| Sr, NO | SQl Querry | Output |
|---|---|---|
| 1 | CREATE TABLE Library( Lib_ID int UNIQUE PRIMARY KEY NOT NULL, Lib_name varchar(50) NOT NULL UNIQUE, Qualification varchar(50) NOT NULL CHECK( Qualification >=12), Ph_No INT UNIQUE NOT NULL CHECK(Ph_no BETWEEN 0000000000 AND 9999999999) ); | **Library** <table><tr><td>Lib_ID</td><td>Lib_name</td><td>Qualification</td><td>Ph_No</td></tr><tr><td colspan="4">empty</td></tr></table> |
| 2. | CREATE TABLE Book ( Book_ID INT PRIMARY KEY UNIQUE NOT NULL, Book_Name VARCHAR(50) UNIQUE NOT NULL, Author VARCHAR(50) NOT NULL, Publisher VARCHAR(50) NOT NULL, Lib_ID INT NOT NULL, is_issued INTEGER DEFAULT 0, FOREIGN KEY (Lib_ID) REFERENCES Library(Lib_ID) ON DELETE CASCADE ); | **Book** <table><tr><td>Book_ID</td><td>Book_Name</td><td>Author</td><td>Publisher</td><td>Lib_ID</td><td>is_issued</td></tr><tr><td colspan="6">empty</td></tr></table> |
| 3. | CREATE TABLE Student( Student_ID int UNIQUE NOT NULL PRIMARY KEY, | **Student** <table><tr><td>Student_ID</td><td>S_name</td><td>Roll_no</td><td>Branch</td><td>Lib_ID</td></tr><tr><td colspan="5">empty</td></tr></table> |

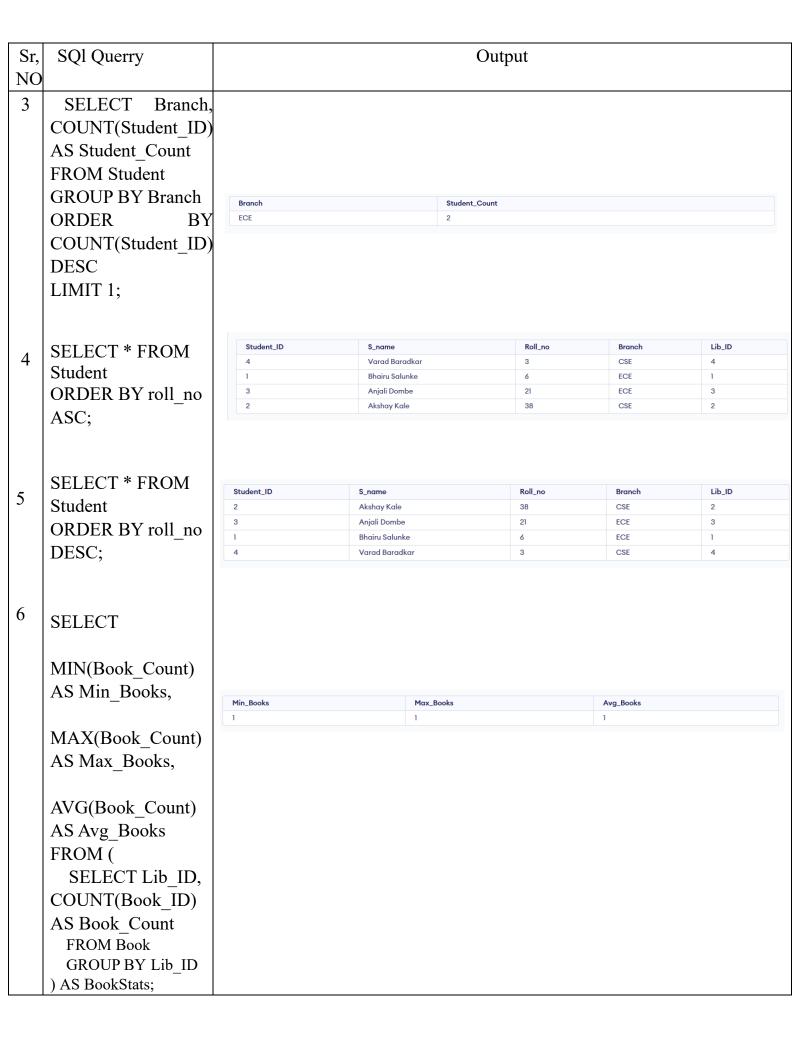| | | |
|---|---|---|
| | S_name varchar(50) NOT NULL,<br>   Roll_no int UNIQUE NOT NULL,<br>   Branch varchar(50) DEFAULT 'ECE'<br>   ); | |
| 4. | INSERT INTO Library (Lib_ID, lib_name, qualification, Ph_no) VALUES<br>(1, 'MIT Library', 'BTech', 9511270523),<br>(2, 'Central Library ', 'MTech', 8080626200),<br>(3, 'Techno Library', 'Diploma', 9356866971),<br>(4, 'School Library', 'PhD', 8767349797),<br>(5, 'Gandhi Library', 'BTech', 9223645893); | **Library**<br><br>| Lib_ID | Lib_name | Qualification | Ph_No |<br>|---|---|---|---|<br>| 1 | MIT Library | BTech | 9511270523 |<br>| 2 | Central Library | MTech | 8080626200 |<br>| 3 | Techno Library | Diploma | 9356866971 |<br>| 4 | School Library | PhD | 8767349797 |<br>| 5 | Gandhi Library | BTech | 9223645893 | |
| 5. | INSERT INTO Student (Student_ID, s_name, roll_no, Branch, Lib_ID) VALUES<br>(1, 'Bhairu Salunke', '06', 'ECE', 1),<br>(2, 'Akshay Kale ', '38', 'CSE', 2),<br>(3, 'Anjali Dombe', '21', 'ECE', 3),<br>(4, 'Varad Baradkar', '03', 'CSE', 4); | **Student**<br><br>| Student_ID | S_name | Roll_no | Branch | Lib_ID |<br>|---|---|---|---|---|<br>| 1 | Bhairu Salunke | 6 | ECE | 1 |<br>| 2 | Akshay Kale | 38 | CSE | 2 |<br>| 3 | Anjali Dombe | 21 | ECE | 3 |<br>| 4 | Varad Baradkar | 3 | CSE | 4 | |

| Sr, NO | SQl Querry | Output |
|---|---|---|
| 6. | INSERT INTO Book (Book_ID, book_name, author, publisher, Lib_ID, is_issued) VALUES (1001, 'The Alchemist', 'Paulo Coelho', 'HarperCollins', 2, 1), (1002, 'Think and Grow Rich', 'Napoleon Hill', 'TarcherPerigee', 3,1), (1003, 'Rich Dad Poor Dad', 'Robert Kiyosaki', 'Warner Books', 1,1), (1004, 'The 5 AM Club', 'Robin Sharma', 'HarperCollins', 4,1); | **Book**<br><br>| Book_ID | Book_Name | Author | Publisher | Lib_ID | is_issued |<br>|---|---|---|---|---|---|<br>| 1001 | The Alchemist | Paulo Coelho | HarperCollins | 2 | 1 |<br>| 1002 | Think and Grow Rich | Napoleon Hill | TarcherPerigee | 3 | 1 |<br>| 1003 | Rich Dad Poor Dad | Robert Kiyosaki | Warner Books | 1 | 1 |<br>| 1004 | The 5 AM Club | Robin Sharma | HarperCollins | 4 | 1 | |

| Sr, NO | SQl Querry | Output |
|--------|-----------|--------|
| 1. | SELECT Branch, COUNT(Student_ID) AS Total_Students FROM Student GROUP BY Branch ORDER BY COUNT(Student_ID) DESC; | <table><tr><th>Branch</th><th>Total_Students</th></tr><tr><td>ECE</td><td>2</td></tr><tr><td>CSE</td><td>2</td></tr></table> |
| 2. | SELECT l.lib_name, COUNT(b.Book_ID) AS Total_Books FROM Library l JOIN Book b ON l.Lib_ID = b.Lib_ID GROUP BY l.lib_name ORDER BY COUNT(b.Book_ID) DESC; | <table><tr><th>Lib_name</th><th>Total_Books</th></tr><tr><td>Techno Library</td><td>1</td></tr><tr><td>School Library</td><td>1</td></tr><tr><td>MIT Library</td><td>1</td></tr><tr><td>Central Library</td><td>1</td></tr></table> |

| Sr, NO | SQl Querry | Output |
|---|---|---|
| 3 | SELECT Branch, COUNT(Student_ID) AS Student_Count FROM Student GROUP BY Branch ORDER BY COUNT(Student_ID) DESC LIMIT 1; | <table><tr><th>Branch</th><th>Student_Count</th></tr><tr><td>ECE</td><td>2</td></tr></table> |
| 4 | SELECT * FROM Student ORDER BY roll_no ASC; | See table below |
| 5 | SELECT * FROM Student ORDER BY roll_no DESC; | See table below |
| 6 | SELECT MIN(Book_Count) AS Min_Books, MAX(Book_Count) AS Max_Books, AVG(Book_Count) AS Avg_Books FROM ( SELECT Lib_ID, COUNT(Book_ID) AS Book_Count FROM Book GROUP BY Lib_ID ) AS BookStats; | See table below |

Output for query 4:

| Student_ID | S_name | Roll_no | Branch | Lib_ID |
|---|---|---|---|---|
| 4 | Varad Baradkar | 3 | CSE | 4 |
| 1 | Bhairu Salunke | 6 | ECE | 1 |
| 3 | Anjali Dombe | 21 | ECE | 3 |
| 2 | Akshay Kale | 38 | CSE | 2 |

Output for query 5:

| Student_ID | S_name | Roll_no | Branch | Lib_ID |
|---|---|---|---|---|
| 2 | Akshay Kale | 38 | CSE | 2 |
| 3 | Anjali Dombe | 21 | ECE | 3 |
| 1 | Bhairu Salunke | 6 | ECE | 1 |
| 4 | Varad Baradkar | 3 | CSE | 4 |

Output for query 6:

| Min_Books | Max_Books | Avg_Books |
|---|---|---|
| 1 | 1 | 1 |

| Sr, NO | SQl Querry | Output |
|---|---|---|
| 7 | SELECT MIN(Roll_no) AS Min_Roll, MAX(Roll_no) AS Max_Roll, AVG(Roll_no) AS Avg_Roll FROM Student; | <table><tr><td>Min_Roll</td><td>Max_Roll</td><td>Avg_Roll</td></tr><tr><td>3</td><td>38</td><td>17</td></tr></table> |
| 8 | SELECT SUM(is_issued) AS Total_Issued_Books FROM Book; | <table><tr><td>Total_Issued_Books</td></tr><tr><td>4</td></tr></table> |