MINOR PROJECT (SET-2)

Ques1:
Create relational database for Amity University, Noida Parking area in basement. Take the entity and their attributes as per need.

Ans:
Entities and Attributes:

1. User:
   - UserID (Primary Key)
   - Name
   - Role (Student, Staff, Visitor)
   - ContactNumber

2. Vehicle:
   - VehicleID (Primary Key)
   - LicenseNumber
   - Type (Car, Bike)
   - Color
   - OwnerID (Foreign Key to User)

3. ParkingSlot:
   - SlotID (Primary Key)
   - SlotNumber
   - Location (e.g., Basement Level 1, Basement Level 2)
   - Status (Available, Occupied)

4. ParkingTicket:
   - TicketID (Primary Key)
   - IssueTime
   - ExitTime
   - VehicleID (Foreign Key to Vehicle)
   - SlotID (Foreign Key to ParkingSlot)

Relationships:
- A User can own one or more Vehicles.
- A Vehicle is parked in one ParkingSlot at a time, tracked via a ParkingTicket.
- A ParkingSlot can have many ParkingTickets but only one active at a time.

Relational Database:

```
CREATE TABLE USER (
    USERID INT PRIMARY KEY,
    NAME VARCHAR(100),
    ROLE VARCHAR(50), -- 'STUDENT', 'STAFF', 'VISITOR'
    CONTACTNUMBER VARCHAR(15)
);
INSERT INTO USER (USERID, NAME, ROLE, CONTACTNUMBER) VALUES
(1, 'AMIT SINGH', 'STUDENT', '9876543210'),
(2, 'PRIYA SHARMA', 'STAFF', '9123456780'),
(3, 'RAJ MALHOTRA', 'VISITOR', '9988776655');
SELECT * FROM USER;
```

USERID	NAME	ROLE	CONTACTNUMBER
1	AMIT SINGH	STUDENT	9876543210
2	PRIYA SHARMA	STAFF	9123456780
3	RAJ MALHOTRA	VISITOR	9988776655

```
CREATE TABLE VEHICLE (
    VEHICLEID INT PRIMARY KEY,
    LICENSENUMBER VARCHAR(20) UNIQUE,
    TYPE VARCHAR(20), -- 'CAR', 'BIKE'
    COLOR VARCHAR(20),
    OWNERID INT, -- FOREIGN KEY TO USER
    FOREIGN KEY (OWNERID) REFERENCES USER(USERID)
);
INSERT INTO VEHICLE (VEHICLEID, LICENSENUMBER, TYPE, COLOR, OWNERID) VALUES
(1, 'UP16AB1234', 'CAR', 'RED', 1),
(2, 'DL12XY5678', 'BIKE', 'BLACK', 2),
(3, 'HR29KL7890', 'CAR', 'WHITE', 3);
SELECT * FROM VEHICLE;
```

VEHICLEID	LICENSENUMBER	TYPE	COLOR	OWNERID
1	UP16AB1234	CAR	RED	1
2	DL12XY5678	BIKE	BLACK	2
3	HR29KL7890	CAR	WHITE	3

```
CREATE TABLE PARKINGSLOT (
    SLOTID INT PRIMARY KEY,
    SLOTNUMBER VARCHAR(10),
    LOCATION VARCHAR(100), -- BASEMENT LEVEL 1, BASEMENT LEVEL 2, ETC.
    STATUS VARCHAR(20) -- 'AVAILABLE', 'OCCUPIED'
);
INSERT INTO PARKINGSLOT (SLOTID, SLOTNUMBER, LOCATION, STATUS) VALUES
(1, 'S1', 'BASEMENT LEVEL 1', 'OCCUPIED'),
(2, 'S2', 'BASEMENT LEVEL 1', 'AVAILABLE'),
(3, 'S3', 'BASEMENT LEVEL 2', 'AVAILABLE');
SELECT * FROM PARKINGSLOT;
```

SLOTID	SLOTNUMBER	LOCATION	STATUS
1	S1	BASEMENT LEVEL 1	OCCUPIED
2	S2	BASEMENT LEVEL 1	AVAILABLE
3	S3	BASEMENT LEVEL 2	AVAILABLE

```
CREATE TABLE PARKINGTICKET (
    TICKETID INT PRIMARY KEY,
    ISSUETIME DATETIME,
    EXITTIME DATETIME,
    VEHICLEID INT, -- FOREIGN KEY TO VEHICLE
    SLOTID INT, -- FOREIGN KEY TO PARKINGSLOT
    FOREIGN KEY (VEHICLEID) REFERENCES VEHICLE(VEHICLEID),
    FOREIGN KEY (SLOTID) REFERENCES PARKINGSLOT(SLOTID)
);
INSERT INTO PARKINGTICKET (TICKETID, ISSUETIME, EXITTIME, VEHICLEID, SLOTID)
VALUES
(1, '2024-10-15 08:00:00', NULL, 1, 1),
(2, '2024-10-15 09:30:00', NULL, 2, 2);
SELECT * FROM PARKINGTICKET;
```

TICKETID	ISSUETIME	EXITTIME	VEHICLEID	SLOTID
1	2024-10-15 08:00:00		1	1
2	2024-10-15 09:30:00		2	2

Ques2.

Perform at least 10 queries based on join, nested query, view.

1. Get all Vehicles with their Owners.
   - SELECT VEHICLE.LICENSENUMBER, VEHICLE.TYPE, VEHICLE.COLOR, USER.NAME AS OWNER
     FROM VEHICLE
     JOIN USER ON VEHICLE.OWNERID = USER.USERID;

LICENSENUMBER	TYPE	COLOR	Owner
UP16AB1234	CAR	RED	AMIT SINGH
DL12XY5678	BIKE	BLACK	PRIYA SHARMA
HR29KL7890	CAR	WHITE	RAJ MALHOTRA

2. Find all Available Parking Slots.
   - SELECT * FROM PARKINGSLOT WHERE STATUS = 'AVAILABLE';

SLOTID	SLOTNUMBER	LOCATION	STATUS
2	S2	BASEMENT LEVEL 1	AVAILABLE
3	S3	BASEMENT LEVEL 2	AVAILABLE

3. Get the Vehicles currently parked.
   - SELECT VEHICLE.LICENSENUMBER, PARKINGSLOT.SLOTNUMBER, PARKINGTICKET.ISSUETIME
     FROM PARKINGTICKET
     JOIN VEHICLE ON PARKINGTICKET.VEHICLEID = VEHICLE.VEHICLEID
     JOIN PARKINGSLOT ON PARKINGTICKET.SLOTID = PARKINGSLOT.SLOTID
     WHERE PARKINGTICKET.EXITTIME IS NULL;

LICENSENUMBER	SLOTNUMBER	ISSUETIME
UP16AB1234	S1	2024-10-15 08:00:00
DL12XY5678	S2	2024-10-15 09:30:00

4. Find all Vehicles owned by Staff.
   - SELECT VEHICLE.LICENSENUMBER, VEHICLE.TYPE, VEHICLE.COLOR
     FROM VEHICLE
     JOIN USER ON VEHICLE.OWNERID = USER.USERID
     WHERE USER.ROLE = 'STAFF';

LICENSENUMBER	TYPE	COLOR
DL12XY5678	BIKE	BLACK

5. Count the total number of Vehicles in the parking area.
   - SELECT COUNT(*) AS TOTALVEHICLES FROM VEHICLE;

TOTALVEHICLES
3

6. Create a view to display the current parking status of Vehicles.
   - CREATE VIEW CURRENTPARKINGSTATUS AS
     SELECT VEHICLE.LICENSENUMBER, PARKINGSLOT.SLOTNUMBER,
     PARKINGTICKET.ISSUETIME
     FROM PARKINGTICKET
     JOIN VEHICLE ON PARKINGTICKET.VEHICLEID = VEHICLE.VEHICLEID
     JOIN PARKINGSLOT ON PARKINGTICKET.SLOTID = PARKINGSLOT.SLOTID
     WHERE PARKINGTICKET.EXITTIME IS NULL;

7. Use the view to display all currently parked vehicles.
   - SELECT * FROM CURRENTPARKINGSTATUS;

LICENSENUMBER	SLOTNUMBER	ISSUETIME
UP16AB1234	S1	2024-10-15 08:00:00
DL12XY5678	S2	2024-10-15 09:30:00

8. Find the name of the User who owns the vehicle with LicenseNumber 'UP16AB1234'.
   - SELECT NAME FROM USER
     WHERE USERID = (SELECT OWNERID FROM VEHICLE WHERE LICENSENUMBER = 'UP16AB1234');

NAME
AMIT SINGH

9. Find vehicles that are not currently parked.
   - SELECT LICENSENUMBER FROM VEHICLE
     WHERE VEHICLEID NOT IN (SELECT VEHICLEID FROM PARKINGTICKET WHERE EXITTIME IS NULL);

LICENSENUMBER
HR29KL7890

10. Get the total number of available parking slots.
    - SELECT COUNT(*) AS AVAILABLESLOTS FROM PARKINGSLOT WHERE STATUS = 'AVAILABLE';

AvailableSlots
2