BOOK_ISSUE B
JOIN LMS_MEMBERS M ON B.MEMBER_ID = M.MEMBER_ID
WHERE B.DATE_RETURNED > B.DATE_RETURN;
```



MEMBER_ID	MEMBER_NAME	DUE_DATE	DATE_RETURNED
LM002	ABDHUL	2012-06-06	2012-11-01
LM003	GAYAN	2012-05-06	2012-10-05
LM005	GURU	2012-08-16	2012-08-19

7. Members who have not taken any book

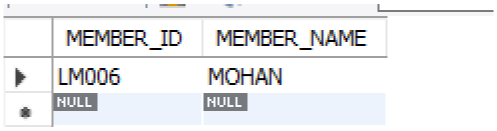
```
SELECT MEMBER_ID, MEMBER_NAME, DATE_REGISTER
FROM LMS_MEMBERS
WHERE MEMBER_ID NOT IN (
SELECT MEMBER_ID FROM LMS_BOOK_ISSUE);
```



MEMBER_ID	MEMBER_NAME	DATE_REGISTER
LM006	MOHAN	2012-04-12
NULL	NULL	NULL

8. Members who paid no fine in 2012

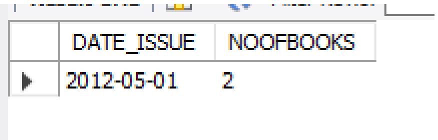
```
SELECT M.MEMBER_ID, M.MEMBER_NAME
FROM LMS_MEMBERS M
WHERE M.MEMBER_ID NOT IN (
  SELECT B.MEMBER_ID
  FROM LMS_BOOK_ISSUE B
  WHERE YEAR(B.DATE_ISSUE) = 2012
);
```



MEMBER_ID	MEMBER_NAME
LM006	MOHAN
NULL	NULL

9. Date with maximum books issued

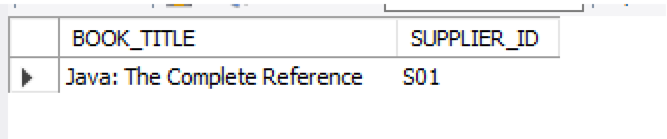
```
SELECT DATE_ISSUE, COUNT(*) AS NOOFBOOKS
FROM LMS_BOOK_ISSUE
GROUP BY DATE_ISSUE
ORDER BY NOOFBOOKS DESC
LIMIT 1;
```



	DATE_ISSUE	NOOFBOOKS
▶	2012-05-01	2

10. Books by Herbert Schildt, edition 5, supplier S01

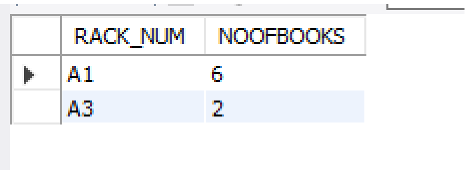
```
SELECT BOOK_TITLE, SUPPLIER_ID
FROM LMS_BOOK_DETAILS
WHERE AUTHOR = 'Herbert Schildt'
  AND BOOK_EDITION = 5
  AND SUPPLIER_ID = 'S01';
```



	BOOK_TITLE	SUPPLIER_ID
▶	Java: The Complete Reference	S01

11. Number of books in each rack

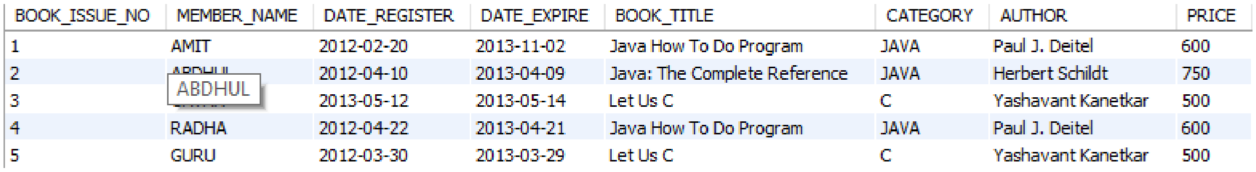
```
SELECT RACK_NUM, COUNT(*) AS NOOFBOOKS
FROM LMS_BOOK_DETAILS
GROUP BY RACK_NUM
ORDER BY RACK_NUM ASC;
```



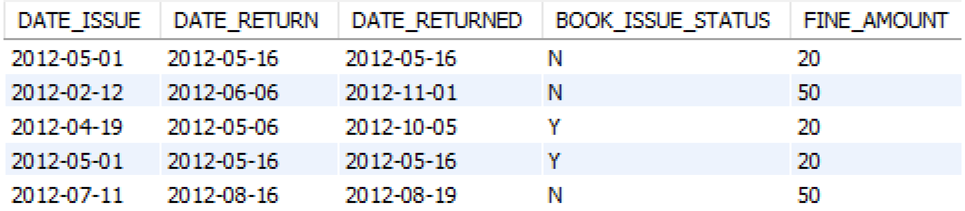
	RACK_NUM	NOOFBOOKS
▶	A1	6
	A3	2

12. Full book issue details

```
SELECT B.BOOK_ISSUE_NO, M.MEMBER_NAME, M.DATE_REGISTER,
M.DATE_EXPIRE,
D.BOOK_TITLE, D.CATEGORY, D.AUTHOR, D.PRICE,
 B.DATE_ISSUE, B.DATE_RETURN, B.DATE_RETURNED,
 B.BOOK_ISSUE_STATUS, F.FINE_AMOUNT
FROM LMS_BOOK_ISSUE B
JOIN LMS_MEMBERS M ON B.MEMBER_ID = M.MEMBER_ID
JOIN LMS_BOOK_DETAILS D ON B.BOOK_CODE = D.BOOK_CODE
JOIN LMS_FINE_DETAILS F ON B.FINE_RANGE = F.FINE_RANGE;
```



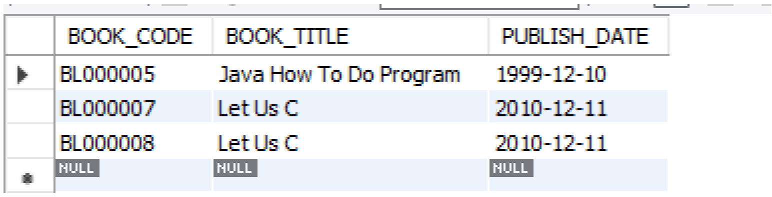
BOOK_ISSUE_NO	MEMBER_NAME	DATE_REGISTER	DATE_EXPIRE	BOOK_TITLE	CATEGORY	AUTHOR	PRICE
1	AMIT	2012-02-20	2013-11-02	Java How To Do Program	JAVA	Paul J. Deitel	600
2	ABDHUL	2012-04-10	2013-04-09	Java: The Complete Reference	JAVA	Herbert Schildt	750
3	[illegible]	2013-05-12	2013-05-14	Let Us C	C	Yashavant Kanetkar	500
4	RADHA	2012-04-22	2013-04-21	Java How To Do Program	JAVA	Paul J. Deitel	600
5	GURU	2012-03-30	2013-03-29	Let Us C	C	Yashavant Kanetkar	500



DATE_ISSUE	DATE_RETURN	DATE_RETURNED	BOOK_ISSUE_STATUS	FINE_AMOUNT
2012-05-01	2012-05-16	2012-05-16	N	20
2012-02-12	2012-06-06	2012-11-01	N	50
2012-04-19	2012-05-06	2012-10-05	Y	20
2012-05-01	2012-05-16	2012-05-16	Y	20
2012-07-11	2012-08-16	2012-08-19	N	50

13. Books published in December

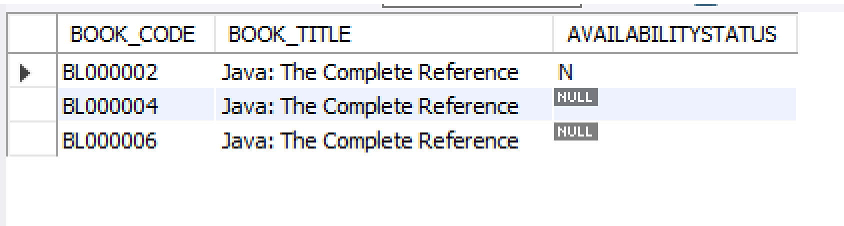
```
SELECT BOOK_CODE, BOOK_TITLE, PUBLISH_DATE
FROM LMS_BOOK_DETAILS
WHERE MONTH(PUBLISH_DATE) = 12;
```



	BOOK_CODE	BOOK_TITLE	PUBLISH_DATE
▶	BL000005	Java How To Do Program	1999-12-10
	BL000007	Let Us C	2010-12-11
	BL000008	Let Us C	2010-12-11
*	NULL	NULL	NULL

14. JAVA books edition 5 with availability

```
SELECT D.BOOK_CODE, D.BOOK_TITLE, B.BOOK_ISSUE_STATUS AS
AVAILABILITYSTATUS
FROM LMS_BOOK_DETAILS D
LEFT JOIN LMS_BOOK_ISSUE B ON D.BOOK_CODE = B.BOOK_CODE
WHERE D.CATEGORY = 'JAVA' AND D.BOOK_EDITION = 5;
```

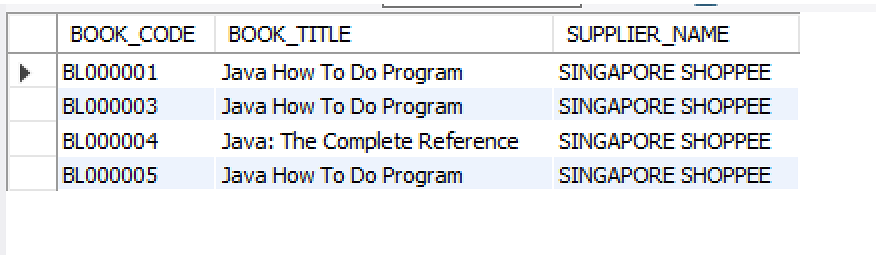


	BOOK_CODE	BOOK_TITLE	AVAILABILITYSTATUS
▶	BL000002	Java: The Complete Reference	N
	BL000004	Java: The Complete Reference	NULL
	BL000006	Java: The Complete Reference	NULL

COMPLEX PROBLEMS

1. Books from supplier with maximum supply

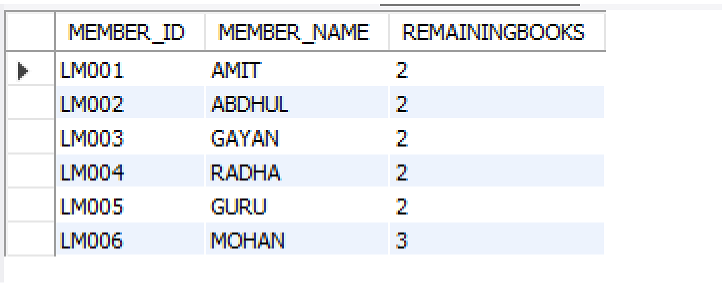
```
SELECT BD.BOOK_CODE, BD.BOOK_TITLE, SD.SUPPLIER_NAME
FROM LMS_BOOK_DETAILS BD
JOIN LMS_SUPPLIERS_DETAILS SD ON BD.SUPPLIER_ID = SD.SUPPLIER_ID
WHERE BD.SUPPLIER_ID = (
SELECT SUPPLIER_ID
FROM (SELECT SUPPLIER_ID, COUNT(*) AS book_count FROM
LMS_BOOK_DETAILS GROUP BY SUPPLIER_ID
ORDER BY book_count DESC LIMIT 1 ) AS MaxSupplier );
```



	BOOK_CODE	BOOK_TITLE	SUPPLIER_NAME
▶	BL000001	Java How To Do Program	SINGAPORE SHOPPEE
	BL000003	Java How To Do Program	SINGAPORE SHOPPEE
	BL000004	Java: The Complete Reference	SINGAPORE SHOPPEE
	BL000005	Java How To Do Program	SINGAPORE SHOPPEE

2. **Members with remaining books they can take**

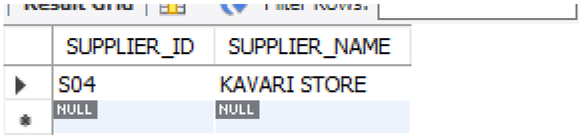
```
SELECT M.MEMBER_ID, M.MEMBER_NAME,
(3 - IFNULL(B.taken_count, 0)) AS REMAININGBOOKS
FROM LMS_MEMBERS M
LEFT JOIN (
  SELECT MEMBER_ID, COUNT(*) AS taken_count
  FROM LMS_BOOK_ISSUE
  GROUP BY MEMBER_ID
) B ON M.MEMBER_ID = B.MEMBER_ID;
```



MEMBER_ID	MEMBER_NAME	REMAININGBOOKS
LM001	AMIT	2
LM002	ABDHUL	2
LM003	GAYAN	2
LM004	RADHA	2
LM005	GURU	2
LM006	MOHAN	3

3. **Supplier with minimum books supplied**

```
SELECT SD.SUPPLIER_ID, SD.SUPPLIER_NAME
FROM LMS_SUPPLIERS_DETAILS SD
WHERE SD.SUPPLIER_ID = (
SELECT SUPPLIER_ID
FROM ( SELECT SUPPLIER_ID, COUNT(*) AS book_count
FROM LMS_BOOK_DETAILS
GROUP BY SUPPLIER_ID
ORDER BY book_count ASC LIMIT 1) AS MinSupplier );
```



SUPPLIER_ID	SUPPLIER_NAME
S04	KAVARI STORE
NULL	NULL