Savitribai Phule Pune University

S.Y. B.C.A. (Science) (Semester-III) Practical Examination

BCA 235: s(Database Management Systems II Laboratory)

Duration: 3Hrs. **Max Marks: 35+15=50**

Note: - 1. Read the questions carefully and insert data in the database accordingly.
2. Insert sufficient number of records in the database.
3. No query should generate empty output.
4. For count queries output should be more than 2 records. (If asked)

Create the following database in 3NF using PostgresSQL. [Total Marks: 10]

Q1) Consider a Railway Reservation System for passengers. The bogie capacity of all the bogies of atrain is same.

Train (Train_no int, train_name varchar (20), depart_time time , arrival_time time, source_stn varchar (20),dest_stn varchar (20), no_of_res_bogies int ,bogie_capacity int)

Passenger (Passenger_id int, passenger_name varchar (20), address varchar (30), age int, gender char)

Relationship:

Train _Passenger: M-M relationship named ticket with descriptive attributes as follows:

Ticket (Train_no int, Passenger_id int, Ticket_no int ,bogie_no int, no_of_berths int,tdate date, ticket_amt decimal (7, 2), ticket_status char)

Constraints: Primary key, ticket_status can be 'W' (waiting) or 'C' (confirmed).

Create a View: **[10]**

1. To display names of ‘Shatabdi Express’ passengers whose ticket status is waiting on 02-03-2022.
2. To display first three bookings for 'Rajdhani Express' on 04-05-2021.

Q.2) Using above database solve following questions: [Total Marks: 20]

1. Write a trigger to restrict the bogie capacity of any train to 25. [**10]**

2. Write a function using cursor to display train wise confirmed bookings on 19-04-2022. [**10]**

Q.3) External Viva **[05]**

Q.4) Internal Evaluation **[15]**

RAILWAY RESERVATION

CREATE TABLE Train (Train_no INTEGER PRIMARY KEY, train_name VARCHAR(20), depart_time TIME, arrival_time TIME, source_stn VARCHAR(20), dest_stn VARCHAR(20), no_of_res_bogies INTEGER, bogie_capacity INTEGER);

CREATE TABLE Passenger (Passenger_id INTEGER PRIMARY KEY, passenger_name VARCHAR(20), address VARCHAR(30), age INTEGER, gender CHAR(1));

CREATE TABLE Ticket (Ticket_no INTEGER PRIMARY KEY, Train_no INTEGER, Passenger_id INTEGER, bogie_no INTEGER, no_of_berths INTEGER, tdate DATE, ticket_amt DECIMAL(7, 2), ticket_status CHAR(1) CHECK (ticket_status IN ('W', 'C')), FOREIGN KEY (Train_no) REFERENCES Train (Train_no), FOREIGN KEY (Passenger_id) REFERENCES Passenger (Passenger_id));

INSERT INTO Train VALUES (101, 'Shatabdi Express', '08:00', '14:00', 'Mumbai', 'Delhi', 10, 72),(102, 'Rajdhani Express', '06:00', '12:00', 'Delhi', 'Chennai', 12, 70);

INSERT INTO Passenger VALUES(1, 'Rahul', 'Mumbai', 30, 'M'),(2, 'Anjali', 'Pune', 25, 'F'),(3, 'Amit', 'Delhi', 35, 'M'),(4, 'Priya', 'Bangalore', 28, 'F'),(5, 'Suresh', 'Hyderabad', 40, 'M');

INSERT INTO Ticket VALUES(1001, 101, 1, 1, 1, '2022-03-02', 1500.00, 'W'),(1002, 101, 2, 1, 1, '2022-03-02', 1500.00, 'C'),(1003, 101, 3, 1, 1, '2022-03-02', 1500.00, 'C'),(1004, 102, 4, 2, 1, '2021-05-04', 2000.00, 'C'),(1005, 102, 5, 2, 1, '2021-05-04', 2000.00, 'C'),(1006, 102, 1, 2, 1, '2021-05-04', 2000.00, 'W'),(1007, 102, 3, 2, 1, '2022-01-01', 2000.00, 'C');

Q.1) Create a View:

1. CREATE VIEW Shatabdi_Waiting AS SELECT P.passenger_name FROM Passenger P JOIN Ticket T ON P.Passenger_id = T.Passenger_id JOIN Train TR ON T.Train_no = TR.Train_no WHERE TR.train_name = 'Shatabdi Express' AND T.ticket_status = 'W' AND T.tdate = '2022-03-02';
   SELECT * FROM Shatabdi_Waiting;

2. CREATE VIEW Rajdhani_Bookings AS SELECT T.Ticket_no, P.passenger_name, T.ticket_amt FROM Passenger P JOIN Ticket T ON P.Passenger_id = T.Passenger_id JOIN Train TR ON T.Train_no = TR.Train_no WHERE TR.train_name = 'Rajdhani Express' AND T.tdate = '2021-05-04' ORDER BY T.Ticket_no LIMIT 3;
   SELECT * FROM Rajdhani_Bookings;

Q.2) Using above database solve following questions:

```
CREATE OR REPLACE FUNCTION restrict_bogie_capacity() RETURNS
TRIGGER AS $$
BEGIN
  IF NEW.bogie_capacity > 25 THEN
    RAISE EXCEPTION 'Bogie capacity cannot exceed 30';
  END IF;
  RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER check_bogie_capacity
BEFORE INSERT OR UPDATE ON Train
FOR EACH ROW EXECUTE FUNCTION restrict_bogie_capacity();

 INSERT INTO Train VALUES (103, 'Duronto Express', '09:00', '15:00',
'Kolkata', 'Mumbai', 12, 35);
```

```
CREATE OR REPLACE FUNCTION
display_train_wise_confirmed_bookings()
RETURNS VOID AS $$
DECLARE
  train_record RECORD;
  ticket_record RECORD;
  booking_date DATE := '2022-04-19';
BEGIN
  FOR train_record IN SELECT Train_no, train_name FROM Train
  LOOP
    RAISE NOTICE 'Train: %, Train Name: %', train_record.Train_no,
train_record.train_name;

    FOR ticket_record IN
      SELECT T.Ticket_no, P.passenger_name, T.ticket_amt
      FROM Ticket T
      JOIN Passenger P ON T.Passenger_id = P.Passenger_id
      WHERE T.Train_no = train_record.Train_no
      AND T.ticket_status = 'C'
      AND T.tdate = booking_date
    LOOP
      RAISE NOTICE 'Ticket No: %, Passenger: %, Amount: %',
ticket_record.Ticket_no, ticket_record.passenger_name,
ticket_record.ticket_amt;
    END LOOP;

  END LOOP;
END;
$$ LANGUAGE plpgsql;

SELECT display_train_wise_confirmed_bookings();
```