

A SQL- BASED DATA ANALYSIS PROJECT

LIABLARY MANAGEMENT SYSTEM

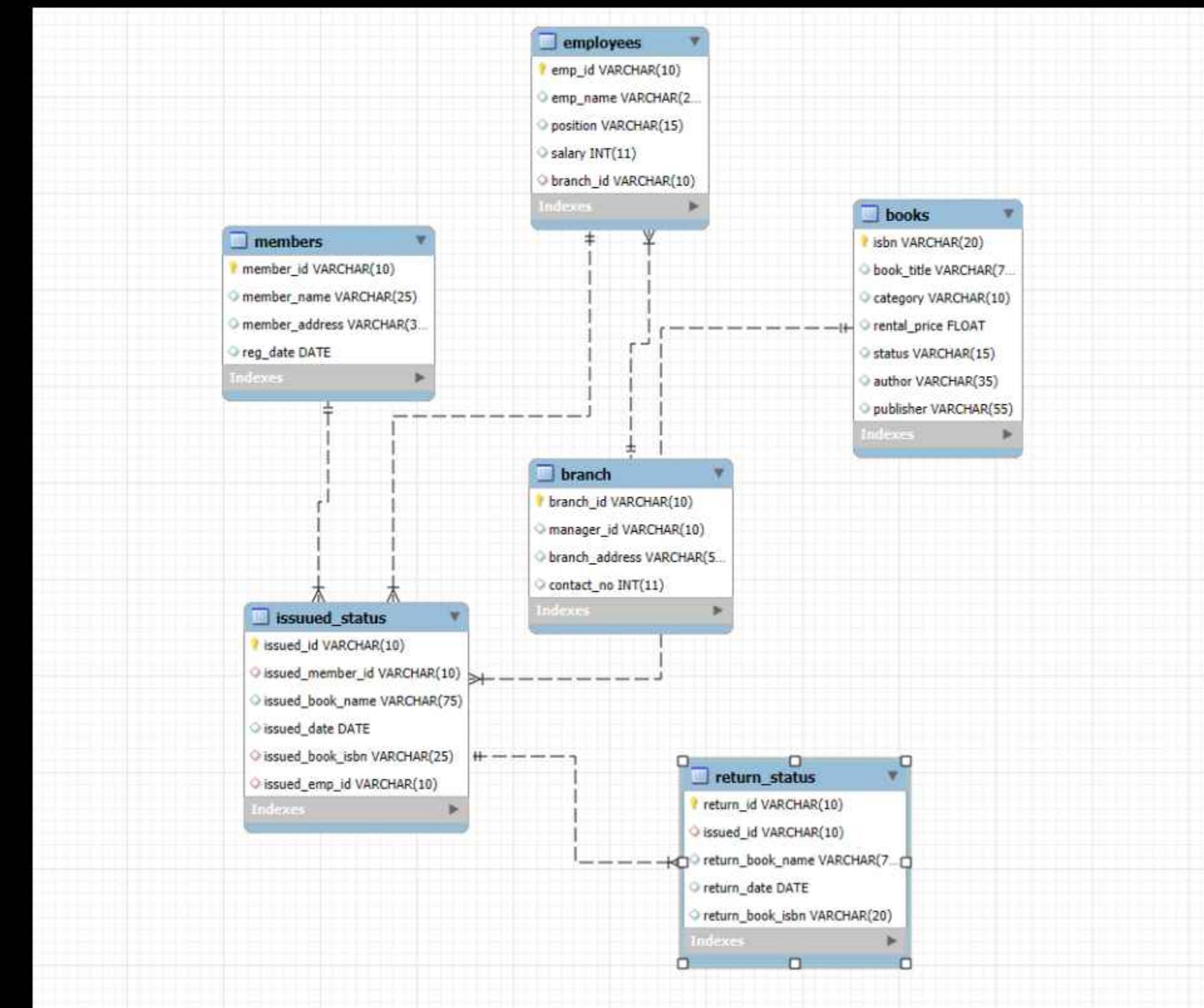
PRESENTATION BY
KRUTIKA JAMKHANDI

OBJECTIVE

The objective of this Library Management System project is to design and implement a structured SQL database that efficiently stores, manages, and retrieves library-related information, including books, members, employees, and issued/returned records. The project aims to handle real-world operations such as issuing books, tracking returns, updating records, and generating useful analytical reports using SQL queries, joins, subqueries, and window functions to support decision-making and improve library management efficiency.



ER- DIAGRAM

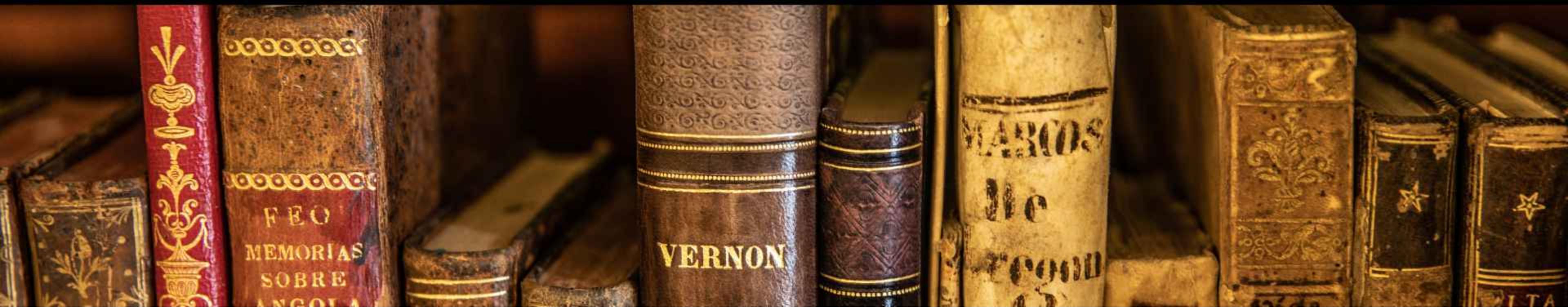


STRUCTURE OF TABLE

```
327 • describe books;
```

Field	Type	Null	Key	Default	Extra
isbn	varchar(20)	NO	PRI	NULL	
book_title	varchar(75)	YES		NULL	
category	varchar(50)	YES		NULL	
rental_price	float	YES		NULL	
status	varchar(15)	YES		NULL	
author	varchar(35)	YES		NULL	
publisher	varchar(55)	YES		NULL	

THE BOOKS TABLE STORES ALL ESSENTIAL INFORMATION ABOUT THE BOOKS AVAILABLE IN THE LIBRARY. EACH ROW REPRESENTS ONE BOOK, AND THE TABLE HELPS MANAGE BOOK DETAILS SUCH AS TITLE, CATEGORY, PRICE, AVAILABILITY, AUTHOR NAME, AND PUBLISHER INFORMATION.

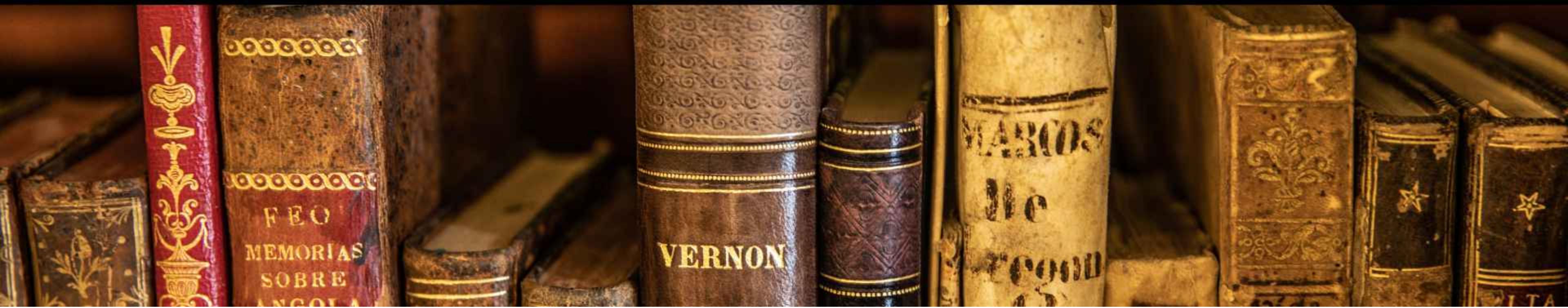


STRUCTURE OF TABLE

328 • `describe branch;`

THE BRANCH TABLE STORES INFORMATION ABOUT EACH LIBRARY BRANCH. EVERY ROW REPRESENTS ONE
BRANCH LOCATION AND INCLUDES DETAILS SUCH AS THE BRANCH ID, MANAGER RESPONSIBLE, ADDRESS,
AND CONTACT NUMBER.

	Field	Type	Null	Key	Default	Extra
▶	branch_id	varchar(10)	NO	PRI	NULL	
	manager_id	varchar(10)	YES		NULL	
	branch_address	varchar(50)	YES		NULL	
	contact_no	int(11)	YES		NULL	

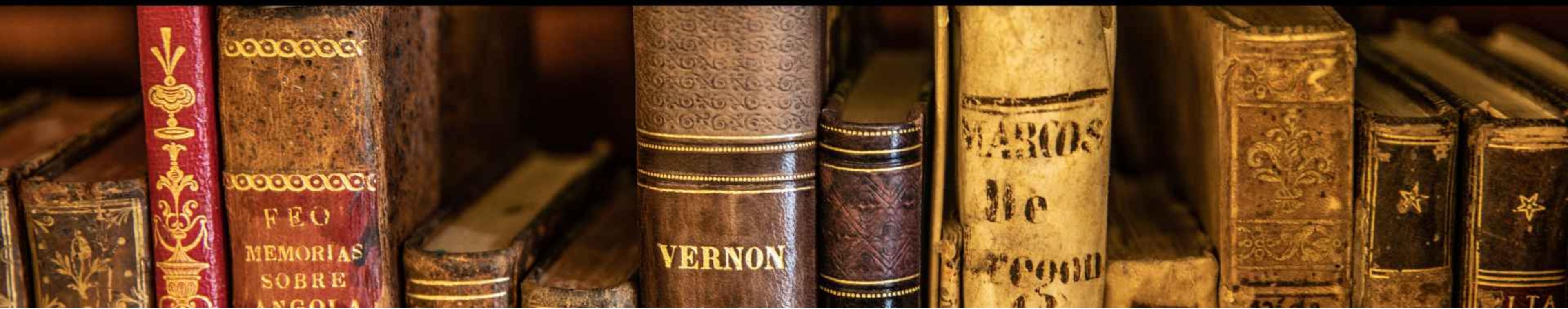


STRUCTURE OF TABLE

```
329 • describe employees;
```

	Field	Type	Null	Key	Default	Extra
▶	emp_id	varchar(10)	NO	PRI	NULL	
	emp_name	varchar(25)	YES		NULL	
	position	varchar(15)	YES		NULL	
	salary	int(11)	YES		NULL	
	branch_id	varchar(10)	YES	MUL	NULL	

THE EMPLOYEES TABLE STORES INFORMATION ABOUT ALL STAFF MEMBERS WORKING IN THE LIBRARY. EACH RECORD REPRESENTS AN EMPLOYEE AND INCLUDES PERSONAL DETAILS, JOB ROLE, SALARY, AND THE BRANCH WHERE THEY ARE ASSIGNED.

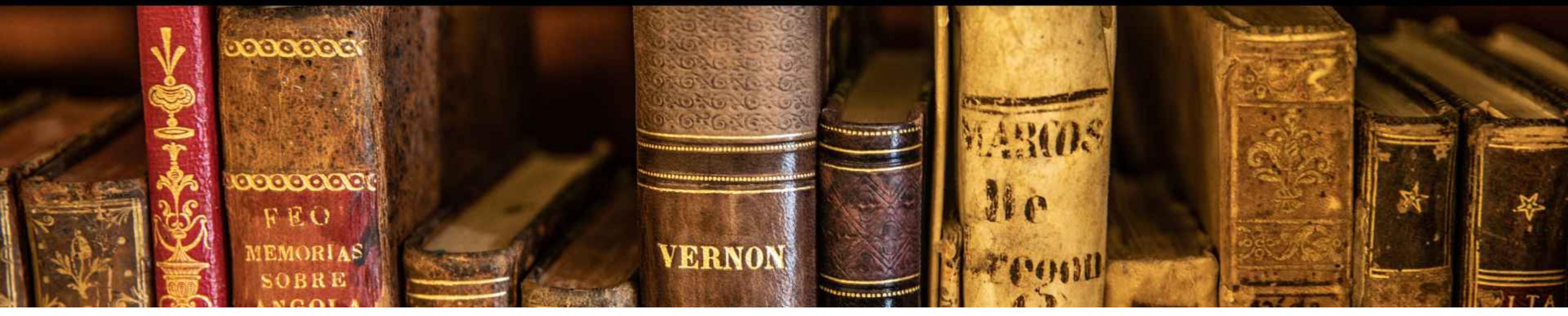


STRUCTURE OF TABLE

```
330 • describe issued_status;
```

Field	Type	Null	Key	Default	Extra
issued_id	varchar(10)	NO	PRI	NULL	
issued_member_id	varchar(10)	YES	MUL	NULL	
issued_book_name	varchar(75)	YES		NULL	
issued_date	date	YES		NULL	
issued_book_isbn	varchar(25)	YES	MUL	NULL	
issued_emp_id	varchar(10)	YES	MUL	NULL	

THE ISSUED_STATUS TABLE KEEPS TRACK OF BOOKS THAT HAVE BEEN ISSUED TO LIBRARY MEMBERS. EACH RECORD REPRESENTS ONE BOOK ISSUED EVENT AND STORES INFORMATION ABOUT WHO BORROWED THE BOOK, WHICH BOOK WAS ISSUED, WHEN IT WAS ISSUED, AND WHICH EMPLOYEE PROCESSED THE TRANSACTION.

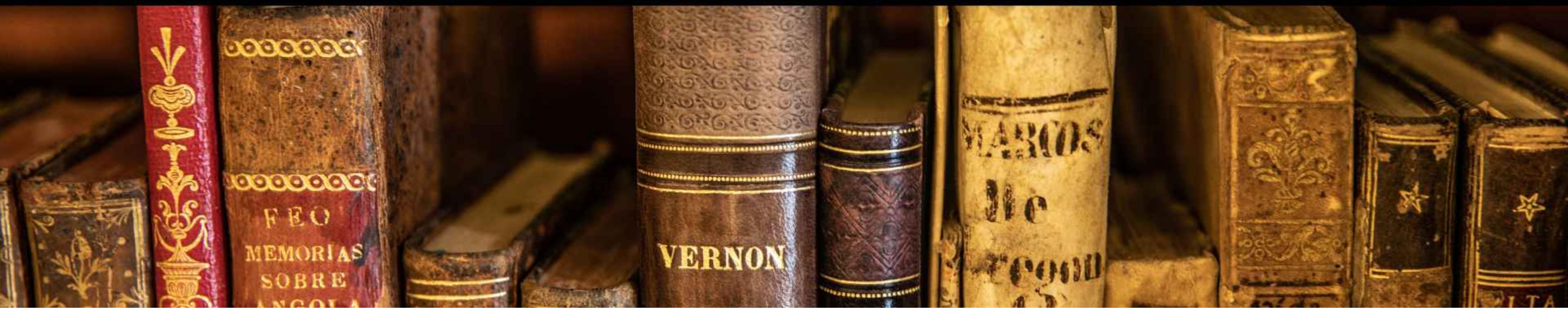


STRUCTURE OF TABLE

```
331 • describe members;
```

	Field	Type	Null	Key	Default	Extra
▶	member_id	varchar(10)	NO	PRI	NULL	
	member_name	varchar(25)	YES		NULL	
	member_address	varchar(30)	YES		NULL	
	reg_date	date	YES		NULL	

THE MEMBERS TABLE STORES INFORMATION ABOUT INDIVIDUALS REGISTERED IN THE LIBRARY. EACH ROW REPRESENTS ONE MEMBER AND INCLUDES IDENTIFYING AND CONTACT DETAILS ALONG WITH THE DATE THEY JOINED THE LIBRARY.

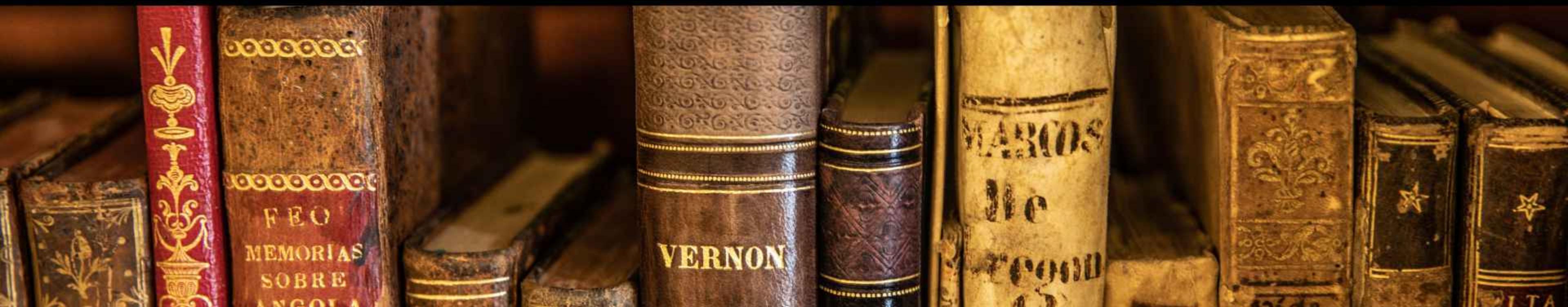


STRUCTURE OF TABLE

332 • `describe return_status;`

Field	Type	Null	Key	Default	Extra
return_id	varchar(10)	NO	PRI	NULL	
issued_id	varchar(10)	YES	MUL	NULL	
return_book_name	varchar(75)	YES		NULL	
return_date	date	YES		NULL	
return_book_isbn	varchar(20)	YES		NULL	

THE RETURN_STATUS TABLE TRACKS THE DETAILS OF BOOKS THAT HAVE BEEN RETURNED TO THE LIBRARY AFTER BEING ISSUED. EACH RECORD CORRESPONDS TO A RETURN TRANSACTION AND LINKS THE RETURN ENTRY TO THE PREVIOUSLY ISSUED RECORD.



CONTENTS OF TABLE

isbn	book_title	category	rental_price	status	author	publisher
978-0-06-025492-6	Where the Wild Things Are	Children	3.5	yes	Maurice Sendak	HarperCollins
978-0-06-112008-4	To Kill a Mockingbird	Classic	5	yes	Harper Lee	J.B. Lippincott & Co.
978-0-06-112241-5	The Kite Runner	Fiction	5.5	yes	Khaled Hosseini	Riverhead Books
978-0-06-440055-8	Charlotte's Web	Children	4	yes	E.B. White	Harper & Row
978-0-09-957807-9	A Game of Thrones	Fantasy	7.5	yes	George R.R. Martin	Bantam
978-0-14-027526-3	A Tale of Two Cities	Classic	4.5	yes	Charles Dickens	Penguin Books
978-0-14-044930-3	The Histories	History	5.5	yes	Herodotus	Penguin Classics
978-0-14-118776-1	One Hundred Years of Solitude	Literary F	6.5	yes	Gabriel Garcia Marquez	Penguin Books
978-0-14-143951-8	Pride and Prejudice	Classic	5	yes	Jane Austen	Penguin Classics
978-0-141-44171-6	Jane Eyre	Classic	4	yes	Charlotte Bronte	Penguin Classics
978-0-19-280551-1	The Guns of August	History	7	yes	Barbara W. Tuchman	Oxford University ...
978-0-307-37840-1	The Alchemist	Fiction	2.5	yes	Paulo Coelho	HarperOne
978-0-307-58837-1	Sapiens: A Brief History of Hu...	History	8	no	Yuval Noah Harari	Harper Perennial
978-0-330-25864-8	Animal Farm	Classic	5.5	yes	George Orwell	Penguin Books
978-0-345-39180-3	Dune	Science Fi	8.5	yes	Frank Herbert	Ace
978-0-375-41398-8	The Diary of a Young Girl	History	6.5	no	Anne Frank	Bantam
978-0-375-50167-0	The Road	Dystopian	7	yes	Cormac McCarthy	Vintage
978-0-385-33312-0	The Shining	Horror	6	yes	Stephen King	Doubleday
978-0-393-05081-8	A Peoples History of the Unite...	History	9	yes	Howard Zinn	Harper Perennial
978-0-393-91257-8	Guns, Germs, and Steel: The ...	History	7	yes	Jared Diamond	W. W. Norton & C...
978-0-451-52993-5	Fahrenheit 451	Dystopian	5.5	yes	Ray Bradbury	Ballantine Books
978-0-451-52994-2	Moby Dick	Classic	6.5	yes	Herman Melville	Penguin Books
978-0-452-28240-7	Brave New World	Dystopian	6.5	yes	Aldous Huxley	Harper Perennial
978-0-525-47535-5	The Great Gatsby	Classic	8	yes	F. Scott Fitzgerald	Scribner
978-0-553-29698-2	The Catcher in the Rye	Classic	7	yes	J.D. Salinger	Little, Brown and ...
978-0-553-57340-1	1984	Dystopian	6.5	yes	George Orwell	Penguin Books
978-0-670-81302-4	The Road	Dystopian	7	yes	Cormac McCarthy	Knopf



CONTENTS OF TABLE

Result Grid | Filter Rows: | Edit:

branch_id	manager_id	branch_address	contact_no
B001	E109	123 Main St	2147483647
B002	E109	456 Elm St	2147483647
B003	E109	789 Oak St	2147483647
B004	E110	567 Pine St	2147483647
B005	E110	890 Maple St	2147483647



CONTENTS OF TABLE

Result Grid | Filter Rows: | Edit:

	emp_id	emp_name	position	salary	branch_id
▶	E101	John Doe	Clerk	60000	B001
	E102	Jane Smith	Clerk	45000	B002
	E103	Mike Johnson	Librarian	55000	B001
	E104	Emily Davis	Assistant	40000	B001
	E105	Sarah Brown	Assistant	42000	B001
	E106	Michelle Ramirez	Assistant	43000	B001
	E107	Michael Thompson	Clerk	62000	B005
	E108	Jessica Taylor	Clerk	46000	B004
	E109	Daniel Anderson	Manager	57000	B003
	E110	Laura Martinez	Manager	41000	B005
	E111	Christopher Lee	Assistant	65000	B005
	HULL	HULL	HULL	HULL	HULL



CONTENTS OF TABLE

337 • `select * from issued_status;`

issued_id	issued_member_id	issued_book_name	issued_date	issued_book_isbn	issued_emp_id
IS106	C106	Animal Farm	2024-03-10	978-0-330-25864-8	E104
IS107	C107	One Hundred Years of Solitude	2024-03-11	978-0-14-118776-1	E104
IS108	C108	The Great Gatsby	2024-03-12	978-0-525-47535-5	E104
IS109	C109	Jane Eyre	2024-03-13	978-0-141-44171-6	E105
IS110	C110	The Alchemist	2024-03-14	978-0-307-37840-1	E105
IS111	C109	Harry Potter and the Sorcerer's Stone	2024-03-15	978-0-679-76489-8	E105
IS112	C109	A Game of Thrones	2024-03-16	978-0-09-957807-9	E106
IS113	C109	A People's History of the United States	2024-03-17	978-0-393-05081-8	E106
IS114	C109	The Guns of August	2024-03-18	978-0-19-280551-1	E106
IS115	C109	The Histories	2024-03-19	978-0-14-044930-3	E107
IS116	C110	Guns, Germs, and Steel: The Fates of...	2024-03-20	978-0-393-91257-8	E107
IS117	C110	1984	2024-03-21	978-0-679-64115-3	E107
IS118	C101	Pride and Prejudice	2024-03-22	978-0-14-143951-8	E108
IS119	C110	Brave New World	2024-03-23	978-0-452-28240-7	E108
IS120	C110	The Road	2024-03-24	978-0-670-81302-4	E108
IS122	C102	Fahrenheit 451	2024-03-26	978-0-451-52993-5	E109
IS123	C103	Dune	2024-03-27	978-0-345-39180-3	E109
IS124	C104	Where the Wild Things Are	2024-03-28	978-0-06-025492-6	E110
IS125	C105	The Kite Runner	2024-03-29	978-0-06-112241-5	E110
IS126	C105	Charlotte's Web	2024-03-30	978-0-06-440055-8	E110
IS127	C105	Beloved	2024-03-31	978-0-679-77644-3	E110
IS128	C105	A Tale of Two Cities	2024-04-01	978-0-14-027526-3	E110
IS129	C105	The Stand	2024-04-02	978-0-7434-7679-3	E110
IS130	C106	Moby Dick	2024-04-03	978-0-451-52994-2	E101
IS131	C106	To Kill a Mockingbird	2024-04-04	978-0-06-112008-4	E101
IS132	C106	The Hobbit	2024-04-05	978-0-7432-7356-4	E106
IS133	C107	Angels & Demons	2024-04-06	978-0-7432-4722-5	E106
IS134	C107	The Diary of a Young Girl	2024-04-07	978-0-375-41398-8	E106
IS135	C107	Sapiens: A Brief History of Humankind	2024-04-08	978-0-307-58837-1	E108
IS136	C107	1491: New Revelations of the Americ...	2024-04-09	978-0-7432-7357-1	E102



CONTENTS OF TABLE

338 • `select * from members;`

	member_id	member_name	member_address	reg_date
▶	C101	Alice Johnson	125 Main St	2021-05-15
	C102	Bob Smith	456 Elm St	2021-06-20
	C103	Carol Davis	789 Oak St	2021-07-10
	C104	Dave Wilson	567 Pine St	2021-08-05
	C105	Eve Brown	890 Maple St	2021-09-25
	C106	Frank Thomas	234 Cedar St	2021-10-15
	C107	Grace Taylor	345 Walnut St	2021-11-20
	C108	Henry Anderson	456 Birch St	2021-12-10
	C109	Ivy Martinez	567 Oak St	2022-01-05
	C110	Jack Wilson	678 Pine St	2022-02-25
	C118	Sam	133 Pine St	2024-06-01
	C119	John	143 Main St	2024-05-01



CONTENTS OF TABLE

339 • `select * from return_status;`

	return_id	issued_id	return_book_name	return_date	return_book_isbn
▶	RS105	IS107	NULL	2024-05-03	NULL
	RS106	IS108	NULL	2024-05-05	NULL
	RS107	IS109	NULL	2024-05-07	NULL
	RS108	IS110	NULL	2024-05-09	NULL
	RS109	IS111	NULL	2024-05-11	NULL
	RS110	IS112	NULL	2024-05-13	NULL
	RS111	IS113	NULL	2024-05-15	NULL
	RS112	IS114	NULL	2024-05-17	NULL
	RS113	IS115	NULL	2024-05-19	NULL
	RS114	IS116	NULL	2024-05-21	NULL
	RS115	IS117	NULL	2024-05-23	NULL
	RS116	IS118	NULL	2024-05-25	NULL
	RS117	IS119	NULL	2024-05-27	NULL
	RS118	IS120	NULL	2024-05-29	NULL
*	NULL	NULL	NULL	NULL	NULL





CURD OPERATIONS

1. create: inserted sample records into the books table.
- 2.read: retrieved and displayed data from various tables.
- 3.update: updated records in the employees table.
- 4.delete: removed records from the members table as needed.

**TASK 1. CREATE A NEW BOOK RECORD -- "978-1-60129-456-2",
'TO KILL A MOCKINGBIRD', 'CLASSIC', 6.00, 'YES', 'HARPER
LEE', 'J.B. LIPPINCOTT & CO.'"**

```
226 • INSERT INTO books(isbn, book_title, category, rental_price, status, author, publisher)
227   VALUES
228   ('978-1-60129-456-2', 'To Kill a Mockingbird', 'Classic', 6.00, 'yes', 'Harper Lee', 'J.B. Lippincott & Co.');
229 • select * from books
```

978-1-60129-456-2	To Kill a Mockingbird	Classic	6	yes	Harper Lee	J.B. Lippincott & Co.
-------------------	-----------------------	---------	---	-----	------------	-----------------------

**INSIGHT:- A NEW BOOK ENTRY WAS ADDED SUCCESSFULLY, SHOWING
THAT THE DATABASE ACCEPTS NEW RECORDS AND SUPPORTS
UPDATING THE LIBRARY'S CATALOG WITH ESSENTIAL BOOK DETAILS.**



TASK 2: UPDATE AN EXISTING MEMBER'S ADDRESS

```
232 ✘ update members
233      set member_address = '125 Main St'
234      where member_id = 'C101';
```

	member_id	member_name	member_address	reg_date
▶	C101	Alice Johnson	125 Main St	2021-05-15

INSIGHT:- THE MEMBER'S ADDRESS WAS SUCCESSFULLY UPDATED, DEMONSTRATING THAT EXISTING RECORDS IN THE DATABASE CAN BE MODIFIED USING THE UPDATE STATEMENT.



TASK 3: DELETE A RECORD FROM THE ISSUED STATUS TABLE

-- OBJECTIVE: DELETE THE RECORD WITH ISSUED_ID = 'IS121' FROM THE ISSUED_STATUS TABLE.

```
239 • delete from issued_status  
240      where issued_id = 'IS121';
```

issued_id	issued_member_id	issued_book_name	issued_date	issued_book_isbn	issued_emp_id
IS118	C101	Pride and Prejudice	2024-03-22	978-0-14-143951-8	E108
IS119	C110	Brave New World	2024-03-23	978-0-452-28240-7	E108
IS120	C110	The Road	2024-03-24	978-0-670-81302-4	E108
IS122	C102	Fahrenheit 451	2024-03-26	978-0-451-52993-5	E109
IS123	C103	Dune	2024-03-27	978-0-345-39180-3	E109
IS121	C101	War and Peace	2024-03-22	978-0-26-225402-6	E110

INSIGHT:- A RECORD WAS SUCCESSFULLY REMOVED FROM THE ISSUED_STATUS TABLE, SHOWING THAT THE DATABASE SUPPORTS DELETING SPECIFIC ENTRIES USING THE DELETE STATEMENT.



TASK 4: RETRIEVE ALL BOOKS ISSUED BY A SPECIFIC EMPLOYEE

-- OBJECTIVE: SELECT ALL BOOKS ISSUED BY THE EMPLOYEE WITH
EMP_ID = 'E101'.

```
245 • select * from issued_status
```

```
246 where issued_emp_id = 'E101';
```

	issued_id	issued_member_id	issued_book_name	issued_date	issued_book_isbn	issued_emp_id
▶	IS130	C106	Moby Dick	2024-04-03	978-0-451-52994-2	E101
	IS131	C106	To Kill a Mockingbird	2024-04-04	978-0-06-112008-4	E101

INSIGHT:- THIS QUERY FILTERS THE ISSUED RECORDS BASED ON THE EMPLOYEE ID, ALLOWING US TO VIEW ONLY THE BOOKS PROCESSED BY A SPECIFIC STAFF MEMBER. IT SHOWS HOW CONDITIONAL FILTERING HELPS TRACK EMPLOYEE PERFORMANCE AND BOOK ISSUING HISTORY WITHIN THE SYSTEM.



A photograph of a woman with long brown hair, wearing a white blouse and dark trousers, standing in a library. She is reaching up with her right hand to pull a book off a tall, dark wood bookshelf. The shelves are filled with books, and the floor is made of light-colored tiles with a diamond pattern. The lighting is warm and focused on the woman and the bookshelf.

ANALYTICAL QUERIES

Group by :- GROUP BY is used in SQL to group rows that have the same values in a column so we can perform calculations like count, sum, or average on each group.

Instead of showing every row, GROUP BY groups similar data together and shows one result for each group.

GROUP BY helps organize data into categories so we can analyze each group separately

Having:- HAVING is used to filter the grouped results created by GROUP BY. It works like WHERE, but WHERE filters rows before grouping, while HAVING filters after grouping.

e.g:- Show only groups that meet certain conditions (like groups with more than one record).

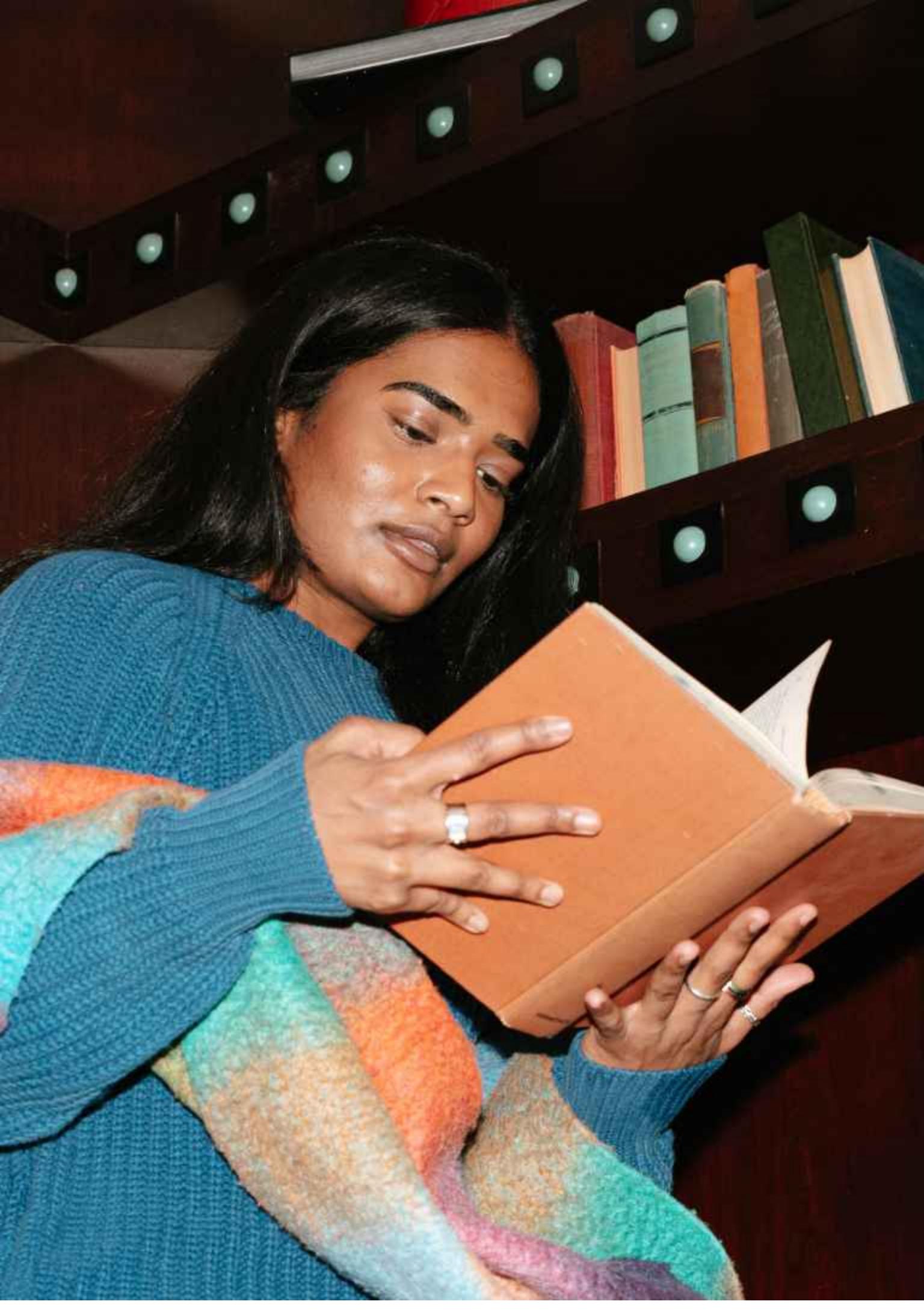
TASK 5: LIST MEMBERS WHO HAVE ISSUED MORE THAN ONE BOOK

```
250 •  select issued_emp_id,  
251      count(issued_id) as total_book_issued  
252      from issued_status  
253      group by issued_emp_id  
254      having total_book_issued > 1;
```

	issued_emp_id	total_book_issued
▶	E101	2
	E102	2
	E104	4
	E105	4
	E106	6
	E107	3
	E108	4
	E109	2
	E110	6

INSIGHT:- THIS QUERY NOT ONLY GROUPS ISSUING RECORDS BY EMPLOYEE BUT ALSO USES THE HAVING CLAUSE TO FILTER THE RESULTS SO THAT ONLY EMPLOYEES WHO ISSUED MORE THAN ONE BOOK ARE SHOWN. THIS MAKES IT USEFUL FOR MONITORING STAFF ACTIVITY, IDENTIFYING FREQUENT ISSUERS, AND ANALYZING WORKLOAD DISTRIBUTION WITHIN THE LIBRARY.





JOINS

A JOIN is used in SQL when you want to combine information from two or more tables.

Instead of storing all data in one huge table, databases store related data in separate tables.

A JOIN helps you connect those tables so you can see the related information in one result.

TASK 6: LIST EMPLOYEES WITH THEIR BRANCH MANAGER'S NAME AND THEIR BRANCH DETAILS:

```
272  -- List Employees with Their Branch Manager's Name and their branch details:
273  • SELECT
274      e1.emp_id,
275      e1.emp_name,
276      e1.position,
277      e1.salary,
278      b.*,
279      e2.emp_name as manager
280  FROM employees as e1
281  JOIN
282  branch as b
283  ON e1.branch_id = b.branch_id
284  JOIN
285  employees as e2
286  ON e2.emp_id = b.manager_id
```

emp_id	emp_name	position	salary	branch_id	manager_id	branch_address	contact_no	manager
E101	John Doe	Clerk	60000	B001	E109	123 Main St	2147483647	Daniel Anderson
E103	Mike Johnson	Librarian	55000	B001	E109	123 Main St	2147483647	Daniel Anderson
E104	Emily Davis	Assistant	40000	B001	E109	123 Main St	2147483647	Daniel Anderson
E105	Sarah Brown	Assistant	42000	B001	E109	123 Main St	2147483647	Daniel Anderson
E106	Michelle Ramirez	Assistant	43000	B001	E109	123 Main St	2147483647	Daniel Anderson
E102	Jane Smith	Clerk	45000	B002	E109	456 Elm St	2147483647	Daniel Anderson
E109	Daniel Anderson	Manager	57000	B003	E109	789 Oak St	2147483647	Daniel Anderson
E108	Jessica Taylor	Clerk	46000	B004	E110	567 Pine St	2147483647	Laura Martinez
E107	Michael Thompson	Clerk	62000	B005	E110	890 Maple St	2147483647	Laura Martinez
E110	Laura Martinez	Manager	41000	B005	E110	890 Maple St	2147483647	Laura Martinez
E111	Christopher Lee	Assistant	65000	B005	E110	890 Maple St	2147483647	Laura Martinez

INSIGHT:- THIS QUERY USES A SELF JOIN ON THE EMPLOYEES TABLE TO MATCH EACH EMPLOYEE WITH THEIR BRANCH MANAGER. BY JOINING THE BRANCH TABLE IN BETWEEN, IT ALSO RETRIEVES ADDITIONAL BRANCH INFORMATION SUCH AS BRANCH ID, ADDRESS, AND CONTACT NUMBER. THE FINAL RESULT CLEARLY SHOWS EACH EMPLOYEE ALONG WITH THEIR MANAGER'S NAME AND BRANCH DETAILS, MAKING IT EASIER TO UNDERSTAND THE REPORTING STRUCTURE AND BRANCH ASSIGNMENTS WITHIN THE ORGANIZATION.



TASK 7: FIND TOTAL RENTAL INCOME BY CATEGORY:

```
261 • SELECT
262     b.category,
263     SUM(b.rental_price)
264 FROM
265 issued_status as ist
266 JOIN
267 books as b
268 ON b.isbn = ist.issued_book_isbn
269 GROUP BY 1;
```

category	SUM(b.rental_price)
Children	7.5
Classic	59
Dystopian	25.5
Fantasy	28.5
Fiction	14.5
History	49.5
Horror	7
Literary F	6.5
Mystery	7.5
Science Fi	8.5

INSIGHT:- THIS QUERY SHOWS THE TOTAL RENTAL INCOME FOR EACH BOOK CATEGORY BY COMBINING DATA FROM TWO TABLES AND GROUPING THE RESULTS BY CATEGORY. THE OUTPUT HIGHLIGHTS WHICH CATEGORIES EARN MORE, WITH GENRES LIKE CLASSIC, FANTASY AND HISTORY GENERATING HIGHER INCOME COMPARED TO OTHERS.



TASK 8:RETRIEVE THE LIST OF BOOKS NOT YET RETURNED

```
select
distinct ist.issued_book_name
from issued_status as ist
left join
return_status as rs
on ist.issued_id = rs.issued_id
where rs.return_id is null;
select * from return_status;
```

issued_book_name
Animal Farm
Fahrenheit 451
Dune
Where the Wild Things Are
The Kite Runner
Charlotte's Web
Beloved
A Tale of Two Cities
The Stand
Moby Dick
To Kill a Mockingbird
The Hobbit
Angels & Demons
The Diary of a Young Girl
Sapiens: A Brief History o...
1491: New Revelations of...
The Catcher in the Rye
The Great Gatsby
Harry Potter and the Sorc...

INSIGHT:- THIS QUERY IDENTIFIES THE BOOKS THAT HAVE BEEN ISSUED BUT NOT YET RETURNED. BY USING A LEFT JOIN AND CHECKING FOR NULL RETURN RECORDS, IT FILTERS OUT ALL BOOKS THAT DO NOT HAVE A MATCHING RETURN ENTRY. THE RESULT SHOWS A LIST OF BOOKS THAT ARE STILL CURRENTLY ISSUED AND PENDING RETURN, WHICH HELPS IN TRACKING OVERDUE OR ACTIVE BORROWED ITEMS IN THE LIBRARY SYSTEM.





ADVANCED SQL

Windows function :-

SQL window functions allow performing calculations across a set of rows that are related to the current row, without collapsing the result into a single value. They are commonly used for tasks like aggregates, rankings and running totals.

The OVER clause defines the “window” of rows for the calculation. It can:

- PARTITION BY: divide the data into groups.

ORDER BY: specify the order of rows within each group.



TASK 9: RANK EMPLOYEES BY SALARY WITHIN EACH BRANCH

```
299 • SELECT
300     emp_id,
301     emp_name,
302     branch_id,
303     salary,
304     RANK() OVER (PARTITION BY branch_id ORDER BY salary DESC) AS salary_rank
305 FROM employees;
```

	emp_id	emp_name	branch_id	salary	salary_rank
▶	E101	John Doe	B001	60000	1
	E103	Mike Johnson	B001	55000	2
	E106	Michelle Ramirez	B001	43000	3
	E105	Sarah Brown	B001	42000	4
	E104	Emily Davis	B001	40000	5
	E102	Jane Smith	B002	45000	1
	E109	Daniel Anderson	B003	57000	1
	E108	Jessica Taylor	B004	46000	1
	E111	Christopher Lee	B005	65000	1
	E107	Michael Thompson	B005	62000	2
	E110	Laura Martinez	B005	41000	3

Insight :- This query ranks employees within their respective branches based on salary using a window function. The highest-paid employee in each branch receives rank 1, and employees with the same salary share the same rank. This ranking helps identify top earners per branch and makes it easier to compare salaries within the same location rather than across the entire organization.

SUBQUERY

A SUBQUERY (ALSO CALLED AN INNER QUERY OR NESTED QUERY) IS A SQL QUERY PLACED INSIDE ANOTHER QUERY (SUCH AS SELECT, INSERT, UPDATE, OR DELETE).

THE RESULT OF THE SUBQUERY IS USED BY THE OUTER QUERY TO FILTER, COMPARE, OR RETRIEVE DATA.

SUBQUERIES ARE USEFUL FOR TASKS LIKE COMPARING VALUES, CHECKING CONDITIONS, OR RETRIEVING RELATED DATA FROM OTHER TABLES.



TASK 10: FIND EMPLOYEES EARNING MORE THAN THE AVERAGE SALARY OF THEIR BRANCH

```
308 • SELECT
309     e.emp_id,
310     e.emp_name,
311     e.branch_id,
312     e.salary
313     FROM employees AS e
314 WHERE e.salary > (
315     SELECT AVG(e2.salary)
316     FROM employees AS e2
317     WHERE e2.branch_id = e.branch_id
318 );
```

	emp_id	emp_name	branch_id	salary
▶	E101	John Doe	B001	60000
	E103	Mike Johnson	B001	55000
	E107	Michael Thompson	B005	62000
	E111	Christopher Lee	B005	65000

INSIGHT: - THIS QUERY COMPARES EACH EMPLOYEE'S SALARY WITH THE AVERAGE SALARY OF THEIR OWN BRANCH USING A CORRELATED SUBQUERY. THE RESULT SHOWS ONLY THOSE EMPLOYEES WHO ARE PAID HIGHER THAN THE AVERAGE SALARY WITHIN THEIR BRANCH. THIS HELPS IDENTIFY TOP EARNERS OR POTENTIALLY SENIOR EMPLOYEES IN EACH BRANCH BASED ON SALARY PATTERNS.



CONCLUSION

- THIS LIBRARY MANAGEMENT SYSTEM PROJECT DEMONSTRATES HOW SQL CAN MANAGE AND ORGANIZE LIBRARY DATA EFFICIENTLY USING STRUCTURED TABLES AND RELATIONSHIPS.
- CRUD OPERATIONS, JOINS, FILTERS, AND GROUPING FUNCTIONS HELPED SIMULATE REAL LIBRARY OPERATIONS LIKE ISSUING BOOKS, TRACKING RETURNS, AND UPDATING RECORDS.
- ADVANCED FEATURES SUCH AS WINDOW FUNCTIONS AND SUBQUERIES PROVIDED DEEPER INSIGHTS, INCLUDING EMPLOYEE SALARY RANKINGS AND BRANCH-WISE ANALYSIS.
- OVERALL, THE PROJECT SHOWS HOW SQL SUPPORTS BOTH OPERATIONAL TASKS AND ANALYTICAL REPORTING, MAKING THE SYSTEM MORE ORGANIZED, ACCURATE, AND USEFUL FOR DECISION-MAKING.



THANK YOU

PRESENTATION BY
KRUTIKA JAMKHANDI

