

DATABASE MANAGEMENT SYSTEM

UE22CS351A

DATE : 18-11-2024

JACKFRUIT PROBLEM

HACKATHON

TEAM MEMBERS :

1) NAME : PRERANA S NAIK

SRN : PES2UG22CS415

SEM : 5 SEC : G

2) NAME : PRINCIA DSOUZA

SRN : PES2UG22CS416

SEM : 5 SEC : G

3) NAME : RAAHHITHYA JAYARAM

SRN : PES2UG22CS421

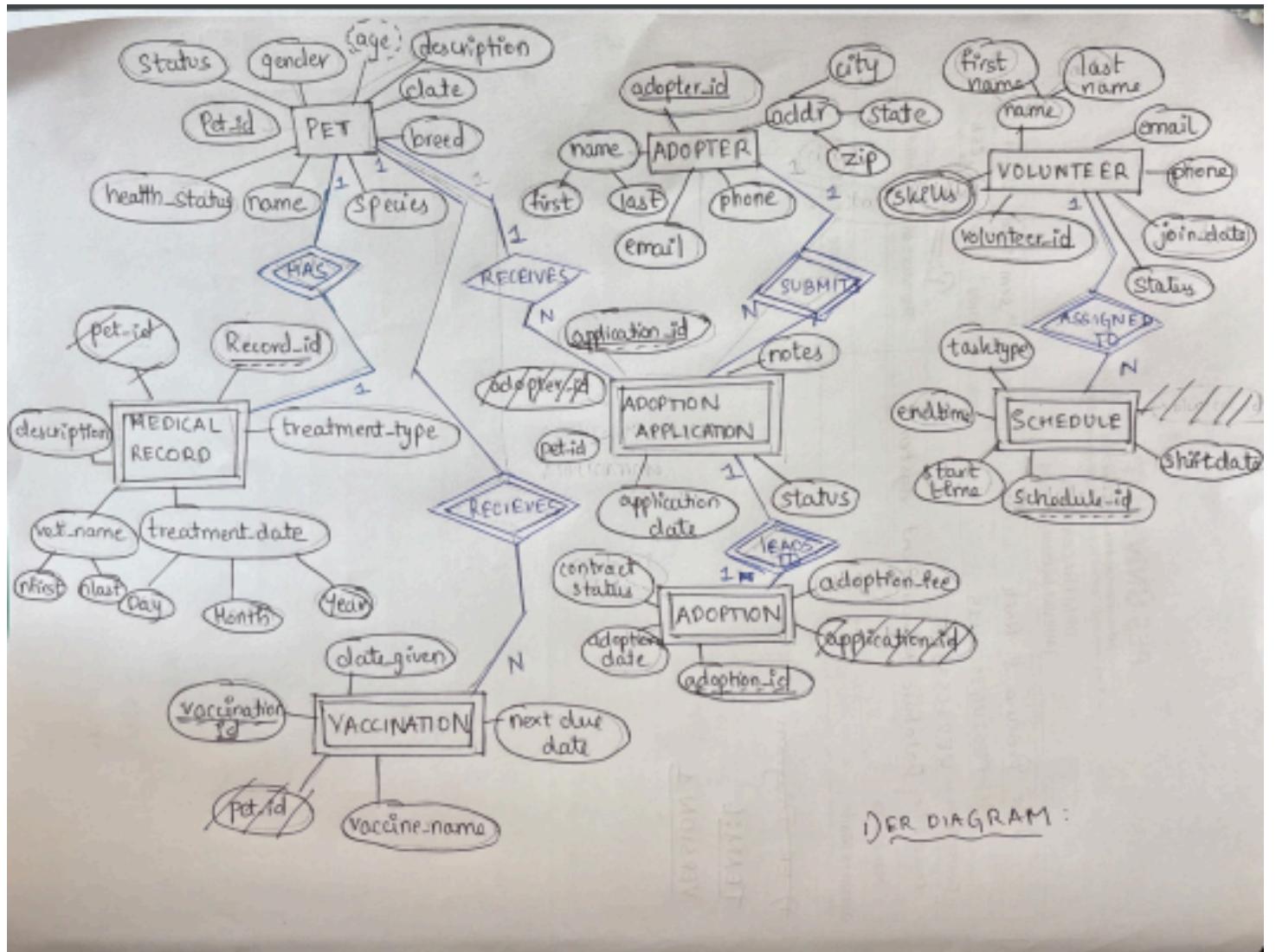
SEM : 5 SEC : G

4) NAME : RAASHI BAFNA

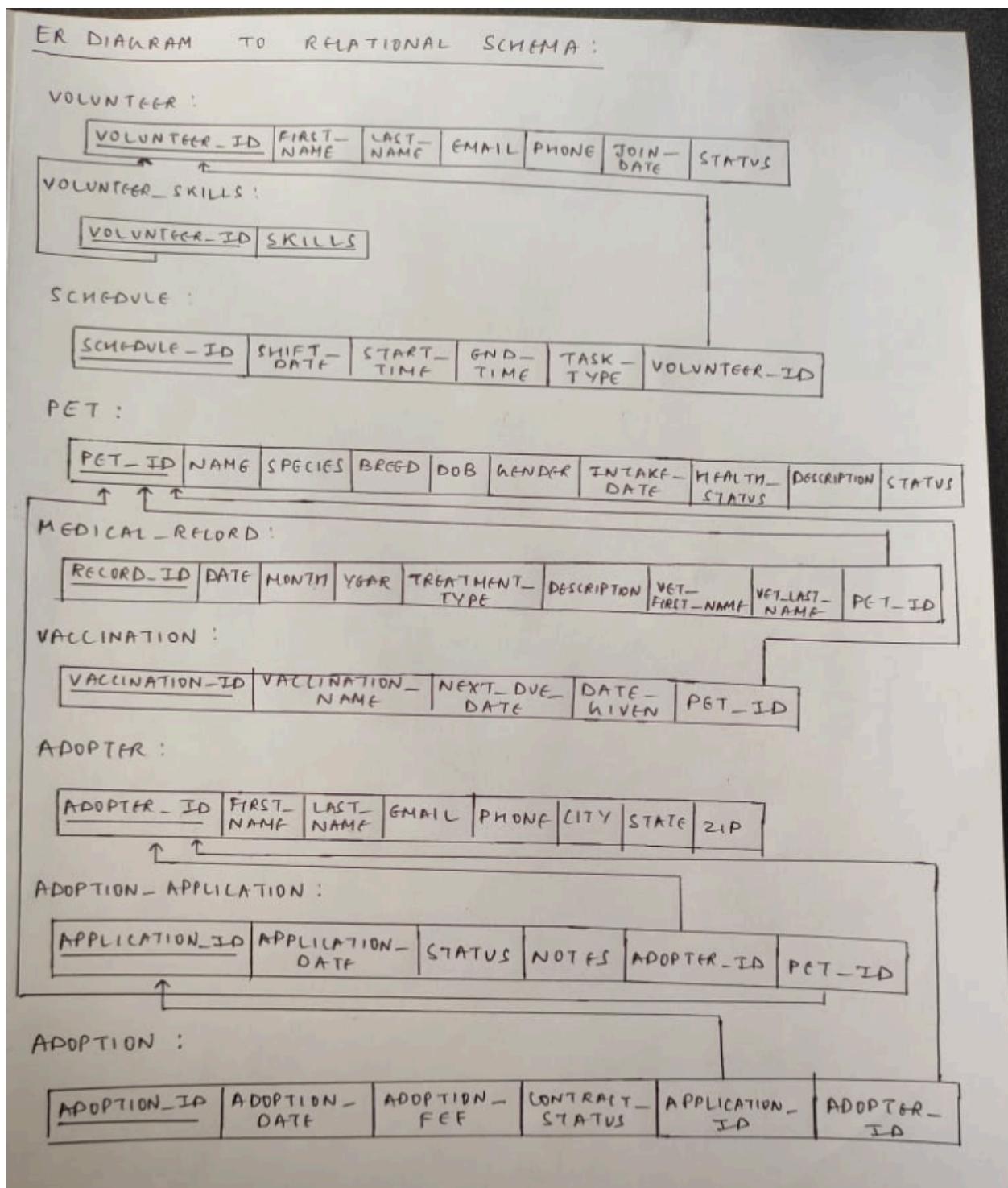
SRN : PES2UG22CS422

SEM : 5 SEC : G

Q1) ER Diagram:



Q2) Relational Schema:



Q3) Database Creation:

```
415_416_421_422 >CREATE DATABASE IF NOT EXISTS remote_animal_shelter_db;
Query OK, 1 row affected (0.01 sec)

415_416_421_422 >USE remote_animal_shelter_db;
Database changed

415_416_421_422 >CREATE TABLE IF NOT EXISTS Pet (
    ->     pet_id INT PRIMARY KEY NOT NULL,
    ->     name VARCHAR(50) NOT NULL,
    ->     breed VARCHAR(50),
    ->     age INT CHECK (age >= 0),
    ->     weight DECIMAL(5,2) CHECK (weight > 0),
    ->     health_condition VARCHAR(20) CHECK (health_condition IN ('Good', 'Fair', 'Poor', 'Needs Vaccination', 'Underweight')),
    ->     status VARCHAR(20) DEFAULT 'Available' CHECK (status IN ('Available', 'Adopted', 'In Review', 'High Demand')),
    ->     last_updated TIMESTAMP DEFAULT CURRENT_TIMESTAMP
    -> );
Query OK, 0 rows affected (0.03 sec)

415_416_421_422 >CREATE TABLE IF NOT EXISTS Adopter (
    ->     adopter_id INT PRIMARY KEY NOT NULL,
    ->     full_name VARCHAR(100) NOT NULL,
    ->     contact_info VARCHAR(100)
    -> );
Query OK, 0 rows affected (0.03 sec)

415_416_421_422 >CREATE TABLE IF NOT EXISTS Adoption_Application (
    ->     application_id INT PRIMARY KEY NOT NULL,
    ->     pet_id INT,
    ->     adopter_id INT,
    ->     status VARCHAR(20) DEFAULT 'Pending' CHECK (status IN ('Pending', 'Approved', 'Rejected')),
    ->     application_date DATE NOT NULL,
    ->     FOREIGN KEY (pet_id) REFERENCES Pet(pet_id),
    ->     FOREIGN KEY (adopter_id) REFERENCES Adopter(adopter_id)
    -> );
Query OK, 0 rows affected (0.05 sec)
```

```
415_416_421_422 >CREATE TABLE IF NOT EXISTS Adoption_Record (
-->     adoption_id INT PRIMARY KEY NOT NULL,
-->     pet_id INT,
-->     adopter_id INT,
-->     adoption_date DATE NOT NULL,
-->     FOREIGN KEY (pet_id) REFERENCES Pet(pet_id),
-->     FOREIGN KEY (adopter_id) REFERENCES Adopter(adopter_id)
--> );^C
415_416_421_422 >CREATE TABLE IF NOT EXISTS Volunteer (
-->     volunteer_id INT PRIMARY KEY NOT NULL,
-->     full_name VARCHAR(100) NOT NULL,
-->     contact_info VARCHAR(100),
-->     skills VARCHAR(255),
-->     availability VARCHAR(20) CHECK (availability IN ('Weekdays', 'Weekends', 'Flexible')),
-->     last_assigned_date DATE DEFAULT NULL
--> );
Query OK, 0 rows affected (0.03 sec)

415_416_421_422 >CREATE TABLE IF NOT EXISTS Volunteer_Schedule (
-->     schedule_id INT PRIMARY KEY NOT NULL,
-->     volunteer_id INT,
-->     shift_date DATE NOT NULL,
-->     task_description VARCHAR(255),
-->     FOREIGN KEY (volunteer_id) REFERENCES Volunteer(volunteer_id)
--> );
Query OK, 0 rows affected (0.06 sec)

415_416_421_422 >CREATE TABLE IF NOT EXISTS Volunteer_Audit (
-->     audit_id INT PRIMARY KEY NOT NULL,
-->     volunteer_id INT,
-->     shift_date DATE NOT NULL,
-->     update_timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
-->     FOREIGN KEY (volunteer_id) REFERENCES Volunteer(volunteer_id)
--> );
Query OK, 0 rows affected (0.04 sec)
```

```

415_416_421_422 >CREATE TABLE IF NOT EXISTS Adoption_Record (
->     adoption_id INT PRIMARY KEY NOT NULL,
->     pet_id INT,
->     adopter_id INT,
->     adoption_date DATE NOT NULL,
->     FOREIGN KEY (pet_id) REFERENCES Pet(pet_id),
->     FOREIGN KEY (adopter_id) REFERENCES Adopter(adopter_id)
-> );
Query OK, 0 rows affected (0.04 sec)

415_416_421_422 >CREATE USER IF NOT EXISTS 'fed-userP'@'%' IDENTIFIED BY 'fed-pswdP';
Query OK, 0 rows affected (0.02 sec)

415_416_421_422 >GRANT ALL PRIVILEGES ON remote_animal_shelter_db.* TO 'fed-userP'@'%';
Query OK, 0 rows affected (0.02 sec)

415_416_421_422 >FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)

415_416_421_422 >INSERT INTO Pet (pet_id, name, breed, age, weight, health_condition, status) VALUES
-> (1, 'Buddy', 'Golden Retriever', 3, 25.5, 'Good', 'Available'),
-> (2, 'Luna', 'Siberian Husky', 2, 20.2, 'Needs Vaccination', 'In Review'),
-> (3, 'Milo', 'Persian Cat', 4, 4.8, 'Good', 'Available'),
-> (4, 'Coco', 'Cockatiel', 1, 0.3, 'Underweight', 'Available'),
-> (5, 'Max', 'Labrador Retriever', 5, 28.0, 'Fair', 'Adopted'),
-> (6, 'Bella', 'Chihuahua', 2, 5.5, 'Poor', 'High Demand'),
-> (7, 'Charlie', 'Tabby Cat', 6, 5.2, 'Good', 'Available');

Query OK, 7 rows affected (0.02 sec)
Records: 7 Duplicates: 0 Warnings: 0

415_416_421_422 >INSERT INTO Adopter (adopter_id, full_name, contact_info) VALUES
-> (1, 'Rahul Sharma', 'rahul.sharma@example.com'),
-> (2, 'Priya Verma', 'priya.verma@example.com'),
-> (3, 'Anjali Gupta', 'anjali.gupta@example.com'),
-> (4, 'Amit Kumar', 'amit.kumar@example.com'),
-> (5, 'Neha Mehta', 'neha.mehta@example.com'),
-> (6, 'Rohan Joshi', 'rohan.joshi@example.com'),

```

```

415_416_421_422 >INSERT INTO Adoption_Application (application_id, pet_id, adopter_id, status, application_date) VALUES
-> (1, 1, 1, 'Pending', '2024-01-15'),
-> (2, 2, 2, 'Pending', '2024-01-16'),
-> (3, 3, 3, 'Approved', '2024-01-17'),
-> (4, 4, 4, 'Rejected', '2024-01-18'),
-> (5, 5, 5, 'Approved', '2024-01-19'),
-> (6, 6, 6, 'Pending', '2024-01-20'),
-> (7, 7, 7, 'Approved', '2024-01-21');
Query OK, 7 rows affected (0.01 sec)
Records: 7 Duplicates: 0 Warnings: 0

```

```

415_416_421_422 >INSERT INTO Adoption_Record (adoption_id, pet_id, adopter_id, adoption_date) VALUES
-> (1, 3, 3, '2024-02-01'),
-> (2, 5, 5, '2024-02-02'),
-> (3, 7, 7, '2024-02-03'),
-> (4, 2, 2, '2024-02-04'),
-> (5, 4, 4, '2024-02-05'),
-> (6, 6, 6, '2024-02-06'),
-> (7, 1, 1, '2024-02-07');
Query OK, 7 rows affected (0.00 sec)
Records: 7 Duplicates: 0 Warnings: 0

```

```

415_416_421_422 >INSERT INTO Volunteer (volunteer_id, full_name, contact_info, skills, availability, last_assigned_date) VALUES
-> (1, 'Arjun Nair', 'arjun.nair@example.com', 'Feeding, Grooming', 'Weekdays', '2024-01-10'),
-> (2, 'Kavya Desai', 'kavya.desai@example.com', 'Event Management', 'Weekends', '2024-01-11'),
-> (3, 'Rajiv Menon', 'rajiv.menon@example.com', 'Grooming, Cleaning', 'Flexible', '2024-01-12'),
-> (4, 'Pooja Reddy', 'pooja.reddy@example.com', 'Adoption Assistance', 'Weekdays', '2024-01-13'),
-> (5, 'Vikas Singh', 'vikas.singh@example.com', 'Feeding, Cleaning', 'Weekends', '2024-01-14'),
-> (6, 'Ishita Roy', 'ishita.roy@example.com', 'Event Management, Grooming', 'Flexible', '2024-01-15'),
-> (7, 'Manoj Patel', 'manoj.patel@example.com', 'Adoption Assistance', 'Weekends', '2024-01-16');
Query OK, 7 rows affected (0.01 sec)
Records: 7 Duplicates: 0 Warnings: 0

```

```
415_416_421_422 >INSERT INTO Volunteer_Schedule (schedule_id, volunteer_id, shift_date, task_description) VALUES
-> (1, 1, '2024-01-17', 'Morning Feeding'),
-> (2, 2, '2024-01-18', 'Event Setup'),
-> (3, 3, '2024-01-19', 'Grooming Session'),
-> (4, 4, '2024-01-20', 'Adoption Process Assistance'),
-> (5, 5, '2024-01-21', 'Cleaning Kennels'),
-> (6, 6, '2024-01-22', 'Adoption Follow-up'),
-> (7, 7, '2024-01-23', 'Evening Feeding');
Query OK, 7 rows affected (0.01 sec)
Records: 7  Duplicates: 0  Warnings: 0

415_416_421_422 >INSERT INTO Volunteer_Audit (audit_id, volunteer_id, shift_date, update_timestamp) VALUES
-> (1, 1, '2024-01-17', CURRENT_TIMESTAMP),
-> (2, 2, '2024-01-18', CURRENT_TIMESTAMP),
-> (3, 3, '2024-01-19', CURRENT_TIMESTAMP),
-> (4, 4, '2024-01-20', CURRENT_TIMESTAMP),
-> (5, 5, '2024-01-21', CURRENT_TIMESTAMP),
-> (6, 6, '2024-01-22', CURRENT_TIMESTAMP),
-> (7, 7, '2024-01-23', CURRENT_TIMESTAMP);
Query OK, 7 rows affected (0.02 sec)
Records: 7  Duplicates: 0  Warnings: 0
```

Q4) Federated Storage Engine for Synchronizing Adoption Records

LOCAL:

```
CREATE DATABASE IF NOT EXISTS local_animal_shelter_db;

USE local_animal_shelter_db;

CREATE TABLE IF NOT EXISTS Pet (
    pet_id INT PRIMARY KEY NOT NULL,
    name VARCHAR(50) NOT NULL,
    breed VARCHAR(50),
    age INT CHECK (age >= 0),
    weight DECIMAL(5,2) CHECK (weight > 0),
    health_condition VARCHAR(20) CHECK (health_condition IN ('Good', 'Fair', 'Poor', 'Needs Vaccination', 'Underweight')),
    status VARCHAR(20) DEFAULT 'Available' CHECK (status IN ('Available', 'Adopted', 'In Review', 'High Demand')),
    last_updated TIMESTAMP DEFAULT CURRENT_TIMESTAMP
)
ENGINE = FEDERATED
DEFAULT CHARSET = utf8mb4
CONNECTION = 'mysql://fed-userP:fed-pswdP@192.168.250.197:3306/remote_animal_shelter_db/Pet';
```

```
mysql> prompt 415_416_421_422 >
PROMPT set to '415_416_421_422 >'  
415_416_421_422 >show engines;  
+-----+-----+-----+-----+-----+-----+-----+  
| Engine | Support | Comment | Transactions | XA | Savepoints |  
+-----+-----+-----+-----+-----+-----+-----+  
| MEMORY | YES | Hash based, stored in memory, useful for temporary tables | NO | NO | NO |  
| MRG_MYISAM | YES | Collection of identical MyISAM tables | NO | NO | NO |  
| CSV | YES | CSV storage engine | NO | NO | NO |  
| FEDERATED | YES | Federated MySQL storage engine | NO | NO | NO |  
| PERFORMANCE_SCHEMA | YES | Performance Schema | NO | NO | NO |  
| MyISAM | YES | MyISAM storage engine | NO | NO | NO |  
| InnoDB | DEFAULT | Supports transactions, row-level locking, and foreign keys | YES | YES | YES |  
| ndbinfo | NO | MySQL Cluster system information storage engine | NULL | NULL | NULL |  
| BLACKHOLE | YES | /dev/null storage engine (anything you write to it disappears) | NO | NO | NO |  
| ARCHIVE | YES | Archive storage engine | NO | NO | NO |  
| ndbcluster | NO | Clustered, fault-tolerant tables | NULL | NULL | NULL |  
+-----+-----+-----+-----+-----+-----+-----+  
11 rows in set (0.01 sec)
```

REMOTE:

```
mysql> prompt 415_416_421_422 >
PROMPT set to '415_416_421_422 >'  
415_416_421_422 >show engines;  
+-----+-----+-----+-----+-----+-----+  
| Engine | Support | Comment | Transactions | XA | Savepoints |  
+-----+-----+-----+-----+-----+-----+  
| MEMORY | YES | Hash based, stored in memory, useful for temporary tables | NO | NO | NO |  
| MRG_MYISAM | YES | Collection of identical MyISAM tables | NO | NO | NO |  
| CSV | YES | CSV storage engine | NO | NO | NO |  
| FEDERATED | YES | Federated MySQL storage engine | NO | NO | NO |  
| PERFORMANCE_SCHEMA | YES | Performance Schema | NO | NO | NO |  
| MyISAM | YES | MyISAM storage engine | NO | NO | NO |  
| InnoDB | DEFAULT | Supports transactions, row-level locking, and foreign keys | YES | YES | YES |  
| ndbinfo | NO | MySQL Cluster system information storage engine | NULL | NULL | NULL |  
| BLACKHOLE | YES | /dev/null storage engine (anything you write to it disappears) | NO | NO | NO |  
| ARCHIVE | YES | Archive storage engine | NO | NO | NO |  
| ndbcluster | NO | Clustered, fault-tolerant tables | NULL | NULL | NULL |  
+-----+-----+-----+-----+-----+-----+  
11 rows in set (0.01 sec)
```

TESTING:

LOCAL:

```
INSERT INTO Adoption_Record (adoption_id, pet_id, adopter_id, adoption_date) VALUES (8, 3, 3, '2024-02-10');
```

REMOTE:

BEFORE:

```
415_416_421_422 >select * from Adoption_Record;  
+-----+-----+-----+-----+  
| adoption_id | pet_id | adopter_id | adoption_date |  
+-----+-----+-----+-----+  
| 1 | 3 | 3 | 2024-02-01 |  
| 2 | 5 | 5 | 2024-02-02 |  
| 3 | 7 | 7 | 2024-02-03 |  
| 4 | 2 | 2 | 2024-02-04 |  
| 5 | 4 | 4 | 2024-02-05 |  
| 6 | 6 | 6 | 2024-02-06 |  
| 7 | 1 | 1 | 2024-02-07 |  
+-----+-----+-----+-----+  
7 rows in set (0.02 sec)
```

AFTER:

```
415_416_421_422 >select * from Adoption_Record;
```

adoption_id	pet_id	adopter_id	adoption_date
1	3	3	2024-02-01
2	5	5	2024-02-02
3	7	7	2024-02-03
4	2	2	2024-02-04
5	4	4	2024-02-05
6	6	6	2024-02-06
7	1	1	2024-02-07
8	3	3	2024-02-10

```
8 rows in set (0.01 sec)
```

Q5) Recursive Query for Tracking Volunteer Shifts Over Consecutive Days

```
415_416_421_422 >select * from Volunteer;
+-----+-----+-----+-----+-----+-----+
| volunteer_id | full_name | contact_info | skills | availability | last_assigned_date |
+-----+-----+-----+-----+-----+-----+
| 1 | Arjun Nair | arjun.nair@example.com | Feeding, Grooming | Weekdays | 2024-01-10 |
| 2 | Kavya Desai | kavya.desai@example.com | Event Management | Weekends | 2024-01-11 |
| 3 | Rajiv Menon | rajiv.menon@example.com | Grooming, Cleaning | Flexible | 2024-01-12 |
| 4 | Pooja Reddy | pooja.reddy@example.com | Adoption Assistance | Weekdays | 2024-01-13 |
| 5 | Vikas Singh | vikas.singh@example.com | Feeding, Cleaning | Weekends | 2024-01-14 |
| 6 | Ishita Roy | ishita.roy@example.com | Event Management, Grooming | Flexible | 2024-01-15 |
| 7 | Manoj Patel | manoj.patel@example.com | Adoption Assistance | Weekends | 2024-01-16 |
| 8 | Neha Verma | neha.verma@example.com | Feeding, Grooming | Flexible | 2024-01-17 |
| 9 | Ramesh Kumar | ramesh.kumar@example.com | Event Management | Weekends | 2024-01-18 |
| 10 | Anita Sharma | anita.sharma@example.com | Adoption Assistance | Weekdays | 2024-01-19 |
| 11 | Suresh Gupta | suresh.gupta@example.com | Cleaning, Feeding | Flexible | 2024-01-20 |
| 12 | Priya Singh | priya.singh@example.com | Event Coordination | Weekdays | 2024-01-21 |
| 13 | Amit Desai | amit.desai@example.com | Adoption Assistance | Flexible | 2024-01-22 |
| 14 | Rekha Iyer | rekha.iyer@example.com | Grooming, Cleaning | Weekends | 2024-01-23 |
+-----+-----+-----+-----+-----+
14 rows in set (0.00 sec)

415_416_421_422 >select * from Volunteer_Schedule;
+-----+-----+-----+-----+
| schedule_id | volunteer_id | shift_date | task_description |
+-----+-----+-----+-----+
| 1 | 1 | 2024-01-17 | Morning Feeding |
| 2 | 2 | 2024-01-18 | Event Setup |
| 3 | 3 | 2024-01-19 | Grooming Session |
| 4 | 4 | 2024-01-20 | Adoption Process Assistance |
| 5 | 5 | 2024-01-21 | Cleaning Kennels |
| 6 | 6 | 2024-01-22 | Adoption Follow-up |
| 7 | 7 | 2024-01-23 | Evening Feeding |
| 8 | 8 | 2024-01-24 | Morning Feeding |
| 9 | 8 | 2024-01-25 | Evening Feeding |
| 10 | 9 | 2024-01-26 | Event Setup |
| 11 | 9 | 2024-01-28 | Event Coordination |
| 12 | 10 | 2024-01-29 | Adoption Process Assistance |
| 13 | 10 | 2024-01-30 | Adoption Follow-up |
| 14 | 12 | 2024-01-31 | Event Preparation |
+-----+-----+-----+-----+
```

Objective: Identify volunteers with consecutive shifts and display their details, including volunteer ID, name, starting date of the sequence, and consecutive shift dates.

Process:

- **Anchor Member:** Retrieves all individual shifts as potential starting points.

- **Recursive Member:** Finds shifts occurring exactly one day after the previous shift, extending the sequence for the same volunteer.
- **Final Filtering:** Removes single-day shifts and formats the result.

Output: Shows volunteers who worked on consecutive days, listing the sequence start date and each day in the sequence.

```
415_416_421_422 >WITH RECURSIVE ConsecutiveShifts AS (
    -- Anchor Member: Start with all individual shifts
    SELECT
        v.volunteer_id,
        v.full_name,
        vs.shift_date,
        vs.shift_date AS start_date
    FROM Volunteer v
    JOIN Volunteer_Schedule vs
    ON v.volunteer_id = vs.volunteer_id
    --
    UNION ALL
    --
    -- Recursive Member: Find next consecutive shift
    SELECT
        c.volunteer_id,
        c.full_name,
        vs.shift_date,
        c.start_date
    FROM ConsecutiveShifts c
    JOIN Volunteer_Schedule vs
    ON c.volunteer_id = vs.volunteer_id
    AND vs.shift_date = DATE_ADD(c.shift_date, INTERVAL 1 DAY)
)
SELECT DISTINCT
    volunteer_id,
    full_name,
    start_date,
    shift_date
FROM ConsecutiveShifts
WHERE shift_date != start_date
ORDER BY volunteer_id, start_date, shift_date;
+-----+-----+-----+-----+
| volunteer_id | full_name | start_date | shift_date |
+-----+-----+-----+-----+
|      8 | Neha Verma | 2024-01-24 | 2024-01-25 |
|     10 | Anita Sharma | 2024-01-29 | 2024-01-30 |
+-----+-----+-----+-----+
2 rows in set (0.01 sec)
```

Q6) Stored Procedure for Adoption Process

STORED PROCEDURE:

```
415_416_421_422 >DELIMITER //
415_416_421_422 >
415_416_421_422 >CREATE PROCEDURE finalize_adoption(
->      IN p_application_id INT,
->      OUT p_status VARCHAR(100)
-> )
-> BEGIN
->     -- Declare variables
->     DECLARE v_pet_id INT;
->     DECLARE v_adopter_id INT;
->     DECLARE v_pet_status VARCHAR(20);
->     DECLARE v_application_status VARCHAR(20);
->     DECLARE v_next_adoption_id INT;
->
->     -- Declare handler for SQL exceptions
->     DECLARE EXIT HANDLER FOR SQLEXCEPTION
->     BEGIN
->         ROLLBACK;
->         SET p_status = 'Error: Transaction rolled back due to error';
->     END;
->
->     -- Start transaction
->     START TRANSACTION;
->
->     -- Get application details
->     SELECT pet_id, adopter_id, status
->     INTO v_pet_id, v_adopter_id, v_application_status
->     FROM Adoption_Application
->     WHERE application_id = p_application_id;
->
->     -- Check if application exists
->     IF v_pet_id IS NULL THEN
->         SET p_status = 'Error: Invalid application ID';
->         ROLLBACK;
->     ELSE
->         -- Check application status
->         IF v_application_status != 'Approved' THEN
->             SET p_status = 'Error: Application is not approved';
```

```

ELSE
    -- Check application status
    IF v_application_status != 'Approved' THEN
        SET p_status = 'Error: Application is not approved';
        ROLLBACK;
    ELSE
        -- Check current pet status
        SELECT status INTO v_pet_status
        FROM Pet
        WHERE pet_id = v_pet_id;

        IF v_pet_status = 'Adopted' THEN
            SET p_status = 'Error: Pet is already adopted';
            ROLLBACK;
        ELSE
            -- Get next adoption ID
            SELECT COALESCE(MAX(adoption_id) + 1, 1)
            INTO v_next_adoption_id
            FROM Adoption_Record;

            -- Update pet status
            UPDATE Pet
            SET status = 'Adopted',
                last_updated = CURRENT_TIMESTAMP
            WHERE pet_id = v_pet_id;

            -- Create adoption record
            INSERT INTO Adoption_Record (
                adoption_id,
                pet_id,
                adopter_id,
                adoption_date
            ) VALUES (
                v_next_adoption_id,
                v_pet_id,
                v_adopter_id,
                CURDATE()
            );
        END IF;
    END IF;
END IF;

-- Update application status

```

```

->      ;
->
->      -- Update application status
->      UPDATE Adoption_Application
->      SET status = 'Approved'
->      WHERE application_id = p_application_id;
->
->      COMMIT;
->      SET p_status = 'Success: Adoption finalized successfully';
->      END IF;
->      END IF;
->      END IF;
-> END //;

Query OK, 0 rows affected (0.01 sec)

415_416_421_422 >
415_416_421_422 >DELIMITER ;

```

QUERY:

```
415_416_421_422 >SET @status = '';
Query OK, 0 rows affected (0.03 sec)

415_416_421_422 >CALL finalize_adoption(3, @status);
Query OK, 0 rows affected (0.01 sec)

415_416_421_422 >SELECT @status;
+-----+
| @status           |
+-----+
| Success: Adoption finalized successfully |
+-----+
1 row in set (0.00 sec)

415_416_421_422 >CALL finalize_adoption(3, @status);
Query OK, 0 rows affected (0.00 sec)

415_416_421_422 >SELECT @status;
+-----+
| @status           |
+-----+
| Error: Pet is already adopted |
+-----+
1 row in set (0.00 sec)

415_416_421_422 >CALL finalize_adoption(999, @status);
Query OK, 0 rows affected (0.00 sec)

415_416_421_422 >SELECT @status;
+-----+
| @status           |
+-----+
| Error: Invalid application ID |
+-----+
1 row in set (0.00 sec)
```

```
415_416_421_422 >CALL finalize_adoption(1, @status);
Query OK, 0 rows affected (0.00 sec)

415_416_421_422 >SELECT @status;
+-----+
| @status           |
+-----+
| Error: Application is not approved |
+-----+
1 row in set (0.00 sec)
```

PET

BEFORE:

```
415_416_421_422 >select * from Pet;
+-----+-----+-----+-----+-----+-----+-----+-----+
| pet_id | name   | breed      | age    | weight | health_condition | status  | last_updated |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1     | Buddy   | Golden Retriever | 3     | 25.50  | Good          | Available | 2024-11-18 10:43:11 |
| 2     | Luna    | Siberian Husky  | 2     | 20.20  | Needs Vaccination | In Review | 2024-11-18 10:43:11 |
| 3     | Milo    | Persian Cat     | 4     | 4.80   | Good          | Available | 2024-11-18 10:43:11 |
| 4     | Coco    | Cockatiel       | 1     | 0.30   | Underweight    | Available | 2024-11-18 10:43:11 |
| 5     | Max     | Labrador Retriever | 5     | 28.00  | Fair           | Adopted   | 2024-11-18 10:43:11 |
| 6     | Bella   | Chihuahua       | 2     | 5.50   | Poor           | High Demand | 2024-11-18 10:43:11 |
| 7     | Charlie | Tabby Cat       | 6     | 5.20   | Good          | Available | 2024-11-18 10:43:11 |
+-----+-----+-----+-----+-----+-----+-----+-----+
7 rows in set (0.11 sec)

415_416_421_422 >
```

AFTER:

```
415_416_421_422 >select * from pet;
+-----+-----+-----+-----+-----+-----+-----+-----+
| pet_id | name   | breed      | age    | weight | health_condition | status  | last_updated |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1     | Buddy   | Golden Retriever | 3     | 25.50  | Good          | Available | 2024-11-18 10:58:10 |
| 2     | Luna    | Siberian Husky  | 2     | 20.20  | Needs Vaccination | In Review | 2024-11-18 10:58:10 |
| 3     | Milo    | Persian Cat     | 4     | 4.80   | Good          | Adopted   | 2024-11-18 12:25:24 |
| 4     | Coco    | Cockatiel       | 1     | 0.30   | Underweight    | Available | 2024-11-18 10:58:10 |
| 5     | Max     | Labrador Retriever | 5     | 28.00  | Fair           | Adopted   | 2024-11-18 10:58:10 |
| 6     | Bella   | Chihuahua       | 2     | 5.50   | Poor           | High Demand | 2024-11-18 10:58:10 |
| 7     | Charlie | Tabby Cat       | 6     | 5.20   | Good          | Available | 2024-11-18 10:58:10 |
+-----+-----+-----+-----+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

ADOPTION_RECORD

BEFORE:

```
415_416_421_422 >select * from Adoption_Record;
+-----+-----+-----+-----+
| adoption_id | pet_id | adopter_id | adoption_date |
+-----+-----+-----+-----+
| 1           | 3       | 3          | 2024-02-01   |
| 2           | 5       | 5          | 2024-02-02   |
| 3           | 7       | 7          | 2024-02-03   |
| 4           | 2       | 2          | 2024-02-04   |
| 5           | 4       | 4          | 2024-02-05   |
| 6           | 6       | 6          | 2024-02-06   |
| 7           | 1       | 1          | 2024-02-07   |
+-----+-----+-----+-----+
7 rows in set (0.01 sec)

415_416_421_422 >
```

AFTER:

415_416_421_422 >select * from Adoption_Record;			
adoption_id	pet_id	adopter_id	adoption_date
1	3	3	2024-02-01
2	5	5	2024-02-02
3	7	7	2024-02-03
4	2	2	2024-02-04
5	4	4	2024-02-05
6	6	6	2024-02-06
7	1	1	2024-02-07
9	3	3	2024-11-18

8 rows in set (0.00 sec)

Objective:

Finalize a pet's adoption by updating its status, creating an adoption record, and ensuring consistency using a transaction.

Process:

1. Input Validation:

- Fetch application details.
- Check if the application exists, is approved, and the pet is not already adopted.

2. Transaction Steps:

- Mark the pet as "Adopted."
- Insert a new adoption record.
- Update the application status to "Approved."

- Commit changes if successful; otherwise, roll back.

3. Error Handling:

- Handle invalid application IDs, non-approved applications, or already adopted pets with appropriate messages.
-

Output:

Returns success or error messages such as "Adoption finalized successfully" or "Pet is already adopted."

Q7) Trigger for Enhancing Volunteer Schedule Updates and Coordination

```
415_416_421_422 >DELIMITER //
415_416_421_422 >
415_416_421_422 >CREATE TRIGGER update_last_assigned_date
-> AFTER INSERT ON Volunteer_Schedule
-> FOR EACH ROW
-> BEGIN
->     -- Update the volunteer's last assigned date
->     UPDATE Volunteer
->     SET last_assigned_date = NEW.shift_date
->     WHERE volunteer_id = NEW.volunteer_id;
->
->     -- Log the update in the Volunteer_Audit table
->     INSERT INTO Volunteer_Audit (
->         audit_id,
->         volunteer_id,
->         shift_date,
->         update_timestamp
->     )
->     VALUES (
->         (SELECT COALESCE(MAX(audit_id), 0) + 1 FROM Volunteer_Audit va),
->         NEW.volunteer_id,
->         NEW.shift_date,
->         CURRENT_TIMESTAMP
->     );
-> END //
415_416_421_422 >
415_416_421_422 >DELIMITER ;
```

Query OK, 0 rows affected (0.04 sec)

```
415_416_421_422 >INSERT INTO Volunteer_Schedule (
->     schedule_id,
->     volunteer_id,
->     shift_date,
->     task_description
-> ) VALUES (
->     9,    -- New schedule_id
->     2,    -- volunteer_id for Kavya Desai
->     '2024-02-16',  -- New shift date
->     'Weekend Event Coordination'
-> );
Query OK, 1 row affected (0.03 sec)
```

VOLUNTEER:

BEFORE

```
415_416_421_422 >select * from Volunteer;
+-----+-----+-----+-----+-----+-----+-----+
| volunteer_id | full_name | contact_info | skills | availability | last_assigned_date |
+-----+-----+-----+-----+-----+-----+
| 1 | Arjun Nair | arjun.nair@example.com | Feeding, Grooming | Weekdays | 2024-02-15 |
| 2 | Kavya Desai | kavya.desai@example.com | Event Management | Weekends | 2024-01-11 |
| 3 | Rajiv Menon | rajiv.menon@example.com | Grooming, Cleaning | Flexible | 2024-01-12 |
| 4 | Pooja Reddy | pooja.reddy@example.com | Adoption Assistance | Weekdays | 2024-01-13 |
| 5 | Vikas Singh | vikas.singh@example.com | Feeding, Cleaning | Weekends | 2024-01-14 |
| 6 | Ishita Roy | ishita.roy@example.com | Event Management, Grooming | Flexible | 2024-01-15 |
| 7 | Manoj Patel | manoj.patel@example.com | Adoption Assistance | Weekends | 2024-01-16 |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

AFTER

```
415_416_421_422 >select * from Volunteer;
+-----+-----+-----+-----+-----+-----+
| volunteer_id | full_name | contact_info | skills | availability | last_assigned_date |
+-----+-----+-----+-----+-----+-----+
| 1 | Arjun Nair | arjun.nair@example.com | Feeding, Grooming | Weekdays | 2024-02-15 |
| 2 | Kavya Desai | kavya.desai@example.com | Event Management | Weekends | 2024-02-16 |
| 3 | Rajiv Menon | rajiv.menon@example.com | Grooming, Cleaning | Flexible | 2024-01-12 |
| 4 | Pooja Reddy | pooja.reddy@example.com | Adoption Assistance | Weekdays | 2024-01-13 |
| 5 | Vikas Singh | vikas.singh@example.com | Feeding, Cleaning | Weekends | 2024-01-14 |
| 6 | Ishita Roy | ishita.roy@example.com | Event Management, Grooming | Flexible | 2024-01-15 |
| 7 | Manoj Patel | manoj.patel@example.com | Adoption Assistance | Weekends | 2024-01-16 |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

AUDIT:

BEFORE

```
415_416_421_422 >select * from Volunteer_Audit;
+-----+-----+-----+-----+
| audit_id | volunteer_id | shift_date | update_timestamp |
+-----+-----+-----+-----+
| 1 | 1 | 2024-01-17 | 2024-11-18 10:59:24 |
| 2 | 2 | 2024-01-18 | 2024-11-18 10:59:24 |
| 3 | 3 | 2024-01-19 | 2024-11-18 10:59:24 |
| 4 | 4 | 2024-01-20 | 2024-11-18 10:59:24 |
| 5 | 5 | 2024-01-21 | 2024-11-18 10:59:24 |
| 6 | 6 | 2024-01-22 | 2024-11-18 10:59:24 |
| 7 | 7 | 2024-01-23 | 2024-11-18 10:59:24 |
| 8 | 1 | 2024-02-15 | 2024-11-18 12:15:35 |
+-----+-----+-----+-----+
8 rows in set (0.00 sec)
```

AFTER:

```
415_416_421_422 >select * from Volunteer_Audit;
+-----+-----+-----+-----+
| audit_id | volunteer_id | shift_date | update_timestamp |
+-----+-----+-----+-----+
| 1 | 1 | 2024-01-17 | 2024-11-18 10:59:24 |
| 2 | 2 | 2024-01-18 | 2024-11-18 10:59:24 |
| 3 | 3 | 2024-01-19 | 2024-11-18 10:59:24 |
| 4 | 4 | 2024-01-20 | 2024-11-18 10:59:24 |
| 5 | 5 | 2024-01-21 | 2024-11-18 10:59:24 |
| 6 | 6 | 2024-01-22 | 2024-11-18 10:59:24 |
| 7 | 7 | 2024-01-23 | 2024-11-18 10:59:24 |
| 8 | 1 | 2024-02-15 | 2024-11-18 12:15:35 |
| 9 | 2 | 2024-02-16 | 2024-11-18 12:18:21 |
+-----+-----+-----+-----+
9 rows in set (0.00 sec)
```

SCHEDULE

BEFORE:

```
415_416_421_422 >select * from Volunteer_Schedule;
+-----+-----+-----+-----+
| schedule_id | volunteer_id | shift_date | task_description |
+-----+-----+-----+-----+
| 1 | 1 | 2024-01-17 | Morning Feeding |
| 2 | 2 | 2024-01-18 | Event Setup |
| 3 | 3 | 2024-01-19 | Grooming Session |
| 4 | 4 | 2024-01-20 | Adoption Process Assistance |
| 5 | 5 | 2024-01-21 | Cleaning Kennels |
| 6 | 6 | 2024-01-22 | Adoption Follow-up |
| 7 | 7 | 2024-01-23 | Evening Feeding |
| 8 | 1 | 2024-02-15 | Evening Feeding and Care |
+-----+-----+-----+-----+
8 rows in set (0.00 sec)
```

AFTER:

```
415_416_421_422 >select * from Volunteer_Schedule;
+-----+-----+-----+-----+
| schedule_id | volunteer_id | shift_date | task_description |
+-----+-----+-----+-----+
| 1 | 1 | 2024-01-17 | Morning Feeding |
| 2 | 2 | 2024-01-18 | Event Setup |
| 3 | 3 | 2024-01-19 | Grooming Session |
| 4 | 4 | 2024-01-20 | Adoption Process Assistance |
| 5 | 5 | 2024-01-21 | Cleaning Kennels |
| 6 | 6 | 2024-01-22 | Adoption Follow-up |
| 7 | 7 | 2024-01-23 | Evening Feeding |
| 8 | 1 | 2024-02-15 | Evening Feeding and Care |
| 9 | 2 | 2024-02-16 | Weekend Event Coordination |
+-----+-----+-----+-----+
9 rows in set (0.00 sec)
```

GUI:

before:

The screenshot shows the 'Volunteer Scheduler' application window divided into three main sections: 'Volunteers', 'Schedule Volunteer', and 'Audit Log'.

- Volunteers:** A table listing volunteers with their ID, Name, and Last Assigned date.
- Schedule Volunteer:** A form for scheduling a volunteer. It includes fields for 'Selected Volunteer' (set to 'None'), 'Shift Date' (set to '2024-11-18'), and 'Task' (empty). A 'Schedule Volunteer' button is present.
- Audit Log:** A table showing a history of scheduled tasks, including ID, Volunteer, Shift Date, and Updated timestamp.

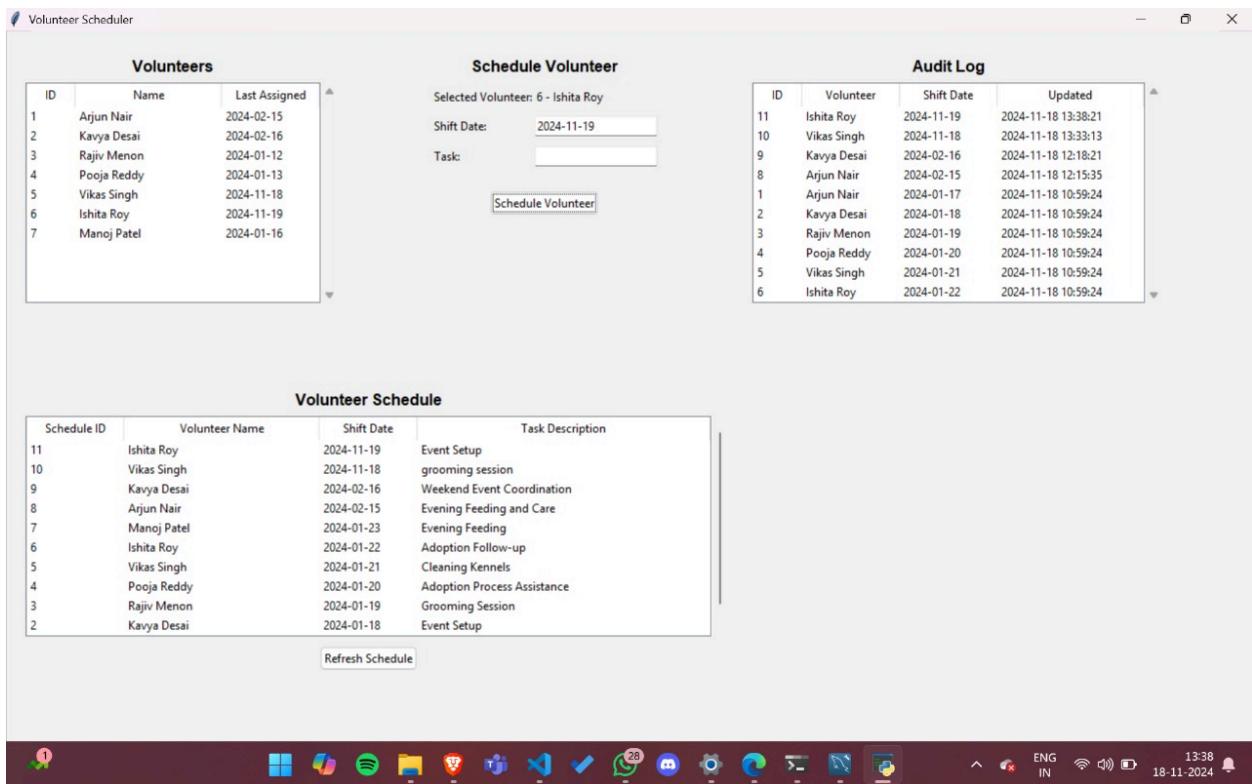
Below these sections is a larger 'Volunteer Schedule' table displaying the current list of scheduled tasks. At the bottom of the screen, a taskbar shows various system icons and the date/time (18-11-2024, 13:37).

ID	Name	Last Assigned
1	Arjun Nair	2024-02-15
2	Kavya Desai	2024-02-16
3	Rajiv Menon	2024-01-12
4	Pooja Reddy	2024-01-13
5	Vikas Singh	2024-11-18
6	Ishita Roy	2024-01-15
7	Manoj Patel	2024-01-16

ID	Volunteer	Shift Date	Updated
10	Vikas Singh	2024-11-18	2024-11-18 13:33:13
9	Kavya Desai	2024-02-16	2024-11-18 12:18:21
8	Arjun Nair	2024-02-15	2024-11-18 12:15:35
1	Arjun Nair	2024-01-17	2024-11-18 10:59:24
2	Kavya Desai	2024-01-18	2024-11-18 10:59:24
3	Rajiv Menon	2024-01-19	2024-11-18 10:59:24
4	Pooja Reddy	2024-01-20	2024-11-18 10:59:24
5	Vikas Singh	2024-01-21	2024-11-18 10:59:24
6	Ishita Roy	2024-01-22	2024-11-18 10:59:24
7	Manoj Patel	2024-01-23	2024-11-18 10:59:24

Schedule ID	Volunteer Name	Shift Date	Task Description
10	Vikas Singh	2024-11-18	grooming session
9	Kavya Desai	2024-02-16	Weekend Event Coordination
8	Arjun Nair	2024-02-15	Evening Feeding and Care
7	Manoj Patel	2024-01-23	Evening Feeding
6	Ishita Roy	2024-01-22	Adoption Follow-up
5	Vikas Singh	2024-01-21	Cleaning Kennels
4	Pooja Reddy	2024-01-20	Adoption Process Assistance
3	Rajiv Menon	2024-01-19	Grooming Session
2	Kavya Desai	2024-01-18	Event Setup
1	Arjun Nair	2024-01-17	Morning Feeding

After:



Objective:

To ensure that whenever a volunteer's schedule is updated, the `last_assigned_date` field in the `Volunteer` table is automatically updated, and the changes are logged into the `Volunteer_Audit` table for tracking purposes.

Process:

1. Trigger Activation:

- The trigger is executed **after** a new entry is inserted into the **Volunteer_Schedule** table.

2. Volunteer Table Update:

- The trigger updates the `last_assigned_date` column in the `Volunteer` table to reflect the `shift_date` from the new record in `Volunteer_Schedule`.

3. Audit Logging:

- A new record is inserted into the `Volunteer_Audit` table.
- This record includes:
 - An `audit_id` (incremented value based on the maximum existing `audit_id`).
 - The `volunteer_id` associated with the update.
 - The new `shift_date` and the exact timestamp of the update.

Output:

1. The `Volunteer` table shows updated `last_assigned_date` values for volunteers whose schedules have been modified.
2. The `Volunteer_Audit` table tracks all changes for accountability, including the update timestamp and the affected volunteer's details.

Q8) User-Defined Function for Health Assessment and Monitoring of Pets

FUNCTION:

```
415_416_421_422 >DELIMITER $$  
415_416_421_422 >  
415_416_421_422 >CREATE FUNCTION calculate_health_score(pet_age INT, pet_weight DECIMAL(5,2), pet_condition VARCHAR(20))  
-> RETURNS DECIMAL(10,2)  
-> DETERMINISTIC  
-> BEGIN  
->     DECLARE condition_rating INT;  
->  
->     -- Assign condition ratings based on the health condition  
->     CASE pet_condition  
->         WHEN 'Good' THEN SET condition_rating = 10;  
->         WHEN 'Fair' THEN SET condition_rating = 5;  
->         WHEN 'Poor' THEN SET condition_rating = 2;  
->         ELSE SET condition_rating = 1; -- Default rating for undefined conditions  
->     END CASE;  
->  
->     -- Prevent division by zero  
->     IF pet_age = 0 THEN  
->         RETURN 0;  
->     END IF;  
->  
->     -- Calculate the health score  
->     RETURN (pet_weight / pet_age) * condition_rating;  
-> END$$  
Query OK, 0 rows affected (0.02 sec)  
  
415_416_421_422 >  
415_416_421_422 >DELIMITER ;
```

QUERY:

```
415_416_421_422 >DELIMITER ;  
415_416_421_422 >SELECT  
    ->     pet_id,  
    ->     name,  
    ->     age,  
    ->     weight,  
    ->     health_condition,  
    ->     calculate_health_score(age, weight, health_condition) AS health_score  
    -> FROM ...  
    -> Pet;
```

OUTPUT:

pet_id	name	age	weight	health_condition	health_score
1	Buddy	3	25.50	Good	85.00
2	Luna	2	20.20	Needs Vaccination	10.10
3	Milo	4	4.80	Good	12.00
4	Coco	1	0.30	Underweight	0.30
5	Max	5	28.00	Fair	28.00
6	Bella	2	5.50	Poor	5.50
7	Charlie	6	5.20	Good	8.67

7 rows in set, 1 warning (0.02 sec)

```

415_416_421_422 >SELECT calculate_health_score(5, 25.5, 'Good') AS health_score;
+-----+
| health_score |
+-----+
|      51.00 |
+-----+
1 row in set (0.00 sec)

415_416_421_422 >SELECT calculate_health_score(3, 18.0, 'Fair') AS health_score;
+-----+
| health_score |
+-----+
|      30.00 |
+-----+
1 row in set (0.00 sec)

```

Process: The code creates a function `calculate_health_score` that determines a pet's health score based on three parameters:

- `pet_age` (integer)
- `pet_weight` (decimal)
- `pet_condition` (varchar/string)

Output: As shown in the query results:

- Pets in "Good" condition (like Buddy) get higher scores
- Scores vary based on the weight-to-age ratio and condition
- Example outputs:
 - Good condition: 81.00 (for 25.5 weight)
 - Fair condition: 30.00 (for 18.9 weight)
 - The function considers both physical metrics (weight/age) and health status

Safety Features:

- DETERMINISTIC declaration ensures consistent results
- Zero-age check prevents calculation errors
- Decimal precision (5,2) for weight and (10,2) for final score ensures accuracy

The function provides a standardized way to evaluate pet health based on multiple factors, useful for veterinary clinics or pet care facilities to track animal wellness.

Q9) Demonstrate Transaction Isolation Levels

SESSION1

```
415_416_421_422 >-- Insert a new pet record with pet_id = 13
415_416_421_422 >INSERT INTO Pet (pet_id, name, breed, age, weight, health_condition, status)
-> VALUES (13, 'Zara', 'Beagle', 3, 15.0, 'Good', 'Available');
Query OK, 1 row affected (0.01 sec)
```

```
415_416_421_422 >select * from Pet;
+-----+-----+-----+-----+-----+-----+-----+-----+
| pet_id | name   | breed      | age    | weight  | health_condition | status   | last_updated |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1     | Buddy  | Golden Retriever | 3     | 25.50  | Good       | Available | 2024-11-18 10:43:11 |
| 2     | Luna   | Siberian Husky  | 2     | 20.20  | Needs Vaccination | In Review | 2024-11-18 10:43:11 |
| 3     | Milo   | Persian Cat     | 4     | 4.80   | Good       | Available  | 2024-11-18 10:43:11 |
| 4     | Coco   | Cockatiel       | 1     | 0.30   | Underweight | Available  | 2024-11-18 10:43:11 |
| 5     | Max    | Labrador Retriever | 5     | 28.00  | Fair       | Adopted   | 2024-11-18 10:43:11 |
| 6     | Bella  | Chihuahua       | 2     | 5.50   | Poor       | High Demand | 2024-11-18 10:43:11 |
| 7     | Charlie | Tabby Cat     | 6     | 5.20   | Good       | Available  | 2024-11-18 10:43:11 |
| 8     | Rocky  | Boxer         | 2     | 18.50  | Good       | Adopted   | 2024-11-18 13:45:54 |
| 9     | Bella  | Bulldog        | 3     | 25.00  | Good       | Adopted   | 2024-11-18 13:48:29 |
| 11    | Rocky  | Rottweiler     | 4     | 35.00  | Good       | In Review | 2024-11-18 14:03:41 |
| 13    | Zara   | Beagle        | 3     | 15.00  | Good       | Available  | 2024-11-18 14:09:30 |
+-----+-----+-----+-----+-----+-----+-----+-----+
11 rows in set (0.00 sec)

415_416_421_422 >
```

SESSION1

```
415_416_421_422 >-- Start a new session (Session 1) and set the isolation level to Read Committed
415_416_421_422 >SET TRANSACTION ISOLATION LEVEL READ COMMITTED;
Query OK, 0 rows affected (0.00 sec)

415_416_421_422 >
415_416_421_422 >-- Start a new transaction
415_416_421_422 >START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

415_416_421_422 >
415_416_421_422 >-- Update the status of pet_id = 13 to 'Adopted'
415_416_421_422 >UPDATE Pet
-> SET status = 'Adopted'
-> WHERE pet_id = 13;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

415_416_421_422 >
415_416_421_422 >-- Do not commit yet, leave the transaction open.
415_416_421_422 >-- This update is not visible to other sessions until committed.
415_416_421_422 >
```

SESSION2

```
415_416_421_422 >-- Start a new session (Session 2) and set the isolation level to Read Committed
415_416_421_422 >SET TRANSACTION ISOLATION LEVEL READ COMMITTED;
Query OK, 0 rows affected (0.00 sec)

415_416_421_422 >
415_416_421_422 >-- Start a new transaction
415_416_421_422 >START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

415_416_421_422 >
415_416_421_422 >-- Attempt to update the status of pet_id = 13 to 'In Review'
415_416_421_422 >UPDATE Pet
    -> SET status = 'In Review'
    -> WHERE pet_id = 13;
|
```

SESSION1

```
415_416_421_422 >-- Start a new transaction
415_416_421_422 >START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

415_416_421_422 >
415_416_421_422 >-- Update the status of pet_id = 13 to 'Adopted'
415_416_421_422 >UPDATE Pet
    -> SET status = 'Adopted'
    -> WHERE pet_id = 13;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

415_416_421_422 >
415_416_421_422 >-- Do not commit yet, leave the transaction open.
415_416_421_422 >-- This update is not visible to other sessions until committed.
415_416_421_422 >select * from Pet;
+-----+-----+-----+-----+-----+-----+-----+
| pet_id | name      | breed          | age   | weight | health_condition | status        | last_updated |
+-----+-----+-----+-----+-----+-----+-----+
| 1     | Buddy     | Golden Retriever | 3    | 25.50  | Good           | Available    | 2024-11-18 10:43:11 |
| 2     | Luna      | Siberian Husky  | 2    | 20.20  | Needs Vaccination | In Review   | 2024-11-18 10:43:11 |
| 3     | Milo      | Persian Cat     | 4    | 4.80   | Good           | Available    | 2024-11-18 10:43:11 |
| 4     | Coco      | Cockatiel       | 1    | 0.30   | Underweight    | Available    | 2024-11-18 10:43:11 |
| 5     | Max       | Labrador Retriever | 5    | 28.00  | Fair            | Adopted      | 2024-11-18 10:43:11 |
| 6     | Bella     | Chihuahua       | 2    | 5.50   | Poor            | High Demand | 2024-11-18 10:43:11 |
| 7     | Charlie   | Tabby Cat       | 6    | 5.20   | Good           | Available    | 2024-11-18 10:43:11 |
| 8     | Rocky     | Boxer           | 2    | 18.50  | Good           | Adopted      | 2024-11-18 13:45:54 |
| 9     | Bella     | Bulldog         | 3    | 25.00  | Good           | Adopted      | 2024-11-18 13:48:29 |
| 11    | Rocky     | Rottweiler      | 4    | 35.00  | Good           | In Review   | 2024-11-18 14:03:41 |
| 13    | Zara      | Beagle          | 3    | 15.00  | Good           | Adopted      | 2024-11-18 14:09:30 |
+-----+-----+-----+-----+-----+-----+-----+
11 rows in set (0.00 sec)

415_416_421_422 >|
```

SESSION2

```
415_416_421_422 >-- Start a new session (Session 2) and set the isolation level to Read Committed
415_416_421_422 >SET TRANSACTION ISOLATION LEVEL READ COMMITTED;
Query OK, 0 rows affected (0.00 sec)

415_416_421_422 >
415_416_421_422 >-- Start a new transaction
415_416_421_422 >START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

415_416_421_422 >
415_416_421_422 >-- Attempt to update the status of pet_id = 13 to 'In Review'
415_416_421_422 >UPDATE Pet
    -> SET status = 'In Review'
    -> WHERE pet_id = 13;
Query OK, 1 row affected (40.88 sec)
Rows matched: 1  Changed: 1  Warnings: 0

415_416_421_422 >
415_416_421_422 >-- Commit the changes in Session 2
415_416_421_422 >COMMIT;
Query OK, 0 rows affected (0.01 sec)

415_416_421_422 >
```

SESSION1

```
415_416_421_422 >-- This update is not visible to other sessions until committed.
415_416_421_422 >select * from Pet;
+-----+-----+-----+-----+-----+-----+-----+-----+
| pet_id | name   | breed      | age   | weight | health_condition | status     | last_updated |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1     | Buddy   | Golden Retriever | 3     | 25.50  | Good          | Available  | 2024-11-18 10:43:11 |
| 2     | Luna    | Siberian Husky  | 2     | 20.20  | Needs Vaccination | In Review | 2024-11-18 10:43:11 |
| 3     | Milo    | Persian Cat     | 4     | 4.80   | Good          | Available  | 2024-11-18 10:43:11 |
| 4     | Coco    | Cockatiel      | 1     | 0.30   | Underweight   | Available  | 2024-11-18 10:43:11 |
| 5     | Max     | Labrador Retriever | 5     | 28.00  | Fair          | Adopted    | 2024-11-18 10:43:11 |
| 6     | Bella   | Chihuahua      | 2     | 5.50   | Poor          | High Demand | 2024-11-18 10:43:11 |
| 7     | Charlie | Tabby Cat     | 6     | 5.20   | Good          | Available  | 2024-11-18 10:43:11 |
| 8     | Rocky   | Boxer         | 2     | 18.50  | Good          | Adopted    | 2024-11-18 13:45:54 |
| 9     | Bella   | Bulldog        | 3     | 25.00  | Good          | Adopted    | 2024-11-18 13:48:29 |
| 11    | Rocky   | Rottweiler     | 4     | 35.00  | Good          | In Review  | 2024-11-18 14:03:41 |
| 13    | Zara    | Beagle        | 3     | 15.00  | Good          | Adopted    | 2024-11-18 14:09:30 |
+-----+-----+-----+-----+-----+-----+-----+-----+
11 rows in set (0.00 sec)

415_416_421_422 >commit;
Query OK, 0 rows affected (0.01 sec)

415_416_421_422 >select * from Pet;
+-----+-----+-----+-----+-----+-----+-----+-----+
| pet_id | name   | breed      | age   | weight | health_condition | status     | last_updated |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1     | Buddy   | Golden Retriever | 3     | 25.50  | Good          | Available  | 2024-11-18 10:43:11 |
| 2     | Luna    | Siberian Husky  | 2     | 20.20  | Needs Vaccination | In Review | 2024-11-18 10:43:11 |
| 3     | Milo    | Persian Cat     | 4     | 4.80   | Good          | Available  | 2024-11-18 10:43:11 |
| 4     | Coco    | Cockatiel      | 1     | 0.30   | Underweight   | Available  | 2024-11-18 10:43:11 |
| 5     | Max     | Labrador Retriever | 5     | 28.00  | Fair          | Adopted    | 2024-11-18 10:43:11 |
| 6     | Bella   | Chihuahua      | 2     | 5.50   | Poor          | High Demand | 2024-11-18 10:43:11 |
| 7     | Charlie | Tabby Cat     | 6     | 5.20   | Good          | Available  | 2024-11-18 10:43:11 |
| 8     | Rocky   | Boxer         | 2     | 18.50  | Good          | Adopted    | 2024-11-18 13:45:54 |
| 9     | Bella   | Bulldog        | 3     | 25.00  | Good          | Adopted    | 2024-11-18 13:48:29 |
| 11    | Rocky   | Rottweiler     | 4     | 35.00  | Good          | In Review  | 2024-11-18 14:03:41 |
| 13    | Zara    | Beagle        | 3     | 15.00  | Good          | In Review  | 2024-11-18 14:09:30 |
+-----+-----+-----+-----+-----+-----+-----+-----+
11 rows in set (0.00 sec)

415_416_421_422 >
```

Process: This MySQL code demonstrates a transaction isolation scenario with READ COMMITTED level, focusing on:

- How multiple database sessions handle concurrent updates
- When transactions can see each other's changes
- How commit order affects final data state

Function Components:

1. Session Setup:

- Multiple sessions access same data
- Each uses READ COMMITTED isolation
- Transactions start independently

2. Transaction Rules:

- Sessions only see committed data
- Updates are isolated until commit
- Last commit typically wins

Output:

- Final data state depends on commit order
- Each transaction executes independently
- Changes become visible only after commit

Key Behavior:

- Prevents dirty reads
- Maintains data consistency
- Handles concurrent access effectively

This pattern is useful for multi-user databases where data integrity and consistent views of data are important.