

# Project Phase 2

Team 025

March 15, 2025

## Contents

<b>1</b>	<b>Abstract Code + SQL</b>	<b>2</b>
1.1	Login . . . . .	2
1.2	Add Dog . . . . .	2
1.3	View Dog . . . . .	4
1.4	Update Dog Information . . . . .	5
1.5	Add Expenses . . . . .	7
1.6	Adoption Review . . . . .	7
1.7	Add Application . . . . .	9
1.8	Search Applications . . . . .	10
1.9	Set Adoption Date . . . . .	12
1.10	View Dog Dashboard . . . . .	12
1.11	View Animal Control Report . . . . .	14
1.12	View Adoption Report . . . . .	18
1.13	View Expense Analysis Report . . . . .	20
1.14	View Volunteer Lookup Report . . . . .	20
1.15	View Volunteer Birthdays Report . . . . .	20

# 1 Abstract Code + SQL

## 1.1 Login

- User enters *emailAddress*, *password* input fields.
- When User clicks **Enter** button

```
SELECT v.email
FROM Volunteer v
JOIN User u ON v.email = u.email
WHERE v.email = @emailAddress
      AND u.password = @password;
```

- Validate *emailAddress* and *password* are not blank.
  - If *emailAddress* is found, validate *password*.
  - If validation is true, then (table from query is not empty):
    - \* Store User object as instance that can be accessed across the application
    - \* Check if user is Director and save the result in User object.
- ```
SELECT 1 FROM Director dir
WHERE dir.email = @emailAddress;
```
- \* Navigate to ***Dog Dashboard*** form.
  - Else:
    - \* Display error message on ***Login*** form.

## 1.2 Add Dog

- Display form with inputs if **currentNumberOfDogs+1** is less than or equal to **maxCapacity**:

```
INSERT INTO Dog (email,name,description,age,sex,altered,
surrender_date,surrender_phone,by_animal_control)
VALUES (@email, @dogName, @description, @age, @sex,
@altered, @surrenderDate, @surrenderPhone,
@byAnimalControl);
```

```
SELECT LAST_INSERT_ID();
```

- Store result into dogID variable
- For each breed insert into DogBreed table:

```
INSERT INTO DogBreed VALUES (@dogID, @breedName);
```

```
INSERT INTO DogMicrochip Values (@dogID, @microchipID);
```

- **dogName**: text box
- **breeds**: multi-select dropdown
  - \* Multiple selections allowed
  - \* If “Unknown” OR “Mixed” selected:
    - Disable other selections
- **sex**: radio button
  - \* “Male”
  - \* “Female”
  - \* “Unknown”
- **altered**: radio button
  - \* “Yes”
  - \* “No”
- **age**: two input boxes
  - \* **year**: converted into months (default value: 0)
  - \* **months**
- **description**: text box input (optional)
- **manufacturer**: dropdown (nullable)
- **microChipID**: dropdown
  - \* Disabled until a manufacturer is chosen (default value: null)
  - \* Dropdown values update once a manufacturer is selected
  - \* Nullable, must be unique per dog
- **surrenderDate**: calendar input
- **surrenderPhone**: number input
  - \* If **byAnimalControl** is true, input is required
  - \* Else, optional
- **byAnimalControl**: radio button
  - \* “Yes”
  - \* “No”
- Verify if dog is bulldog breed and named “Uga”
  - If true - display error: name not allowed
- Upon:
  - Click **Submit** Button: display options **Dog Details** or **Return to Dashboard** links
    - \* Click **Dog Details** → Jump to the **Dog Detail** task
    - \* Click **Return to Dashboard** → Jump to the **Dog Dashboard** task

### 1.3 View Dog

- Display dog profile with:
- dogID is retrieved from angular route url

```
SELECT
  d.dogID,
  d.name,
  COALESCE(GROUP_CONCAT(DISTINCT b.name
                        ORDER BY b.name SEPARATOR '/'), 'Unknown')
    AS breeds,
  d.sex,
  d.altered,
  FLOOR(d.age / 12) AS age_years,
  d.age % 12 AS age_months,
  d.description,
  dm.microchipID,
  m.manufacturer AS microchipVendor,
  d.surrender_date,
  d.surrender_phone,
  d.by_animal_control
FROM Dog d
LEFT JOIN DogBreed db ON d.dogID = db.dogID
LEFT JOIN Breed b ON db.name = b.name
LEFT JOIN DogMicrochip dm ON d.dogID = dm.dogID
LEFT JOIN Microchip m ON dm.microchipID = m.microchipID
WHERE d.dogID = @dogID;
```

1

- **dogName**
- **breeds** (forward slash-separated in alphabetical if multiple)
- **sex**
- **altered** status
- **age** (displayed in years and months)
- **description** (if available)
- **manufacturer** (if applicable)
- **microChipID** (if applicable)
- **surrenderDate**
- **surrenderPhone** (if applicable)
- **byAnimalControl** status

---

<sup>1</sup>GROUP\_CONCAT() is a function in MySQL. See [https://www.geeksforgeeks.org/mysql-group\\_concat-function/](https://www.geeksforgeeks.org/mysql-group_concat-function/)

- Display expenses section:

```
SELECT e.category, SUM(e.amount) AS totalExpense
FROM Expense e
WHERE e.dogID = @dogID
GROUP BY e.category
ORDER BY totalExpense DESC;
```

```
SELECT SUM(e.amount) AS grandTotal
FROM Expense e
WHERE e.dogID = @dogID;
```

- \* **totalExpense** per vendor
- \* Add all **totalExpense** and display as **grandTotal**

- If user.age is greater than 18:

```
SELECT v.email
FROM Volunteer v
WHERE TIMESTAMPDIFF(YEAR, v.birth_date, CURDATE())
>= 18 AND v.email = @user.emailAddress;
```

<sup>2</sup> <sup>3</sup>

- \* Display **Enter New Expense** link

- If **user** is Director and dog.**microChipID** is not NULL and dog.**altered** is true

```
SELECT d.dogID, d.name, d.altered, dm.microchipID
FROM Dog d
LEFT JOIN DogMicrochip dm ON d.dogID = dm.dogID
WHERE d.dogID = @dogID
AND d.altered = TRUE
AND dm.microchipID IS NOT NULL;
```

- \* Display **Add Adoption** button

## 1.4 Update Dog Information

- Upon:
  - Click **Enter New Expense** link then jump to **Expenses** task
  - Click **Add Application** button then jump to **Add Application** task
- If a dog is adopted, disable everything. Check if user is at least 18 for options

<sup>2</sup>TIMESTAMPDIFF() is a function in MySQL See <https://www.geeksforgeeks.org/timestampdiff-function-in-mysql/>

<sup>3</sup>CURDATE() is a function in MySQL See [https://www.geeksforgeeks.org/mysql-group\\_concat-function/](https://www.geeksforgeeks.org/mysql-group_concat-function/)

```
SELECT a.dogID FROM Adoption a WHERE a.dogID = @dogID;
```

```
SELECT v.email
FROM Volunteer v
WHERE TIMESTAMPDIFF(YEAR, v.birth_date,
    CURDATE()) >= 18 AND v.email = @user.emailAddress;
```

4

- **dogName** (Disabled)
- **breeds** (If unknown or mixed, editable. Otherwise, disable)

```
DELETE FROM DogBreed
WHERE dogID = @dogID AND name IN ('Unknown', 'Mixed');
```

- For each breed insert into DogBreed table:

```
INSERT INTO DogBreed (dogID, name)
VALUES (@dogID, @breed);
```

- **sex** (If unknown, editable. Otherwise, disable)

```
UPDATE Dog d
SET sex = @sex
WHERE sex = 'UNKNOWN' AND d.dogID = @dogID;
```

- **altered** status (If false, editable. Otherwise, disable)

```
UPDATE Dog d
SET altered = @altered
WHERE altered = FALSE AND d.dogID = @dogID;
```

- **age** (disabled)
- **description** (disabled)
- **manufacturer** (If null and user.age greater than 18, editable. Update microChipID dropdown.)
- **microChipID** (If manufacturer is not null and user.age is greater than 18, editable.)

```
INSERT INTO Microchip (microchipID, manufacturer)
VALUES (@microChipID, @manufacturer);
```

```
INSERT INTO DogMicrochip (dogID, microchipID)
VALUES (@dogID, @microChipID);
```

<sup>4</sup>CURDATE() is a function in MySQL See [https://www.geeksforgeeks.org/mysql-group\\_concat-function/](https://www.geeksforgeeks.org/mysql-group_concat-function/)

- **surrenderDate** (disabled)
- **surrenderPhone** (disabled)
- **byAnimalControl** status (disabled)

## 1.5 Add Expenses

```
INSERT INTO Expense (dogID, vendor, date, amount, category)
SELECT @dogID, @vendorName, @expenseDate, @amount,
       @expenseCategory
WHERE (@expenseDate > (SELECT d.surrender_date
FROM Expense e
LEFT JOIN Dog d ON e.DogID = d.DogID
LEFT JOIN Adoption a ON e.DogID = a.DogID
WHERE e.DogId = @dogID
LIMIT 1 )) AND
(SELECT COALESCE(a.decision_date)
FROM Expense e
LEFT JOIN Dog d ON e.DogID = d.DogID
LEFT JOIN Adoption a ON e.DogID = a.DogID
WHERE e.DogId = @dogID LIMIT 1 ) IS NULL;
```

- Display form with inputs:
  - **expenseDate**: calendar input
  - **vendorName**: text box
  - **amount**: double
  - **expenseCategory**: dropdown
- Submission conditions:
  - If dog has not been adopted and dog is surrendered:
    - \* Allow submit
  - If (**expenseCategory** exists and **currentDate** is true) OR user.**age** is less than 18:
    - \* Do not allow submit

## 1.6 Adoption Review

- Display pending applications and applicant contact information.
- Display table containing:
  - Applications in **Pending** state.
  - Applicant contact details:

```
SELECT
a.email,
d.first_name,
d.last_name,
d.street,
d.city,
d.state,
d.zip,
d.phone
FROM Application a
LEFT JOIN Adopter d ON a.email = d.email;
```

\* **firstName**

\* **lastName**

\* **Address:**

· street

· city

· state

· zipCode

\* **phoneNumber**

\* **emailAddress**

– Buttons for:

\* **Approve**

```
INSERT INTO Adoption
(email, date, fee,
 is_fee_waived, decision_date, dogID)
SELECT
a.email,
a.date,
CASE
    WHEN d.name = 'Sideways'
    AND EXISTS (
        SELECT 1 FROM DogBreed db
        JOIN Breed b ON db.name = b.name
        WHERE db.dogID = d.dogID
        AND b.name LIKE '%Terrier%'
    )
    THEN 0
    WHEN d.by_animal_control = TRUE
    THEN
        ROUND(COALESCE(e.total_expense * 0.10, 0), 2)
    ELSE
        ROUND(COALESCE(e.total_expense * 1.25, 0), 2)
```



```
END AS fee,
CASE
  WHEN d.name = 'Sideways'
    AND EXISTS (
      SELECT 1 FROM DogBreed db
      JOIN Breed b ON db.name = b.name
      WHERE db.dogID = d.dogID
      AND b.name LIKE '%Terrier%'
    )
  THEN TRUE
  ELSE FALSE
END AS is_fee_waived,
@decDate,
@dogID
FROM Application a
JOIN Dog d ON d.dogID = @dogID
LEFT JOIN (
  SELECT d.dogID, SUM(e.amount) AS total_expense
  FROM Expense e
  JOIN Dog d ON e.dogID = d.dogID
  GROUP BY d.dogID
) e ON e.dogID = @dogID
WHERE a.email = @user.emailAddress AND a.date = @date;
```

```
DELETE FROM Application
WHERE email = @user.emailAddress
AND date = @date;
```

**\* Reject**

```
INSERT INTO Rejection (email, date)
SELECT email, a.date
FROM Application a
WHERE email = @user.emailAddress AND date = @date;
```

```
DELETE FROM Application
WHERE email = @user.emailAddress
AND date = @date;
```

## 1.7 Add Application

```
INSERT INTO Adopter
VALUES (@emailAddress, @firstName, @lastName, @phoneNumber,
      @street, @city, @state, @zipCode, @householdSize);
```

```
INSERT INTO Application
```

```
VALUES (@emailAddress, @applicationDate);
```

- Display form with inputs:
  - **firstName**: text box
  - **lastName**: text box
  - **street**: text box
  - **city**: text box
  - **state**: text box
  - **zipCode**: number input (numbers only)
  - **phoneNumber**: number input (numbers only)
  - **emailAddress**: text box
  - **householdSize**: number input (numbers only)
- **applicationStatus** will be set to "Pending" upon submission.
- Submission conditions:
  - If **emailAddress** and **applicationDate** exist:
    - \* Display error message indicating only one application per day is allowed.

## 1.8 Search Applications

- Display search input box:
  - Look up via applicant's **lastName** (case insensitive)
- Display list of applicants with:

```
SELECT ad.first_name,  
       ad.last_name,  
       ad.street,  
       ad.city,  
       ad.state,  
       ad.zip AS zip_code,  
       ad.phone AS phone_number,  
       ad.email AS email_address,  
       ad.household  
FROM Adopter ad  
WHERE ad.last_name LIKE '%@lastName%';
```

- **firstName** + **lastName** (clickable link)
- Address: **street**, **city**, **state**, **zipCode**
- **phoneNumber**

- **emailAddress**
- **householdSize**
- Upon selecting an applicant (via **firstName** + **lastName** link):
  - Display most recent application

```
SELECT a.email,
       a.first_name,
       a.last_name,
       a.phone,
       a.street,
       a.city,
       a.state,
       a.zip,
       a2.date,
       d.name,
       CASE WHEN d.name = 'Sideways'
            AND EXISTS (SELECT 1 FROM DogBreed db
                        JOIN Breed b ON db.name = b.name
                        WHERE db.dogID = d.dogID
                        AND b.name LIKE '%Terrier%')
            THEN 0
            WHEN d.by_animal_control = TRUE
            THEN ROUND(COALESCE(SUM(e.amount) * 0.10, 0), 2)
            ELSE ROUND(COALESCE(SUM(e.amount) * 1.25, 0), 2)
       END AS adoptionFee,
       @decDate
FROM Adopter a
LEFT JOIN Application a2 on a.email = a2.email
LEFT JOIN Dog d ON d.dogID = @dogID
LEFT JOIN Expense e ON d.dogID = e.dogID
LEFT JOIN DogBreed db ON d.dogID = db.dogID
WHERE a.first_name = @firstName
AND a.last_name = @lastName
GROUP BY a.email, a.first_name, a.last_name, a.phone,
         a.street, a.city, a.state, a.zip, a2.date, d.dogID,
         d.name
ORDER BY a2.date DESC
LIMIT 1;
```

- Store results in list of applicationDetail objects : applicationDetails
- Calculate and display **adoptionFee**:
  - \* If dog's **breeds** contains "Terrier" and dog's **name** is "Side-ways", set **adoptionFee** to 0

## 1.9 Set Adoption Date

- Upon entering **adoptionDate**:
  - Retrieve application from applicationDetails list
  - Display confirmation screen with:
    - \* Dog name
    - \* Adopter contact info
    - \* Adoption fee
    - \* Adoption date
  - Display **Submit** button:

```
SELECT d.dogID, d.name, d.altered, dm.microchipID
FROM Dog d
LEFT JOIN DogMicrochip dm ON d.dogID = dm.dogID
WHERE d.dogID = @dogID
AND d.altered = FALSE
AND dm.microchipID IS NULL;
```

- \* Disabled if **microchipId** is null and **altered** is false

## 1.10 View Dog Dashboard

- Calculable available space:
  - **availableSpace** = **maxCapacity** - **currentNumberOfDogs**

```
SELECT COUNT(d.dogID)
FROM Dog d
WHERE NOT EXISTS (
    SELECT 1
    FROM Adoption a
    WHERE a.dogID = d.dogID);
```

- Display the following:
  - **Add Adoption Application** button
  - If **maxCapacity** is greater than **currentNumberOfDogs**
    - \* Display **Add Dog** button
  - **Filter** dropdown
- If user is a director:
  - Display the following:
    - \* **Adoption Application Review** button

- \* **Animal Control Report** button
- \* **Monthly Adoption Report** button
- \* **Expense Analysis** button
- \* **Volunteer Lookup** button
- \* **Volunteer Birthdays** button

- Display Available Space: {**availableSpace**}
- Display all dogs currently in shelter in a table:

```
SELECT d.dogID, d.name, GROUP_CONCAT(db.name
ORDER BY db.name SEPARATOR '/')
as breed, d.sex, d.altered, d.age,
CASE
  WHEN EXISTS (SELECT 1 FROM Adoption a
WHERE a.dogID = d.dogID)
  THEN 'Adopted'
  WHEN EXISTS(SELECT 1 FROM DogMicrochip mc
WHERE mc.dogID = d.dogID AND d.altered = 1)
  THEN 'Adoptable'
  ELSE 'Not Adoptable'
END AS adoptability_status
FROM Dog d
JOIN DogBreed db ON db.dogID = d.dogID
GROUP BY d.dogID;
```

- Store dogID but do not display in table. Used for routing to dog details
  - Columns: *dogName* (clickable link), *breeds*, *sex*, *altered*, *age*, and *adoptability*
  - Order dogs oldest to newest by *surrenderDate*
- If user selects a value from the **Filter** dropdown:
  - If value is adoptable:
    - \* Display in table dogs that are only adoptable
  - Else if value is not adoptable:
    - \* Display in table dogs that are not adoptable
  - Else if value is All (default):
    - \* Display all dogs in table
- Upon:
  - Click **dogName** link - Jump to the **Dog Detail** task
  - Click **Adoption Application Review** button - Jump to the **Adoption Application Review** task

- Click **Animal Control Report** button - Jump to the **Animal Control Report** task
- Click **Monthly Adoption Report** button - Jump to the **Monthly Adoption Report** task
- Click **Expense Analysis** button - Jump to the **Expense Analysis** task
- Click **Volunteer Lookup** button - Jump to the **Volunteer Lookup** task
- Click **Volunteer Birthdays** button - Jump to the **Volunteer Birthdays** task
- Click **Add Adoption Application** button - Jump to the **Add Adoption Application** task
- Click **Add Dog** button - Jump to the **Add Dog** task

### 1.11 View Animal Control Report

- Display table with:

```
WITH RECURSIVE calendar(date) AS (
SELECT makedate(2023,1) UNION ALL
SELECT DATE_ADD(date,INTERVAL 1 Month) FROM calendar
WHERE date < CURRENT_DATE
)

SELECT DATE_FORMAT(calendar.date, '%m-%Y') AS Month,
COALESCE(SUM(combined.CountSurrendered),0) AS
'Dogs Surrendered by Animal Control', COALESCE(
SUM(combined.CountAdopted),0) AS 'Adopted Dogs
in Shelter >60 Days', COALESCE(SUM(
combined.TotalExpenses),0) AS 'Total Expenses
From All Adopted Dogs' FROM calendar LEFT JOIN (
-- surrendered by animal control
(SELECT DATE_FORMAT(d.surrender_date, '%m-%Y') AS
`Month`, count(d.dogID) AS CountSurrendered,
0 AS CountAdopted, 0 AS TotalExpenses
FROM Dog d
WHERE d.by_animal_control = 1
AND PERIOD_DIFF(EXTRACT(YEAR_Month FROM NOW()),
EXTRACT(YEAR_Month FROM d.surrender_date)) <= 6
GROUP BY d.surrender_date)
UNION ALL
-- Adoption after 60 days in shelter
(SELECT DATE_FORMAT(ad.decision_date, '%m-%Y') AS
`Month`, 0 AS CountSurrendered, count(ad.dogID)
AS CountAdopted, 0 AS TotalExpenses
```

```

FROM Adoption ad JOIN Dog d ON ad.dogID = d.dogID
WHERE DATEDIFF(ad.decision_date,
  d.surrender_date) >= 60
AND PERIOD_DIFF(EXTRACT(YEAR_Month FROM NOW()),
  EXTRACT(YEAR_Month FROM ad.decision_date)) <= 6
GROUP BY ad.decision_date)
UNION ALL
-- expenses for adopted dogs
(SELECT DATE_FORMAT(ad.decision_date, '%m-%Y') AS
  `Month`, 0 AS CountSurrendered, 0 AS CountAdopted,
  SUM(e.amount) AS TotalExpenses
FROM Adoption ad LEFT JOIN Expense e
  ON ad.dogID = e.dogID
WHERE PERIOD_DIFF(EXTRACT(YEAR_Month FROM NOW()),
  EXTRACT(YEAR_Month FROM ad.decision_date)) <= 6
GROUP BY ad.decision_date) AS combined ON DATE_FORMAT
(calendar.date, '%m-%Y') = combined.Month
WHERE PERIOD_DIFF(EXTRACT(YEAR_Month FROM NOW()),
  EXTRACT(YEAR_Month FROM calendar.date)) <= 6
AND PERIOD_DIFF(EXTRACT(YEAR_Month FROM NOW()),
  EXTRACT(YEAR_Month FROM calendar.date)) >= 0
GROUP BY DATE_FORMAT(calendar.date, '%m-%Y'),
  combined.Month, calendar.date
ORDER BY calendar.date;

```

5 6

- Rows: One for each month (current month plus previous 6 months):
- Columns:
  - \* Month (start/end dates)
  - \* Count of dogs surrendered by animal control (clickable)
  - \* Count of dogs adopted after 60+ days in rescue (clickable)
  - \* Total expenses for adopted dogs
  - \* Current month row shows data up to current date
  - \* Each count/total is clickable and displays corresponding drill-down report
- Upon clicking a cell, display corresponding drill-down report:

<sup>5</sup>DATE\_FORMAT() is a function in MySQL See [https://www.w3schools.com/sql/func\\_mysql\\_date\\_format.asp](https://www.w3schools.com/sql/func_mysql_date_format.asp)

<sup>6</sup>calendar is a recursive object to create the month year dates in scope. If a month year in scope does not exist in the database, the calendar object inserts it with 0s for all fields <https://dev.mysql.com/doc/refman/8.4/en/with.html#common-table-expressions-recursive-date-series>

### – Animal Control Surrenders Drill Down

```
SELECT d.dogID AS 'Dog ID', COALESCE(GROUP_CONCAT
  (DISTINCT db.name ORDER BY db.name SEPARATOR
  '/'), 'Unknown') AS Breed, d.sex AS Sex, d.altered
  AS Altered, dm.microchipID AS 'Microchip ID',
  d.surrender_date AS 'Surrender Date'
FROM Dog d LEFT JOIN DogBreed db ON db.dogID = d.dogID
  LEFT JOIN DogMicrochip dm ON dm.dogID = d.dogID
WHERE d.by_animal_control = 1
AND PERIOD_DIFF(EXTRACT(YEAR_Month FROM NOW()),
  EXTRACT(YEAR_Month FROM d.surrender_date)) <= 6
AND DATE_FORMAT(d.surrender_date, '%m-%Y') = @Month
GROUP BY d.dogID
ORDER BY d.dogID;
```

<sup>7</sup> <sup>8</sup>

\* Display table with columns:

- **dogID**
- **breeds** (forward slash-separated in alphabetical if multiple)
- **sex**
- **altered**
- **microChipID**
- **surrenderDate**

\* Sort by **dogID** ascending

### – Dogs Adopted (60+ days) Drill-down

```
SELECT d.dogID AS 'Dog ID', COALESCE(GROUP_CONCAT
  (DISTINCT db.name ORDER BY db.name SEPARATOR '/'),
  'Unknown') AS Breed, d.sex AS Sex, dm.microchipID
  AS 'Microchip ID', d.surrender_date AS
  'Surrender Date', DATEDIFF(ad.decision_date,
  d.surrender_date) AS 'Days in Rescue'
FROM Adoption ad JOIN Dog d ON ad.dogID = d.dogID
  LEFT JOIN DogBreed db ON db.dogID = d.dogID
  LEFT JOIN DogMicrochip dm ON dm.dogID = d.dogID
WHERE DATEDIFF(ad.decision_date, d.surrender_date) >= 60
AND PERIOD_DIFF(EXTRACT(YEAR_Month FROM NOW()),
  EXTRACT(YEAR_Month FROM ad.decision_date)) <= 6
AND DATE_FORMAT(ad.decision_date, '%m-%Y') = @Month
GROUP BY d.dogID, ad.decision_date
```

<sup>7</sup>PERIOD\_DIFF() is a function in MySQL See [https://www.w3schools.com/sql/func\\_mysql\\_period\\_diff.asp](https://www.w3schools.com/sql/func_mysql_period_diff.asp)

<sup>8</sup>EXTRACT() is a function in MySQL See [https://www.w3schools.com/sql/func\\_mysql\\_extract.asp](https://www.w3schools.com/sql/func_mysql_extract.asp)



```
ORDER BY DATEDIFF(ad.decision_date, d.surrender_date)
DESC, d.dogID DESC;
```

\* Display table with columns:

- **dogID**
- **breeds** (forward slash-separated in alphabetical if multiple)
- **sex**
- **microChipID**
- **surrenderDate**
- **daysInRescue** (count includes both surrender and adoption dates)

\* Sort **daysInRescue** descending then **dogID** descending

#### – Adopted Dogs Expenses Drill-down

```
SELECT combined.dogID AS 'Dog ID', combined.breeds
AS Breed, combined.sex AS Sex, combined.microchipID
AS 'Microchip ID', combined.surrender_date AS
'Surrender Date', combined.by_animal_control AS
'Surrendered by Animal Control',
SUM(combined.amount) AS 'Total Expenses' FROM (
SELECT d.dogID, COALESCE(GROUP_CONCAT(DISTINCT db.name
ORDER BY db.name SEPARATOR '/'), 'Unknown') AS
breeds, d.sex, dm.microchipID, d.surrender_date,
d.by_animal_control, e.amount
FROM Adoption ad JOIN Dog d ON ad.dogID = d.dogID
LEFT JOIN DogBreed db ON db.dogID = d.dogID
LEFT JOIN DogMicrochip dm ON dm.dogID = d.dogID
JOIN Expense e ON d.dogID = e.dogID
WHERE PERIOD_DIFF(EXTRACT(YEAR_Month FROM NOW()),
EXTRACT(YEAR_Month FROM ad.decision_date)) <= 6
AND DATE_FORMAT(ad.decision_date, '%m-%Y') = @Month
GROUP BY d.dogID, e.amount
) AS combined
GROUP BY combined.dogID, combined.breeds, combined.sex,
combined.microchipID, combined.surrender_date,
combined.by_animal_control
ORDER BY combined.dogID;
```

\* Display table with columns:

- **dogID**
- **breeds** (forward slash-separated in alphabetical if multiple)
- **sex**
- **microChipID**
- **surrenderDate**

- **byAnimalControl** indicator
- **totalExpenses** (exclude from expense if dog.byAnimalControl is true)
- \* Sort by **dogID** ascending

## 1.12 View Adoption Report

- Display table for previous 12 months (excluding current month):

```

SELECT
    DATE_FORMAT(date_column, '%m-%Y') AS month_year,
    combined.breeds,
    SUM(table_name = 'table1') AS Surrendered_dogs,
    SUM(table_name = 'table2') AS Adopted_dogs,
    SUM(expense_amount) AS total_expense,
    SUM(adoption_fee) as total_adoptionFee,
    SUM(adoption_fee) - SUM(expense_amount) AS profit
FROM (
    SELECT COALESCE(GROUP_CONCAT(DISTINCT db.name ORDER BY
        db.name SEPARATOR '/'), 'Unknown') AS breeds,
        d.surrender_date AS date_column, 'table1'
        AS table_name,
        0 AS expense_amount, 0 as adoption_fee
    FROM Dog d JOIN DogBreed db ON db.dogID = d.dogID
    WHERE d.surrender_date IS NOT NULL
    GROUP BY d.surrender_date, d.dogID
    UNION ALL
    SELECT COALESCE(GROUP_CONCAT(DISTINCT db.name
        ORDER BY db.name SEPARATOR '/'), 'Unknown')
        AS breeds, a.decision_date AS date_column,
        'table2' AS table_name, 0 AS expense_amount,
        0 as adoption_fee
    FROM Adoption a JOIN DogBreed db on db.dogID = a.dogID
    WHERE a.decision_date IS NOT NULL
    GROUP BY a.decision_date, a.dogID
    UNION ALL
    SELECT COALESCE(GROUP_CONCAT(DISTINCT db.name ORDER BY
        db.name SEPARATOR '/'), 'Unknown') AS breeds, e.`date`
        AS date_column, 'table3' AS table_name, e.amount
        as expense_amount,
        0 as adoption_fee FROM Expense e
    JOIN Dog d ON d.dogID = e.dogID JOIN DogBreed db
        ON db.dogID = d.dogID
    WHERE e.date IS NOT NULL AND d.by_animal_control = 0
    GROUP BY e.`date`, e.amount, d.by_animal_control, e.dogID
    UNION ALL

```

```

SELECT COALESCE(GROUP_CONCAT(DISTINCT db.name ORDER BY
    db.name SEPARATOR '/'), 'Unknown') AS breeds,
    a.decision_date AS date_column, 'table4' AS
    table_name, 0 AS expense_amount,
CASE WHEN d.by_animal_control = 1 THEN e.amount * 0.10
    ELSE e.amount * 1.25 END AS adoption_fee FROM
    Expense e
JOIN Adoption a ON a.dogID = e.dogID JOIN DogBreed db ON
    db.dogID = e.dogID JOIN Dog d ON d.dogID = e.dogID
WHERE e.date IS NOT NULL
GROUP BY e.amount, e.`date`, a.decision_date,
    d.by_animal_control, e.dogID
) AS combined
WHERE combined.date_column IS NOT NULL AND
    PERIOD_DIFF(EXTRACT(YEAR_MONTH FROM NOW()),
        EXTRACT(YEAR_MONTH FROM combined.date_column)) <= 13 AND
    PERIOD_DIFF(EXTRACT(YEAR_MONTH FROM NOW()),
        EXTRACT(YEAR_MONTH FROM combined.date_column)) >= 1
GROUP BY month_year, combined.breeds
ORDER BY STR_TO_DATE(CONCAT('01-', month_year), '%d-%m-%Y');

```

9 10

– Columns:

- \* Month/Year
- \* Number of dogs surrendered
- \* Number of dogs adopted
- \* Total expenses (exclude from expense if dog.byAnimalControl is true)
- \* Total adoption fees (Calculate Per Dog)
  - If **dog.byAnimalControl** is true: (**adoptionFee** = **totalExpenses** / 0.10)
  - Else: (**adoptionFee** = **totalExpenses** / 1.25)
- \* Net Profit:
  - **totalAdoptionFees** - **totalExpenses**
- \* Group Rows by Breed:
  - Show only breeds adopted/surrendered in 12-month period
  - For multiple breeds, combine names alphabetically with delimiter
- \* Sort by:

<sup>9</sup>STR\_TO\_DATE() is a function in MySQL See [https://www.w3schools.com/sql/func\\_mysql\\_str\\_to\\_date.asp](https://www.w3schools.com/sql/func_mysql_str_to_date.asp)

<sup>10</sup>PERIOD\_DIFF() is a function in MySQL See [https://www.w3schools.com/sql/func\\_mysql\\_period\\_diff.asp](https://www.w3schools.com/sql/func_mysql_period_diff.asp)

- Month ascending (earliest to latest)
- Breed name alphabetically

### 1.13 View Expense Analysis Report

- Display table with::

```
SELECT e.vendor AS vendorName, SUM(e.amount)
      AS totalExpenses
FROM Expense e
JOIN Dog d ON e.dogID = d.dogID
WHERE d.by_animal_control = FALSE
GROUP BY e.vendor
ORDER BY totalExpenses DESC;
```

– **vendorName:**

- \* If dog.**byAnimalControl** is true
  - Exclude from expenses
- \* Sort by **totalExpenses** descending:

### 1.14 View Volunteer Lookup Report

- Search by first or last name containing (case insensitive)

```
SELECT u.first_name, u.last_name, u.email, u.phone
FROM Volunteer v JOIN `User` u ON u.email = v.email
WHERE u.first_name LIKE
      CONCAT('%',@searchText, '%')
      OR u.last_name
      LIKE CONCAT('%',@searchText, '%')
ORDER BY u.last_name ASC, u.first_name ASC;
```

– Display:

- \* First name
- \* Last name
- \* Email address
- \* Phone number

– Sort by last name ascending, first name ascending

### 1.15 View Volunteer Birthdays Report

- Display month/year selection:

```
SELECT u.first_name, u.last_name, v.email,
CASE WHEN (@selectedYear - YEAR(v.birth_date)
```

```
- CASE WHEN MONTH(v.birth_date) > @selectedMonth
  OR (MONTH(v.birth_date) = @selectedMonth
    AND DAY(v.birth_date) > DAY(CURRENT_DATE))
  THEN 1 ELSE 0 END) % 10 = 0
THEN 'Yes' ELSE 'No' END AS milestoneBirthday
FROM Volunteer v
JOIN `User` u ON u.email = v.email
WHERE MONTH(v.birth_date) = @selectedMonth
ORDER BY u.last_name ASC, u.first_name ASC;
```

- Dropdown with all months
- Year options limited to current and previous year
- Default to current month/year
- For selected month:
  - If no birthdays: Display message "No volunteer birthdays this month!"
  - If birthdays exist, display table with:
    - \* **firstName**
    - \* **lastName**
    - \* **emailAddress**
    - \* **milestoneBirthday** (Yes/No for ages divisible by 10)
- Sort by **lastName** ascending, then **firstName** ascending