***Library Management System***

*You are going to build a project based on Library Management System. It keeps track of all information about books in the library, their cost, status and total number of books available in the library. Create a database named library and following TABLES in the database: 1. Branch 2. Employee 3. Books 4. Customer 5. IssueStatus 6. ReturnStatus Attributes for the tables: 1. Branch Branch\_no - Set as PRIMARY KEY Manager\_Id Branch\_address Contact\_no 2. Employee Emp\_Id – Set as PRIMARY KEY Emp\_name Position Salary Branch\_no - Set as FOREIGN KEY and it refer Branch\_no in Branch table 3. Books ISBN - Set as PRIMARY KEY Book\_title Category Rental\_Price Status [Give yes if book available and no if book not available] Author Publisher 4. Customer Customer\_Id - Set as PRIMARY KEY Customer\_name Customer\_address Reg\_date 5. IssueStatus Issue\_Id - Set as PRIMARY KEY Issued\_cust – Set as FOREIGN KEY and it refer customer\_id in CUSTOMER table Issued\_book\_name Issue\_date Isbn\_book – Set as FOREIGN KEY and it should refer isbn in BOOKS table 6. ReturnStatus Return\_Id - Set as PRIMARY KEY Return\_cust Return\_book\_name Return\_date Isbn\_book2 - Set as FOREIGN KEY and it should refer isbn in BOOKS table Display all the tables and Write the queries for the following : 1. Retrieve the book title, category, and rental price of all available books. 2. List the employee names and their respective salaries in descending order of salary. 3. Retrieve the book titles and the corresponding customers who have issued those books. 4. Display the total count of books in each category. 5. Retrieve the employee names and their positions for the employees whose salaries are above Rs.50,000. 6. List the customer names who registered before 2022-01-01 and have not issued any books yet. 7. Display the branch numbers and the total count of employees in each branch. 8. Display the names of customers who have issued books in the month of June 2023. 9. Retrieve book\_title from book table containing history. 10.Retrieve the branch numbers along with the count of employees for branches having more than 5 employees 11. Retrieve the names of employees who manage branches and their respective branch addresses. 12. Display the names of customers who have issued books with a rental price higher than Rs. 25.*

*CREATE DATABASE LIBRARY;*

*USE LIBRARY;*

*CREATE TABLE Branch(*

*Branch\_No INT PRIMARY KEY,*

*Manager\_Id INT,*

*Branch\_Address VARCHAR(255),*

*Contact\_No INT*

*);*

*CREATE TABLE Employee (*

*Emp\_Id INT PRIMARY KEY,*

*Emp\_Name VARCHAR(100),*

*Position VARCHAR(50),*

*Salary DECIMAL(10, 2),*

*Branch\_No INT,*

*FOREIGN KEY (Branch\_No) REFERENCES Branch(Branch\_No)*

*);*

*CREATE TABLE Books (*

*ISBN INT PRIMARY KEY,*

*Book\_Title VARCHAR(255),*

*Category VARCHAR(100),*

*Rental\_Price DECIMAL(10, 2),*

*Status VARCHAR(3) CHECK (Status IN ('Yes', 'No')) NOT NULL,*

*Author VARCHAR(100),*

*Publisher VARCHAR(100)*

*);*

*CREATE TABLE Customer (*

*Customer\_Id INT PRIMARY KEY,*

*Customer\_Name VARCHAR(100),*

*Customer\_Address VARCHAR(255),*

*Reg\_Date DATE*

*);*

*CREATE TABLE IssueStatus (*

*Issue\_Id INT PRIMARY KEY,*

*Issued\_Cust INT,*

*Issued\_Book\_Name VARCHAR(255),*

*Issue\_Date DATE,*

*ISBN\_Book INT,*

*FOREIGN KEY (Issued\_Cust) REFERENCES Customer(Customer\_Id),*

*FOREIGN KEY (Isbn\_book) REFERENCES Books(ISBN)*

*);*

*CREATE TABLE ReturnStatus (*

*Return\_Id INT PRIMARY KEY,*

*Return\_Cust INT,*

*Return\_Book\_Name VARCHAR(255),*

*Return\_Date DATE,*

*ISBN\_Book2 INT,*

*FOREIGN KEY (Return\_Cust) REFERENCES Customer(Customer\_Id),*

*FOREIGN KEY (Isbn\_book2) REFERENCES Books(ISBN)*

*);*

*INSERT INTO Branch(Branch\_No, Manager\_Id, Branch\_Address, Contact\_No)*

*VALUES*

*(1, 111, '123 Elm Street, New York, NY', 212555123),*

*(2, 222, '456 Maple Avenue, Los Angeles, CA', 323555234),*

*(3, 333, '789 Pine Road, Chicago, IL', 312555345),*

*(4, 444, '321 Oak Street, Houston, TX', 713555456),*

*(5, 555, '654 Cedar Avenue, Phoenix, AZ', 602555567);*

*INSERT INTO Employee (Emp\_Id, Emp\_Name, Position, Salary, Branch\_No)*

*VALUES*

*(111, 'John Doe', 'Manager', 55000.00, 1),*

*(202, 'Jane Smith', 'Assistant Manager', 45000.00, 2),*

*(203, 'Michael Brown', 'Librarian', 40000.00, 3),*

*(204, 'Emily Davis', 'Clerk', 35000.00, 4),*

*(222, 'William Wilson', 'Manager', 56000.00, 5),*

*(206, 'Alice Johnson', 'Senior Librarian', 50000.00, 3),*

*(207, 'Bob Miller', 'Librarian', 42000.00, 3),*

*(208, 'Charlie Taylor', 'Assistant Librarian', 40000.00, 3),*

*(209, 'David Anderson', 'Clerk', 35000.00, 3),*

*(210, 'Eve Martinez', 'Janitor', 30000.00, 3),*

*(211, 'Michael Brown', 'Librarian', 42000.00, 2),*

*(212, 'Emily Davis', 'Clerk', 36000.00, 2),*

*(213, 'William Wilson', 'Janitor', 30000.00, 2),*

*(214, 'Olivia Taylor', 'Assistant Librarian', 40000.00, 2),*

*(215, 'James Anderson', 'IT Support', 45000.00, 2);*

*INSERT INTO Books (ISBN, Book\_Title, Category, Rental\_Price, Status, Author, Publisher)*

*VALUES*

*(1001, 'The C Programming Language', 'Programming', 25.99, 'Yes', 'Brian W. Kernighan Dennis M. Ritchie', 'Prentice Hall'),*

*(1002, 'Head First Java', 'Programming', 26.50, 'No', 'Kathy Sierra Bert Bates', 'O Reilly Media'),*

*(1003, 'The Great Gatsby', 'Fiction', 3.75, 'Yes', 'F. Scott Fitzgerald', 'Scribner'),*

*(1004, 'Introduction to Algorithms', 'Computer Science', 48.99, 'No', 'Thomas H. Cormen', 'MIT Press'),*

*(1005, '1984', 'Dystopian', 4.25, 'Yes', 'George Orwell', 'Signet Classics'),*

*(1006, 'The History of the Ancient World', 'History', 74.99, 'Yes', 'Susan Wise Bauer', 'W. W. Norton & Company'),*

*(1007, 'Guns, Germs, and Steel', 'History', 16.50, 'No', 'Jared Diamond', 'W. W. Norton & Company'),*

*(1008, 'The Silk Roads: A New History of the World', 'History', 22.25, 'Yes', 'Peter Frankopan', 'Vintage Books');*

*INSERT INTO Customer (Customer\_Id, Customer\_Name, Customer\_Address, Reg\_Date)*

*VALUES*

*(301, 'Alice Johnson', '789 Birch Lane, Seattle, WA', '2024-01-15'),*

*(302, 'Bob Miller', '456 Oak Drive, Austin, TX', '2024-02-10'),*

*(303, 'Charlie Taylor', '123 Pine Avenue, Boston, MA', '2024-03-05'),*

*(304, 'David Anderson', '321 Elm Road, Denver, CO', '2024-04-01'),*

*(305, 'Eve Martinez', '654 Cedar Street, Nashville, TN', '2024-05-12'),*

*(306, 'EMMA MARIA', '432 SAND FRANCISCO, USA', '2020-01-15'),*

*(307, 'TEKIE SEMAHAR', '775 JEBEL ALI, UAE', '2022-10-14'),*

*(308, 'JOHN LUKKA', '22 BLUE BERRY ROAD, CO', '2021-10-14');*

*INSERT INTO IssueStatus (Issue\_Id, Issued\_Cust, Issued\_Book\_Name, Issue\_Date, ISBN\_Book)*

*VALUES*

*(401, 301, 'The C Programming Language', '2024-06-01', 1001),*

*(402, 302, '1984', '2023-06-15', 1005),*

*(403, 303, 'Head First Java', '2024-07-01', 1002),*

*(404, 304, 'The Great Gatsby', '2023-06-10', 1003),*

*(405, 305, 'Introduction to Algorithms', '2024-08-01', 1004);*

*INSERT INTO ReturnStatus (Return\_Id, Return\_Cust, Return\_Book\_Name, Return\_Date, ISBN\_Book2)*

*VALUES*

*(501, 301, 'The C Programming Language', '2024-06-20', 1001),*

*(502, 302, '1984', '2024-06-25', 1005),*

*(503, 303, 'Head First Java', '2024-07-15', 1002),*

*(504, 304, 'The Great Gatsby', '2024-07-20', 1003),*

*(505, 305, 'Introduction to Algorithms', '2024-08-15', 1004);*

*select Book\_Title, category,Rental\_Price from Books ;*

A screenshot of a computer

Description automatically generated

*select Salary,Emp\_Name from employee*

*order by salary desc;*

A screenshot of a computer

Description automatically generated

*select books.Book\_Title ,customer.Customer\_Name from issuestatus*

*join books on issuestatus.ISBN\_Book=books.ISBN*

*join customer on issuestatus.issued\_Cust=customer.customer\_id;*

A screenshot of a computer

Description automatically generated

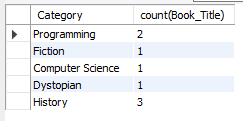
*select Issued\_Book\_Name,Issued\_Cust from issuestatus ;*

A screenshot of a computer

Description automatically generated

*select Category, count(Book\_Title) from books*

*group by Category;*



*select Emp\_Name, Position from employee*

*where Salary>50000;*

A screenshot of a computer

Description automatically generated

*select Customer\_Name from customer*

*where Reg\_Date < "2022-01-01";*

A screenshot of a computer

Description automatically generated

*select Branch\_No, count(\*) as Total\_Emp*

*from employee*

*group by Branch\_No;*

A screenshot of a data

Description automatically generated

*select Issued\_Cust from issuestatus*

*where Issue\_Date between "2023-06-01" and "2023-06-30";*

A screenshot of a computer

Description automatically generated

*select customer.Customer\_Name from customer*

*INNER JOIN issuestatus on customer.Customer\_Id=issuestatus.Issued\_Cust*

*where Issue\_Date between "2023-06-01" and "2023-06-30";*

A screenshot of a computer

Description automatically generated

*select Book\_Title from books where Category = "history";*

A screenshot of a computer

Description automatically generated

*select Branch\_No, count(\*) as count from employee*

*group by Branch\_No having count(\*)>5;*

A screenshot of a computer

Description automatically generated

*select employee.Emp\_Name ,branch.Branch\_Address from employee*

*inner join branch on employee.Emp\_id=branch.Manager\_Id;*

A screenshot of a phone

Description automatically generated

*select distinct Customer\_Name from customer*

*inner join issuestatus on customer.Customer\_Id=issuestatus.Issued\_Cust*

*join books on issuestatus.ISBN\_Book=books.isbn*

*where books.Rental\_Price>25;*

A screenshot of a computer

Description automatically generated