Experiment 1: Library Management System

Tasks

1. Create the following tables with appropriate constraints: books, members, loans.

2. Insert sample data into all tables.

3. Update the price of one book.

4. Delete one member.

5. Write a query to display the names of members along with the titles of books they have borrowed.

6. Write a query to count how many books each member has borrowed.

7. Add a user named 'lib_user' and grant SELECT and INSERT privileges on all three tables.

8. Drop all three tables.

Table Structures

Column	Data Type	Constraints	Description
book_id	INT	PRIMARY KEY	Book ID
title	VARCHAR(255)	NOT NULL	Book Title
author	VARCHAR(255)		Author Name
price	DECIMAL(10,2)	CHECK (price > 0)	Price (INR)
category	VARCHAR(100)	DEFAULT 'General'	Book Category

Column	Data Type	Constraints	Description
member_id	INT	PRIMARY KEY	Member ID
name	VARCHAR(255)	NOT NULL	Member Name
join_date	DATE		Date of Joining
phone	VARCHAR(10)		Phone Number

Column	Data Type	Constraints	Description
loan_id	INT	PRIMARY KEY	Loan ID
book_id	INT	FOREIGN KEY REFERENCES books(book_id)	Book ID
member_id	INT	FOREIGN KEY REFERENCES members(member_id)	Member ID
loan_date	DATE		Loan Date

Sample Data to be Inserted

book_id	title	author	price	category
1	Wings of Fire	A.P.J. Abdul Kalam	299.00	Biography
2	The Guide	R.K. Narayan	250.00	Fiction
3	Train to Pakistan	Khushwant Singh	199.00	Historical Fiction
4	The God of Small Things	Arundhati Roy	350.00	Literary Fiction

member_id	name	join_date	phone
101	Rajesh Kumar	2024-01-15	9876543210
102	Priya Sharma	2024-02-20	8765432109
103	Amit Patel	2024-03-10	7654321098

loan_id	book_id	member_id	loan_date
1	1	101	2025-04-01
2	2	102	2025-04-05
3	3	103	2025-04-10

Experiment 2: Hospital Appointments

Tasks

1. Create tables with appropriate constraints: patients and appointments.

2. Insert sample data into all tables.

3. Update the fee for one appointment.

4. Delete one patient record.

5. Write a query to display patient details with their appointment information.

6. Write a query to count appointments per doctor.

7. Create and execute a stored procedure to get appointments for a specific date.

8. Drop all tables.

Table Structures

patients

Column	Data Type	Constraints	Description
patient_id	INT	PRIMARY KEY	Patient ID
name	VARCHAR(100)	NOT NULL	Patient Name
gender	CHAR(1)	CHECK (gender IN ('M','F'))	Gender
phone	VARCHAR(10)	UNIQUE	Phone Number

appointments

Column	Data Type	Constraints	Description
appt_id	INT	PRIMARY KEY	Appointment ID
patient_id	INT	FOREIGN KEY REFERENCES patients(patient_id)	Patient ID
doctor_name	VARCHAR(100)	NOT NULL	Doctor Name
appt_date	DATE	NOT NULL	Appointment Date

fee	DECIMAL(8,2)	CHECK (fee > 0)	Consultation Fee

Sample Data to be Inserted

patients

patient_id	name	gender	phone
1	Rahul Sharma	M	9876543210
2	Priya Patel	F	8765432109
3	Amit Kumar	M	7654321098

appointments

appt_id	patient_id	doctor_name	appt_date	fee
101	1	Dr. Suresh Mehta	2025-04-10	500.0
102	2	Dr. Anjali Gupta	2025-04-11	600.0
103	3	Dr. Suresh Mehta	2025-04-12	500.0

Experiment 3: Online Store

Tasks

1. Create tables with appropriate constraints: products and orders.

2. Insert sample data into all tables.

3. Update the price of one product.

4. Delete one order.

5. Write a query to display order details with product information.

6. Write a query to show total sales per product.

7. Create and use a function to calculate total order value.

8. Drop all tables.

Table Structures

products

Column	Data Type	Constraints	Description
product_id	INT	PRIMARY KEY	Product ID
name	VARCHAR(100)	NOT NULL	Product Name
price	DECIMAL(10,2)	CHECK (price > 0)	Product Price
stock	INT	CHECK (stock >= 0)	Available Stock

orders

Column	Data Type	Constraints	Description
order_id	INT	PRIMARY KEY	Order ID
product_id	INT	FOREIGN KEY REFERENCES products(product_id)	Product ID
quantity	INT	CHECK (quantity > 0)	Order Quantity
order_date	DATE	DEFAULT CURRENT_DATE	Order Date

Sample Data to be Inserted

products

product_id	name	price	stock
1	Samsung Smart TV	45000.0	10
2	iPhone 13	65000.0	15
3	Sony Headphones	2500.0	25

orders

order_id	product_id	quantity	order_date
101	1	2	2025-04-10
102	2	1	2025-04-11
103	3	3	2025-04-12

Experiment 4: Student Enrollment

Tasks

1. Create tables with appropriate constraints: students, courses, and enrollments.

2. Insert sample data into all tables.

3. Update the credits of one course.

4. Delete one enrollment.

5. Write a query to display student enrollments with course details.

6. Write a query to show total enrollments per course.

7. Create users for faculty and admin with appropriate privileges.

8. Drop all tables.

Table Structures

students

Column	Data Type	Constraints	Description
student_id	INT	PRIMARY KEY	Student ID
name	VARCHAR(100)	NOT NULL	Student Name
branch	VARCHAR(50)	NOT NULL	Branch Name
join_year	INT	CHECK (join_year >= 2020)	Year of Joining

courses

Column	Data Type	Constraints	Description
course_id	INT	PRIMARY KEY	Course ID
course_name	VARCHAR(100)	NOT NULL	Course Name
credits	INT	CHECK (credits > 0)	Course Credits

enrollments

Column	Data Type	Constraints	Description
enroll_id	INT	PRIMARY KEY	Enrollment ID
student_id	INT	FOREIGN KEY REFERENCES students(student_id)	Student ID
course_id	INT	FOREIGN KEY REFERENCES courses(course_id)	Course ID
enroll_date	DATE	DEFAULT CURRENT_DATE	Enrollment Date

Sample Data to be Inserted

students

student_id	name	branch	join_year
1	Rahul Verma	Computer Science	2023
2	Priya Desai	Electronics	2023
3	Amit Shah	Mechanical	2022

courses

course_id	course_name	credits
101	Database Management	4
102	Data Structures	4
103	Digital Electronics	3

enrollments

enroll_id	student_id	course_id	enroll_date
1	1	101	2025-04-10
2	1	102	2025-04-10
3	2	102	2025-04-11

Experiment 5: Airline Booking

Tasks

1. Create tables with appropriate constraints: flights and bookings.

2. Insert sample data into all tables.

3. Update the seats available for one flight.

4. Delete one booking.

5. Write a query to display booking details with flight information.

6. Write a query to show total bookings per flight.

7. Create and execute a stored procedure to check flight availability.

8. Drop all tables.

Table Structures

flights

Column	Data Type	Constraints	Description
flight_id	INT	PRIMARY KEY	Flight ID
flight_no	VARCHAR(10)	NOT NULL	Flight Number
source	VARCHAR(50)	NOT NULL	Source City
destination	VARCHAR(50)	NOT NULL	Destination City
flight_date	DATE	NOT NULL	Flight Date
seats_available	INT	CHECK (seats_available >= 0)	Available Seats

bookings

Column	Data Type	Constraints	Description
booking_id	INT	PRIMARY KEY	Booking ID
flight_id	INT	FOREIGN KEY REFERENCES flights(flight_id)	Flight ID

passenger_name	VARCHAR(100)	NOT NULL	Passenger Name
seats_booked	INT	CHECK (seats_booked > 0)	Number of Seats
booking_date	DATE	DEFAULT CURRENT_DATE	Booking Date

Sample Data to be Inserted

flights

flight_id	flight_no	source	destination	flight_date	seats_available
1	AI101	Mumbai	Delhi	2025-04-15	120
2	6E202	Bangalore	Kolkata	2025-04-16	150
3	SG303	Chennai	Hyderabad	2025-04-15	180

bookings

booking_id	flight_id	passenger_name	seats_booked	booking_date
101	1	Rajesh Kumar	2	2025-04-10
102	2	Priya Sharma	1	2025-04-11
103	1	Amit Patel	3	2025-04-12

Experiment 6: Music Catalog

Tasks

1. Create tables with appropriate constraints: artists and albums.

2. Insert sample data into all tables.

3. Update the price of one album.

4. Delete one artist.

5. Write a query to display album titles with their artist names.

6. Write a query to show how many albums each artist has.

7. Create and use a function to get total value of albums by artist.

8. Drop all tables.

Table Structures

artists

Column	Data Type	Constraints	Description
artist_id	INT	PRIMARY KEY	Artist ID
name	VARCHAR(100)	NOT NULL	Artist Name
city	VARCHAR(50)		City

albums

Column	Data Type	Constraints	Description
album_id	INT	PRIMARY KEY	Album ID
artist_id	INT	FOREIGN KEY REFERENCES artists(artist_id)	Artist ID
title	VARCHAR(100)	NOT NULL	Album Title
release_year	INT		Release Year
price	DECIMAL(8,2)	CHECK (price > 0)	Album Price

Sample Data to be Inserted

artists

artist_id	name	city
1	A. R. Rahman	Chennai
2	Shreya Ghoshal	Mumbai
3	Arijit Singh	Kolkata

albums

album_id	artist_id	title	release_year	price
101	1	Roja	1992	250.0
102	1	Dil Se	1998	300.0
103	2	Melodies	2015	200.0
104	3	Soulful Voice	2018	220.0

Experiment 7: Car Rental

Tasks

1. Create tables with appropriate constraints: cars and rentals.

2. Insert sample data into all tables.

3. Update the daily rate for one car.

4. Delete one rental record.

5. Write a query to display rental details with car information.

6. Write a query to show total rentals per car model.

7. Create users for staff and manager with appropriate privileges.

8. Drop all tables.

Table Structures

cars

Column	Data Type	Constraints	Description
car_id	INT	PRIMARY KEY	Car ID
model	VARCHAR(50)	NOT NULL	Car Model
daily_rate	DECIMAL(8,2)	CHECK (daily_rate > 0)	Daily Rental Rate
available	CHAR(1)	CHECK (available IN ('Y','N'))	Availability Status

rentals

Column	Data Type	Constraints	Description
rental_id	INT	PRIMARY KEY	Rental ID
car_id	INT	FOREIGN KEY REFERENCES cars(car_id)	Car ID
customer_name	VARCHAR(100)	NOT NULL	Customer Name

start_date	DATE	NOT NULL	Rental Start Date
days	INT	CHECK (days > 0)	Number of Days

Sample Data to be Inserted

cars

car_id	model	daily_rate	available
1	Maruti Swift	1200.0	Y
2	Hyundai i20	1500.0	Y
3	Honda City	2000.0	Y

rentals

rental_id	car_id	customer_name	start_date	days
101	1	Vikram Mehta	2025-04-10	3
102	2	Neha Singh	2025-04-11	2
103	3	Rajesh Kumar	2025-04-12	4

Experiment 8: Hotel Management

Tasks

1. Create tables with appropriate constraints: rooms and bookings.

2. Insert sample data into all tables.

3. Update the rate per night for one room type.

4. Delete one booking record.

5. Write a query to display booking details with room information.

6. Write a query to show total revenue per room type.

7. Create and execute a stored procedure to check room availability.

8. Drop all tables.

Table Structures

rooms

Column	Data Type	Constraints	Description
room_id	INT	PRIMARY KEY	Room ID
room_type	VARCHAR(50)	NOT NULL	Room Type
rate_per_night	DECIMAL(8,2)	CHECK (rate_per_night > 0)	Rate per Night
status	VARCHAR(20)	CHECK (status IN ('Available','Occupied','Maintenance'))	Room Status

bookings

Column	Data Type	Constraints	Description
booking_id	INT	PRIMARY KEY	Booking ID
room_id	INT	FOREIGN KEY REFERENCES rooms(room_id)	Room ID
guest_name	VARCHAR(100)	NOT NULL	Guest Name
check_in	DATE	NOT NULL	Check-in Date

nights	INT	CHECK (nights > 0)	Number of Nights
total_amount	DECIMAL(10,2)	CHECK (total_amount > 0)	Total Amount

Sample Data to be Inserted

rooms

room_id	room_type	rate_per_night	status
101	Deluxe	3500.0	Available
102	Super Deluxe	4500.0	Occupied
103	Suite	6000.0	Available

bookings

booking_id	room_id	guest_name	check_in	nights	total_amount
1	101	Anand Verma	2025-04-15	2	7000.0
2	102	Meera Reddy	2025-04-16	3	13500.0
3	103	Suresh Kumar	2025-04-17	1	6000.0

Experiment 9: Bank Account

Tasks

1. Create tables with appropriate constraints: accounts and transactions.

2. Insert sample data into all tables.

3. Update the balance of one account.

4. Delete one transaction record.

5. Write a query to display transaction details with account information.

6. Write a query to show total transactions per account.

7. Create and use a function to calculate account balance after a transaction.

8. Drop all tables.

Table Structures

accounts

Column	Data Type	Constraints	Description
account_id	INT	PRIMARY KEY	Account ID
holder_name	VARCHAR(100)	NOT NULL	Account Holder Name
account_type	VARCHAR(20)	CHECK (account_type IN ('Savings','Current'))	Account Type
balance	DECIMAL(12,2)	CHECK (balance >= 0)	Current Balance

transactions

Column	Data Type	Constraints	Description
transaction_id	INT	PRIMARY KEY	Transaction ID
account_id	INT	FOREIGN KEY REFERENCES accounts(account_id)	Account ID

type	VARCHAR(10)	CHECK (type IN ('Credit','Debit'))	Transaction Type
amount	DECIMAL(10,2)	CHECK (amount > 0)	Transaction Amount
transaction_date	DATETIME	DEFAULT CURRENT_TIMESTAMP	Transaction Date

Sample Data to be Inserted

accounts

account_id	holder_name	account_type	balance
1001	Ravi Shankar	Savings	25000.0
1002	Priya Malhotra	Current	75000.0
1003	Sanjay Patel	Savings	35000.0

transactions

transaction_id	account_id	type	amount	transaction_date
1	1001	Credit	5000.0	2025-04-15 10:30:00
2	1002	Debit	15000.0	2025-04-15 11:45:00
3	1003	Credit	10000.0	2025-04-15 14:20:00

Experiment 10: Social Media

Tasks

1. Create tables with appropriate constraints: users and posts.

2. Insert sample data into all tables.

3. Update the email of one user.

4. Delete one post.

5. Write a query to display posts with user information.

6. Write a query to show total posts per user.

7. Create users for moderator and admin with appropriate privileges.

8. Drop all tables.

Table Structures

users

Column	Data Type	Constraints	Description
user_id	INT	PRIMARY KEY	User ID
username	VARCHAR(50)	UNIQUE NOT NULL	Username
email	VARCHAR(100)	UNIQUE NOT NULL	Email Address
join_date	DATE	NOT NULL	Join Date

posts

Column	Data Type	Constraints	Description
post_id	INT	PRIMARY KEY	Post ID
user_id	INT	FOREIGN KEY REFERENCES users(user_id)	User ID
content	VARCHAR(255)	NOT NULL	Post Content
post_date	DATE	NOT NULL	Post Date

Sample Data to be Inserted

users

user_id	username	email	join_date
1	anil123	anil@gmail.com	2025-04-01
2	priya_s	priya@gmail.com	2025-04-02
3	amitk	amit@gmail.com	2025-04-03

posts

post_id	user_id	content	post_date
101	1	Hello World!	2025-04-10
102	2	Good morning!	2025-04-11
103	1	Enjoying the weather.	2025-04-12

Experiment 11: Movie Theatre

Tasks

1. Create tables with appropriate constraints: movies and shows.

2. Insert sample data into all tables.

3. Update the ticket price for one show.

4. Delete one movie record.

5. Write a query to display show details with movie information.

6. Write a query to show total shows per movie.

7. Create and execute a stored procedure to check show availability.

8. Drop all tables.

Table Structures

movies

Column	Data Type	Constraints	Description
movie_id	INT	PRIMARY KEY	Movie ID
title	VARCHAR(100)	NOT NULL	Movie Title
duration_mins	INT	CHECK (duration_mins > 0)	Duration in Minutes
language	VARCHAR(20)	NOT NULL	Movie Language

shows

Column	Data Type	Constraints	Description
show_id	INT	PRIMARY KEY	Show ID
movie_id	INT	FOREIGN KEY REFERENCES movies(movie_id)	Movie ID
show_date	DATE	NOT NULL	Show Date
show_time	TIME	NOT NULL	Show Time

seats_available	INT	CHECK (seats_available >= 0)	Available Seats
ticket_price	DECIMAL(6,2)	CHECK (ticket_price > 0)	Ticket Price

Sample Data to be Inserted

movies

movie_id	title	duration_mins	language
1	Pathaan	150	Hindi
2	RRR	180	Telugu
3	KGF Chapter 2	168	Kannada

shows

show_id	movie_id	show_date	show_time	seats_available	ticket_price
101	1	2025-04-15	14:30:00	120	250.0
102	2	2025-04-15	18:00:00	150	300.0
103	3	2025-04-15	21:30:00	100	280.0

Experiment 12: Logistics

Tasks

1. Create tables with appropriate constraints: vehicles and deliveries.

2. Insert sample data into all tables.

3. Update the capacity of one vehicle.

4. Delete one delivery record.

5. Write a query to display delivery details with vehicle information.

6. Write a query to show total deliveries per vehicle.

7. Create and use a function to check if a vehicle can carry a given weight.

8. Drop all tables.

Table Structures

vehicles

Column	Data Type	Constraints	Description
vehicle_id	INT	PRIMARY KEY	Vehicle ID
vehicle_no	VARCHAR(20)	UNIQUE NOT NULL	Vehicle Number
type	VARCHAR(30)	NOT NULL	Vehicle Type
capacity	INT	CHECK (capacity > 0)	Capacity (kg)

deliveries

Column	Data Type	Constraints	Description
delivery_id	INT	PRIMARY KEY	Delivery ID
vehicle_id	INT	FOREIGN KEY REFERENCES vehicles(vehicle_id)	Vehicle ID
destination	VARCHAR(100)	NOT NULL	Destination
weight	INT	CHECK (weight > 0)	Weight (kg)

delivery_date	DATE	NOT NULL	Delivery Date

Sample Data to be Inserted

vehicles

vehicle_id	vehicle_no	type	capacity
1	MH12AB1234	Truck	10000
2	DL8CAF4321	Van	2000
3	KA05CD5678	Mini Truck	5000

deliveries

delivery_id	vehicle_id	destination	weight	delivery_date
101	1	Pune	8000	2025-04-15
102	2	Delhi	1500	2025-04-16
103	3	Bangalore	4000	2025-04-17

Experiment 13: Restaurant Orders

Tasks

1. Create tables with appropriate constraints: menu and orders.

2. Insert sample data into all tables.

3. Update the price of one menu item.

4. Delete one order record.

5. Write a query to display order details with menu information.

6. Write a query to show total orders per menu item.

7. Create users for cashier and manager with appropriate privileges.

8. Drop all tables.

Table Structures

menu

Column	Data Type	Constraints	Description
item_id	INT	PRIMARY KEY	Item ID
item_name	VARCHAR(100)	NOT NULL	Item Name
price	DECIMAL(6,2)	CHECK (price > 0)	Item Price
category	VARCHAR(50)	NOT NULL	Category

orders

Column	Data Type	Constraints	Description
order_id	INT	PRIMARY KEY	Order ID
item_id	INT	FOREIGN KEY REFERENCES menu(item_id)	Item ID
customer_name	VARCHAR(100)	NOT NULL	Customer Name
order_date	DATE	NOT NULL	Order Date

quantity	INT	CHECK (quantity > 0)	Quantity Ordered

Sample Data to be Inserted

menu

item_id	item_name	price	category
1	Paneer Butter Masala	220.0	Main Course
2	Masala Dosa	90.0	Breakfast
3	Gulab Jamun	60.0	Dessert

orders

order_id	item_id	customer_name	order_date	quantity
101	1	Rohit Sharma	2025-04-15	2
102	2	Anjali Mehta	2025-04-16	3
103	3	Vikas Gupta	2025-04-17	1

Experiment 14: Sports League

Tasks

1. Create tables with appropriate constraints: teams and matches.

2. Insert sample data into all tables.

3. Update the city for one team.

4. Delete one match record.

5. Write a query to display match details with team names.

6. Write a query to show total matches played by each team.

7. Create and execute a stored procedure to get matches played at a specific venue.

8. Drop all tables.

Table Structures

teams

Column	Data Type	Constraints	Description
team_id	INT	PRIMARY KEY	Team ID
team_name	VARCHAR(100)	NOT NULL	Team Name
city	VARCHAR(50)	NOT NULL	City

matches

Column	Data Type	Constraints	Description
match_id	INT	PRIMARY KEY	Match ID
team1_id	INT	FOREIGN KEY REFERENCES teams(team_id)	Team 1 ID
team2_id	INT	FOREIGN KEY REFERENCES teams(team_id)	Team 2 ID
match_date	DATE	NOT NULL	Match Date

venue	VARCHAR(100)	NOT NULL	Venue
winner_id	INT	FOREIGN KEY REFERENCES teams(team_id)	Winner Team ID

Sample Data to be Inserted

teams

team_id	team_name	city
1	Mumbai Indians	Mumbai
2	Chennai Super Kings	Chennai
3	Royal Challengers Bangalore	Bangalore

matches

match_id	team1_id	team2_id	match_date	venue	winner_id
101	1	2	2025-04-15	Wankhede Stadium	1
102	2	3	2025-04-16	Chepauk Stadium	2
103	1	3	2025-04-17	Chinnaswamy Stadium	3

Experiment 15: Fitness Center

Tasks

1. Create tables with appropriate constraints: members and attendance.

2. Insert sample data into all tables.

3. Update the fee paid for one member.

4. Delete one attendance record.

5. Write a query to display attendance details with member information.

6. Write a query to show total attendance per member.

7. Create and use a function to calculate workout duration.

8. Drop all tables.

Table Structures

members

Column	Data Type	Constraints	Description
member_id	INT	PRIMARY KEY	Member ID
name	VARCHAR(100)	NOT NULL	Member Name
join_date	DATE	NOT NULL	Join Date
membership_type	VARCHAR(20)	CHECK (membership_type IN ('Monthly','Quarterly','Yearly'))	Membership Type
fee_paid	DECIMAL(8,2)	CHECK (fee_paid > 0)	Fee Paid

attendance

Column	Data Type	Constraints	Description
attendance_id	INT	PRIMARY KEY	Attendance ID
member_id	INT	FOREIGN KEY REFERENCES members(member_id)	Member ID
check_in	DATETIME	NOT NULL	Check-in Time

check_out	DATETIME		Check-out Time
workout_type	VARCHAR(50)	NOT NULL	Type of Workout

Sample Data to be Inserted

members

member_id	name	join_date	membership_type	fee_paid
1	Karan Malhotra	2025-01-15	Monthly	2000.0
2	Priya Desai	2025-02-01	Quarterly	5500.0
3	Rahul Singh	2025-03-10	Yearly	20000.0

attendance

attendance_id	member_id	check_in	check_out	workout_type
101	1	2025-04-15 06:30:00	2025-04-15 07:45:00	Cardio
102	2	2025-04-15 08:00:00	2025-04-15 09:30:00	Weight Training
103	3	2025-04-15 17:00:00	2025-04-15 18:15:00	Yoga