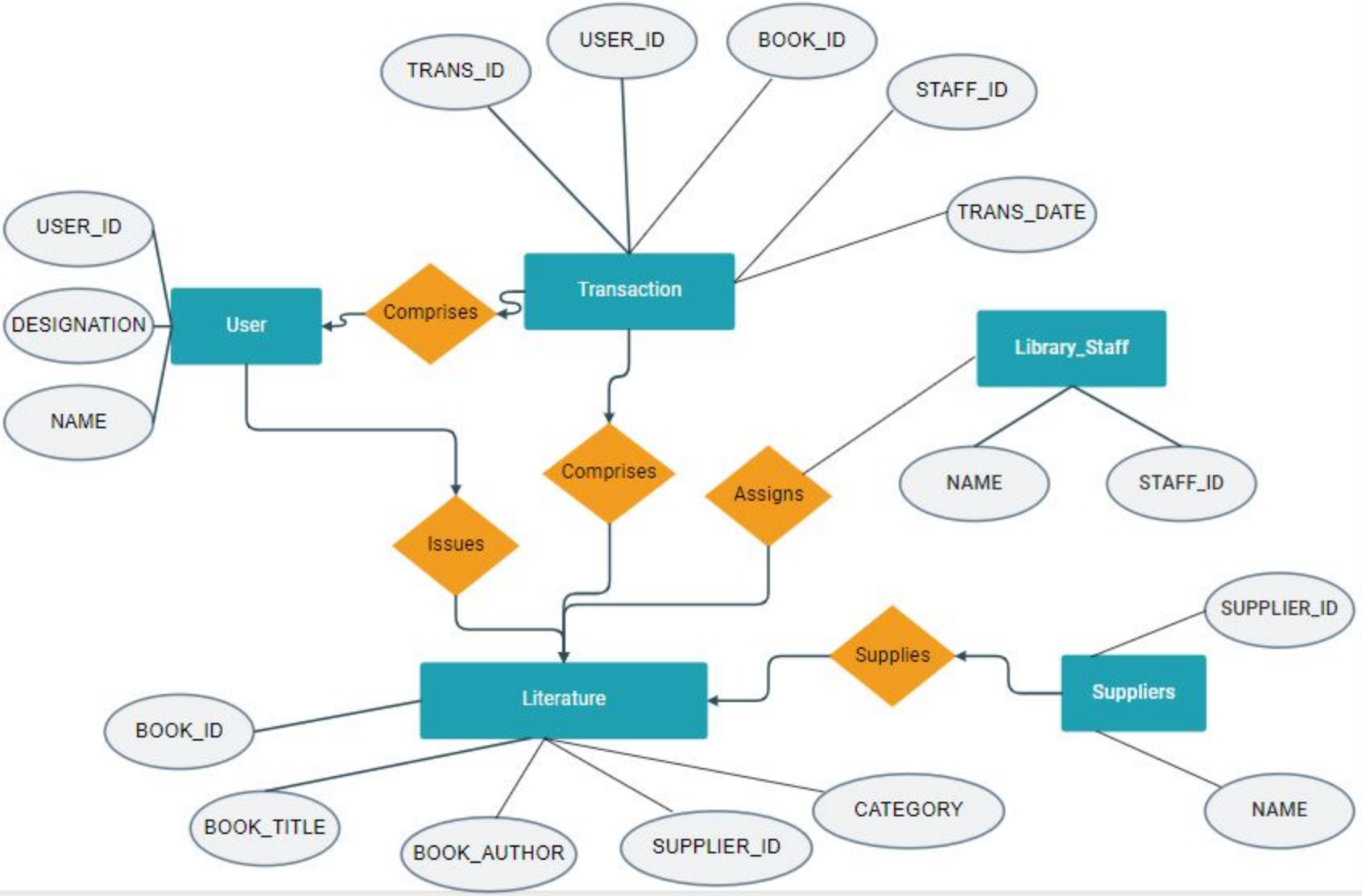
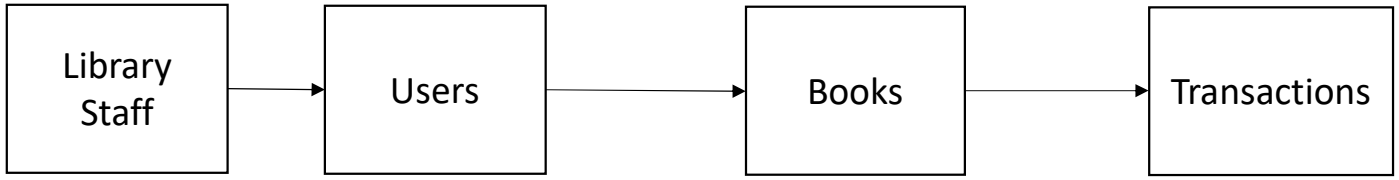
Library Management System ERD



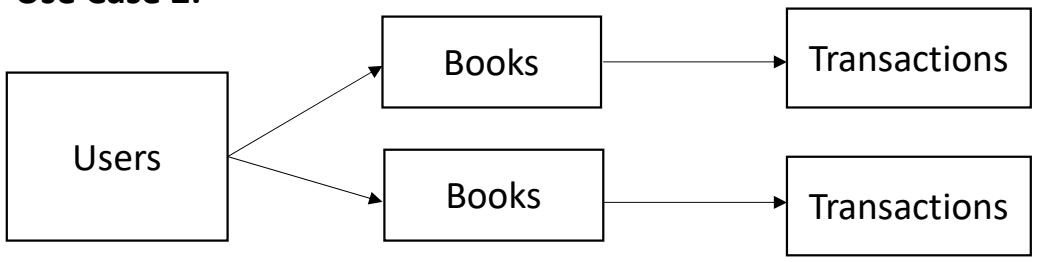
Use Cases

Use Case 1:



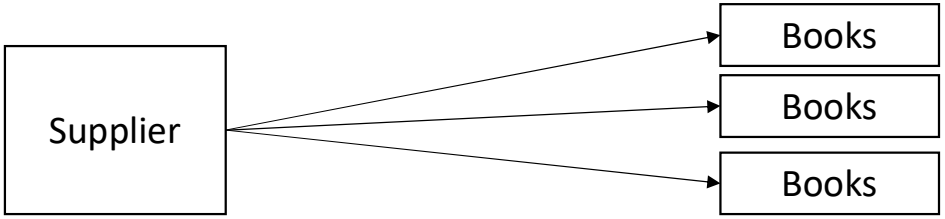
A library staff will issue user with ### user id. That user will borrow/return book with #### book id and it will get recorded in transactions table with unique transaction id.

Use Case 2:



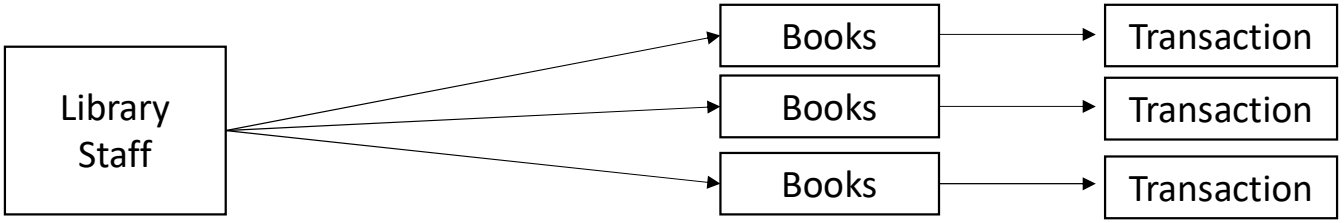
A user with ### user id borrows/returns multiple books and records separate transactions in table.

Use Case 3:



A supplier with ### supplier id will supply multiple books with different book ids

Use Case 4:



A library staff can issue/return multiple books to a user which will be recorded in a multiple transactions.

Tables

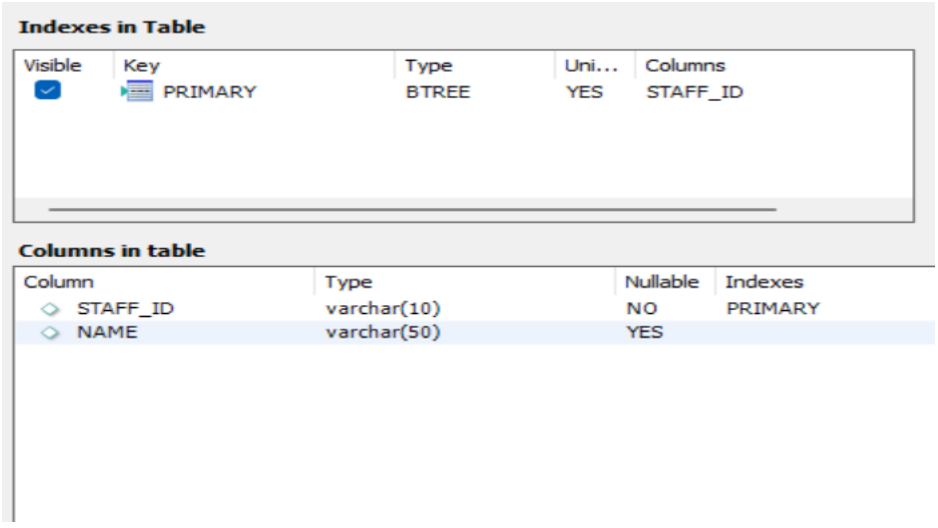
1. Library Staff Table

Query:

```
CREATE TABLE library_staff(
STAFF_ID    VARCHAR(10) NOT NULL,
NAME        VARCHAR(50),
PRIMARY KEY (STAFF_ID)
);
```

Constraints:

```
'CREATE TABLE `library_staff` (
  `STAFF_ID` varchar(10) NOT NULL,
  `NAME` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`STAFF_ID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci'
```



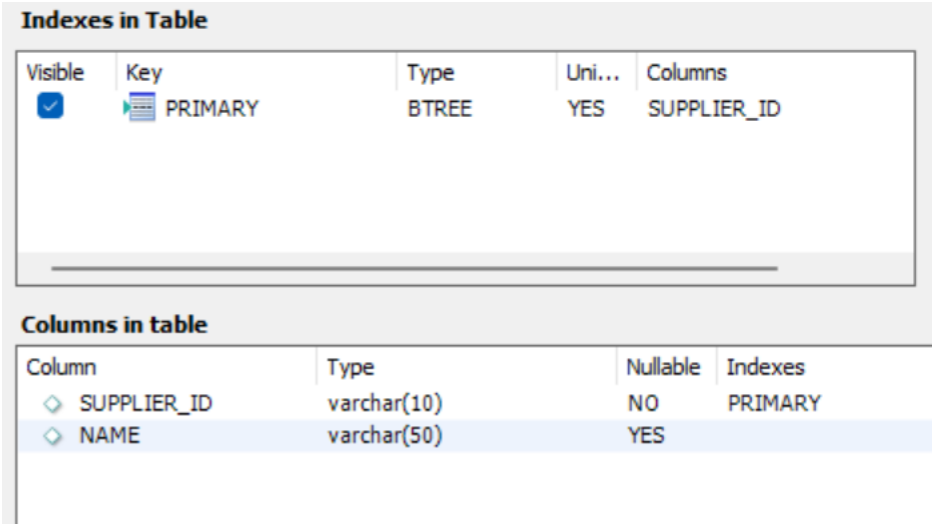
2. Suppliers Table

Query:

```
Create table suppliers(
SUPPLIER_ID    VARCHAR(10) NOT NULL,
NAME           VARCHAR(50),
PRIMARY KEY (SUPPLIER_ID)
);
```

Constraints:

```
'CREATE TABLE `suppliers` (
  `SUPPLIER_ID` varchar(10) NOT NULL,
  `NAME` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`SUPPLIER_ID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci'
```



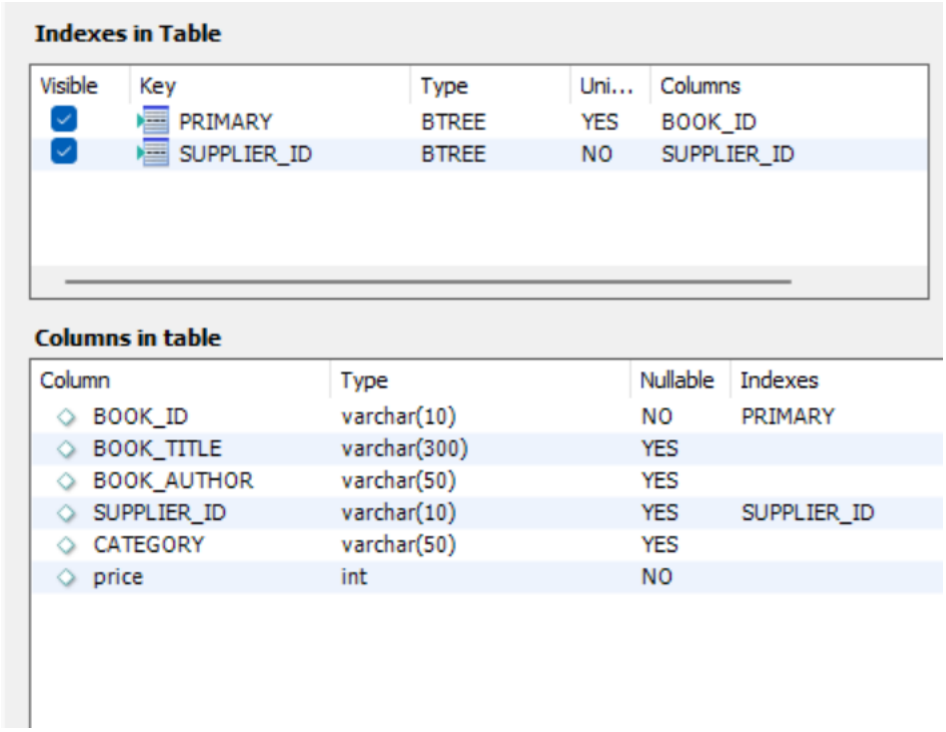
3. Books Table

Query:

Create table books(BOOK_ID VARCHAR(10) NOT NULL,BOOK_TITLE VARCHAR(300),
BOOK_AUTHOR VARCHAR(50), SUPPLIER_ID VARCHAR(10), CATEGORY VARCHAR(50),
price integer not null, PRIMARY KEY (BOOK_ID), FOREIGN KEY (SUPPLIER_ID) REFERENCES
suppliers(SUPPLIER_ID), constraint price_check check (price >= 0), constraint category_check
check (category = 'Finance' or category = 'Marketing' or category = 'Operations')
);

Constraints:

'CREATE TABLE `books` (
  `BOOK_ID` varchar(10) NOT NULL,
  `BOOK_TITLE` varchar(300) DEFAULT NULL,
  `BOOK_AUTHOR` varchar(50) DEFAULT NULL,
  `SUPPLIER_ID` varchar(10) DEFAULT NULL,
  `CATEGORY` varchar(50) DEFAULT NULL,
  `price` int NOT NULL,
  PRIMARY KEY (`BOOK_ID`),
  KEY `SUPPLIER_ID` (`SUPPLIER_ID`),
  CONSTRAINT `books_ibfk_1` FOREIGN KEY (`SUPPLIER_ID`) REFERENCES `suppliers`
(`SUPPLIER_ID`),
  CONSTRAINT `category_check` CHECK (((`category` = _utf8mb4''Finance'') or (`category` =
_utf8mb4''Marketing'') or (`category` = _utf8mb4''Operations''))),
  CONSTRAINT `price_check` CHECK ((`price` >= 0))
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci'



Indexes in Table

Visible	Key	Type	Uni...	Columns
☑	PRIMARY	BTREE	YES	BOOK_ID
☑	SUPPLIER_ID	BTREE	NO	SUPPLIER_ID

Columns in table

Column	Type	Nullable	Indexes
BOOK_ID	varchar(10)	NO	PRIMARY
BOOK_TITLE	varchar(300)	YES	
BOOK_AUTHOR	varchar(50)	YES	
SUPPLIER_ID	varchar(10)	YES	SUPPLIER_ID
CATEGORY	varchar(50)	YES	
price	int	NO	

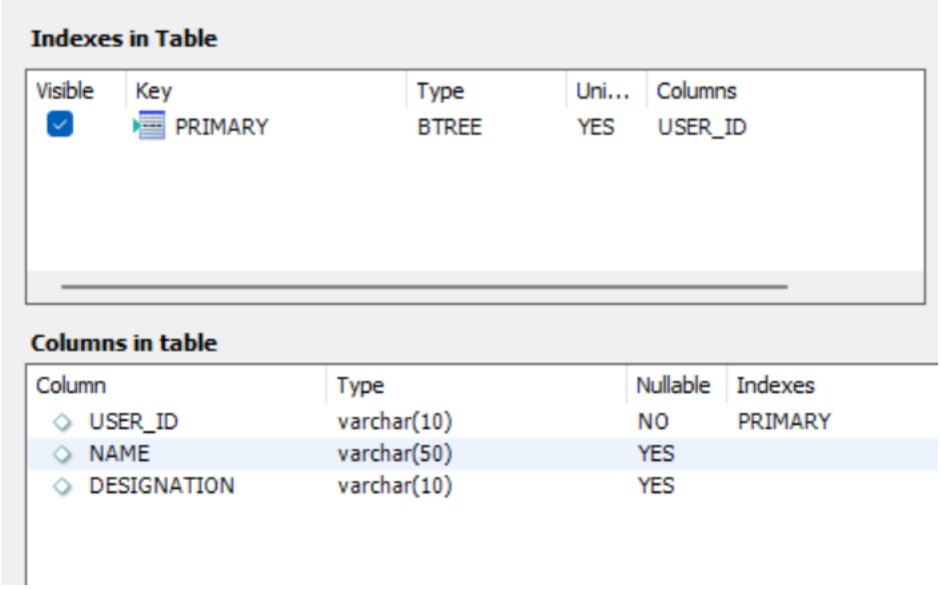
4. Users Table

Query:

```
Create table users(
USER_ID     VARCHAR(10) NOT NULL,
NAME        VARCHAR(50),
DESIGNATION  VARCHAR(10),
PRIMARY KEY (USER_ID),
constraint user_check check (DESIGNATION = 'STUDENT' or DESIGNATION = 'STAFF' or
DESIGNATION = 'OTHERS')
);
```

Constraints:

```
'CREATE TABLE `users` (
  `USER_ID` varchar(10) NOT NULL,
  `NAME` varchar(50) DEFAULT NULL,
  `DESIGNATION` varchar(10) DEFAULT NULL,
  PRIMARY KEY (`USER_ID`),
  CONSTRAINT `user_check` CHECK (((`DESIGNATION` = _utf8mb4''STUDENT'') or (`DESIGNATION`
= _utf8mb4''STAFF'') or (`DESIGNATION` = _utf8mb4''OTHERS'')))
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci'
```



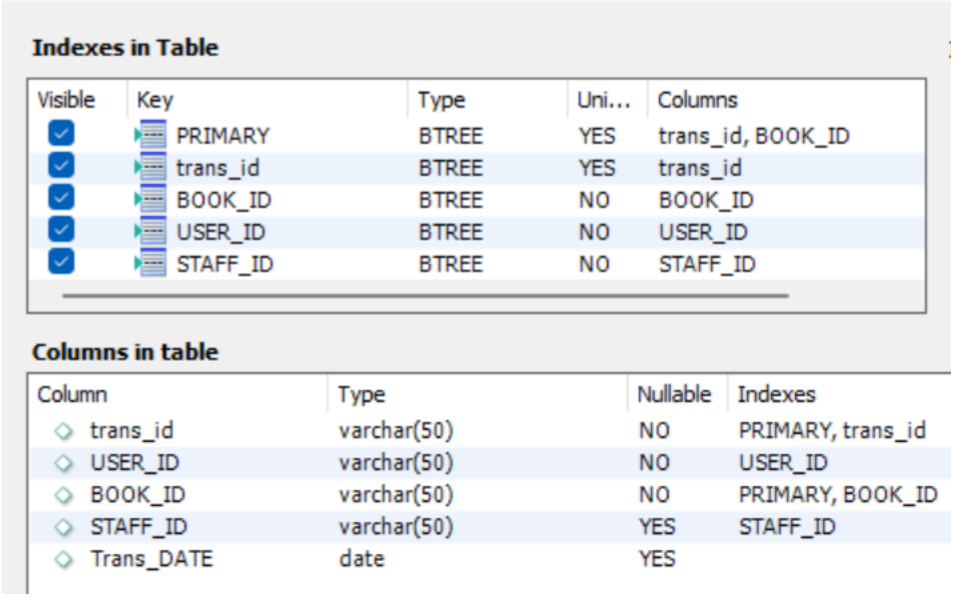
5. Transactions Table

Query:

```
create table Transactions(
trans_id varchar(50) UNIQUE,
USER_ID varchar(50) NOT NULL,
BOOK_ID varchar (50) NOT NULL,
STAFF_ID varchar(50),
Trans_DATE date,
PRIMARY KEY(trans_id, BOOK_ID),
FOREIGN KEY (BOOK_ID) REFERENCES books(BOOK_ID) ,
FOREIGN KEY (USER_ID) REFERENCES Users (USER_ID),
FOREIGN KEY (STAFF_ID) REFERENCES library_staff (STAFF_ID)
);
```

Constraints:

```
'CREATE TABLE `transactions` (
  `trans_id` varchar(50) NOT NULL,
  `USER_ID` varchar(50) NOT NULL,
  `BOOK_ID` varchar(50) NOT NULL,
  `STAFF_ID` varchar(50) DEFAULT NULL,
  `Trans_DATE` date DEFAULT NULL,
  PRIMARY KEY (`trans_id`,`BOOK_ID`),
  UNIQUE KEY `trans_id` (`trans_id`),
  KEY `BOOK_ID` (`BOOK_ID`),
  KEY `USER_ID` (`USER_ID`),
  KEY `STAFF_ID` (`STAFF_ID`),
  CONSTRAINT `transactions_ibfk_1` FOREIGN KEY (`BOOK_ID`) REFERENCES `books` (`BOOK_ID`),
  CONSTRAINT `transactions_ibfk_2` FOREIGN KEY (`USER_ID`) REFERENCES `users` (`USER_ID`),
  CONSTRAINT `transactions_ibfk_3` FOREIGN KEY (`STAFF_ID`) REFERENCES `library_staff`
(`STAFF_ID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci'
```



Indexes in Table

Visible	Key	Type	Uni...	Columns
✓	PRIMARY	BTREE	YES	trans_id, BOOK_ID
✓	trans_id	BTREE	YES	trans_id
✓	BOOK_ID	BTREE	NO	BOOK_ID
✓	USER_ID	BTREE	NO	USER_ID
✓	STAFF_ID	BTREE	NO	STAFF_ID

Columns in table

Column	Type	Nullable	Indexes
trans_id	varchar(50)	NO	PRIMARY, trans_id
USER_ID	varchar(50)	NO	USER_ID
BOOK_ID	varchar(50)	NO	PRIMARY, BOOK_ID
STAFF_ID	varchar(50)	YES	STAFF_ID
Trans_DATE	date	YES	

List of Probable Business Queries

1. List of books which are issued more than 2 times.

Query: select b.book_title from transactions t inner join users u on t.user_id = u.user_id inner join books b on t.book_id = b.book_id group by t.user_id, u.name having count(t.user_id) >= 2 order by u.name;

```
134    #Select books which are isseued more than 2 times
135 •  select b.book_title from transactions t inner join users u on t.user_id = u.user_id
136    inner join books b on t.book_id = b.book_id group by t.user_id, u.name
137    having count(t.user_id) >= 2 order by u.name;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

book_title
Investment concept every Indian should know
Distriution Network
Basics of Accounting
Capital Markets

2. List of books which are issued by a specific supplier.

Query: select b.BOOK_ID, b.BOOK_TITLE, s.NAME from books b inner join Suppliers s on b.supplier_id = s.supplier_id where s.SUPPLIER_ID='Sup888' order by b.book_id;

```
139    # Select books supplied by specific supplier
140 •  select b.BOOK_ID, b.BOOK_TITLE, s.NAME
141    from books b inner join Suppliers s on b.supplier_id = s.supplier_id
142    where s.SUPPLIER_ID='Sup888' order by b.book_id;
143
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

BOOK_ID	BOOK_TITLE	NAME
b074	Marketing Basics	John Dwayne
b120	Marketing Mix	John Dwayne
b127	Advanced Marketing	John Dwayne
b545	Brand Marketing	John Dwayne
b565	Online Marketing	John Dwayne
b895	Product Mix	John Dwayne

3. List of books from Finance Category.

Query: select b.BOOK_TITLE, u.name from transactions t inner join books b on t.book_id = b.book_id inner join users u on u.user_id = t.user_id where b.CATEGORY = 'Finance' order by b.BOOK_TITLE ASC;

```
143    #select books borrowed from finance category by people
144 •  select b.BOOK_TITLE, u.name from transactions t
145    inner join books b on t.book_id = b.book_id
146    inner join users u on u.user_id = t.user_id
147    where b.CATEGORY = 'Finance' order by b.BOOK_TITLE ASC;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

BOOK_TITLE	name
Basics of Accounting	Nishant
Basics of Accounting	Nishant
Capital Markets	Vinod Shah
Capital Markets	Vinod Shah
Financial Accounting	James
Financial Analytics	Avinash Adhikari
Introduction to Capital Budgeting	Nishant
Investment concept every Indian should know	Aditya
Investment concept every Indian should know	Aditya

4. List of books borrowed by a particular user.

Query: select count(b.book_id), u.name from transactions t inner join books b on t.book_id = b.book_id inner join users u on u.user_id = t.user_id where t.USER_ID = 'STUD_001';

```
150    # user with id-### borrowed how many books
151 •  select count(b.book_id), u.name from transactions t
152    inner join books b on t.book_id = b.book_id
153    inner join users u on u.user_id = t.user_id
154    where t.USER_ID = 'STUD_001';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

count(b.book_id)	name
3	Nishant

5. List of users which borrowed more than 2 books.

Query: select count(b.book_id) as book_count, u.name from transactions t inner join books b on t.book_id = b.book_id inner join users u on u.user_id = t.user_id group by t.user_id having count(t.USER_ID) >= 2 ;

```
156    #which user borrowed  more than 2 books
157 •  select count(b.book_id) as book_count, u.name from transactions t
158    inner join books b on t.book_id = b.book_id
159    inner join users u on u.user_id = t.user_id
160    group by t.user_id having count(t.USER_ID) >= 2 ;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

book_count	name
3	Nishant
2	Aryan
2	Aditya
2	Vinod Shah

6. List of count of books which staff which issued maximum

Query : select count(t.STAFF_ID),l.name from transactions t inner join library_staff l on t.STAFF_ID = l.STAFF_ID group by l.STAFF_ID;

```
155    #staff which issued maximum number of books
156 •  select count(t.STAFF_ID),l.name
157    from transactions t inner join library_staff l on t.STAFF_ID = l.STAFF_ID
158    group by l.STAFF_ID;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

count(t.STAFF_ID)	name
2	Johnny Grace
2	Kumari Shankar
6	Kumar Sanu
4	Udit Udaas

7. List of books issued between two dates

Query: select * from transactions as t where t.trans_date in ('2021-02-16','2022-02-10');

```
160
161    #no of transactions between two dates
162 •  select * from transactions as t where t.trans_date in ('2021-02-16','2022-02-10');
163
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

trans_id	USER_ID	BOOK_ID	STAFF_ID	Trans_DATE
trans3009	STAFF_005	b744	LIB_786	2021-02-16
trans3010	STUD_009	b720	LIB_069	2021-02-16
NULL	NULL	NULL	NULL	NULL

8. Who borrowed 'Distribution Network' book?

Query: select u.name from transactions as t inner join users as u on t.USER_ID=u.USER_ID inner join books as b on t.BOOK_ID=b.BOOK_ID where b.BOOK_TITLE= 'Distriution Network';

```
163
164    #which student/staff borrowed  'Distriution Network'
165 •  select u.name from transactions as t inner join users as u on t.USER_ID=u.USER_ID
166    inner join books as b on t.BOOK_ID=b.BOOK_ID
167    where b.BOOK_TITLE= 'Distriution Network';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

name
Aryan
Ben

9. Number of books borrowed by student and those by staff

Query: select count(u.USER_ID), u.designation from users as u inner join transactions as t on u.user_id=t.user_id group by u.designation order by u.designation desc;

```
169    #number of books borrowed by student and staff
170 •  select count(u.USER_ID), u.designation
171    from users as u inner join transactions as t on u.user_id=t.user_id
172    group by u.designation order by u.designation desc;
173
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

count(u.USER_ID)	designation
10	STUDENT
4	STAFF

Git-hub link

https://github.com/aditeeN/SQL_database_project