Welcome to Intermediate SQL!

DATA MANIPULATION IN SQL

SQL

Mona Khalil

Data Scientist, Greenhouse Software

Topics covered:

- CASE statements
- Simple subqueries
- Correlated subqueries
- Window functions

Prerequisites

- Selecting, filtering, and grouping data

```sql
SELECT user_id, SUM(sales)
FROM sales_data
WHERE user_id BETWEEN 300 AND 400
GROUP BY user_id;
```

- Using joins

```sql
SELECT c.country, c.team, SUM(m.goals)
FROM countries AS c
LEFT JOIN matches AS m
ON c.team_id = m.home_team_id
WHERE m.year > 1990
GROUP BY c.country, c.team;
```

Selecting from the European Soccer Database

```sql
SELECT
    l.name AS league,
    COUNT(m.country_id) as total_matches
FROM league AS l
LEFT JOIN match AS m
ON l.country_id = m.country_id
GROUP BY l.name;
```

```
| league                 | total_matches |
|------------------------|---------------|
| Belgium Jupiler League | 732           |
| England Premier League | 1520          |
| France Ligue 1         | 1520          |
| Germany 1. Bundesliga  | 1224          |
```

Selecting from the European Soccer Database

```sql
SELECT
    date,
    id,
    home_goal,
    away_goal
FROM match
WHERE season = '2013/2014';
```

```
| date                | id   | home_goal | away_goal |
|---------------------|------|-----------|-----------|
| 2014-03-29 00:00:00 | 1237 | 2         | 0         |
| 2014-03-29 00:00:00 | 1238 | 0         | 1         |
| 2014-04-05 00:00:00 | 1239 | 1         | 0         |
| 2014-04-05 00:00:00 | 1240 | 0         | 0         |
```

Selecting from the European Soccer Database

```
SELECT
    date,
    id,
    home_goal,
    away_goal
FROM match
WHERE season = '2013/2014'
      AND home_team_goal > away_team_goal;
```

```
| date                | id   | home_goal | away_goal |
|---------------------|------|-----------|-----------|
| 2014-03-29 00:00:00 | 1237 | 2         | 0         |
| 2014-04-05 00:00:00 | 1239 | 1         | 0         |
| 2014-04-12 00:00:00 | 1241 | 2         | 1         |
| 2014-04-12 00:00:00 | 1242 | 2         | 0         |
```

CASE statements

- Contains a `WHEN` , `THEN` , and `ELSE` statement, finished with `END`

```
CASE WHEN x = 1 THEN 'a'
     WHEN x = 2 THEN 'b'
     ELSE 'c' END AS new_column
```

CASE WHEN

```sql
SELECT
  id,
  home_goal,
  away_goal,
  CASE WHEN home_goal > away_goal THEN 'Home Team Win'
       WHEN home_goal < away_goal THEN 'Away Team Win'
       ELSE 'Tie' END AS outcome
FROM match
WHERE season = '2013/2014';
```

```
| id   | home_goal | away_goal | outcome       |
|------|-----------|-----------|---------------|
| 1237 | 2         | 0         | Home Team Win |
| 1238 | 0         | 1         | Away Team Win |
| 1239 | 1         | 0         | Home Team Win |
| 1240 | 0         | 0         | Tie           |
```

Let's practice!

In CASE things get more complex

DATA MANIPULATION IN SQL

SQL

Mona Khalil
Data Scientist, Greenhouse Software

Reviewing CASE WHEN

```
SELECT
    date,
    season,
    CASE WHEN home_goal > away_goal THEN 'Home team win!'
         WHEN home_goal < away_goal THEN 'Away team win!'
         ELSE 'Tie' END AS outcome
FROM match;
```

```
| date       | season    | outcome            |
|------------|-----------|--------------------|
| 2011-08-09 | 2011/2012 | Home team win!     |
| 2011-09-01 | 2011/2012 | Away team win!     |
| 2011-09-14 | 2011/2012 | Tie                |
| 2011-10-04 | 2011/2012 | Home team win!     |
```

CASE WHEN ... AND then some

- Add multiple logical conditions to your `WHEN` clause!

```
SELECT date, hometeam_id, awayteam_id,
  CASE WHEN hometeam_id = 8455 AND home_goal > away_goal
            THEN 'Chelsea home win!'
       WHEN awayteam_id = 8455 AND home_goal < away_goal
            THEN 'Chelsea away win!'
       ELSE 'Loss or tie :(' END AS outcome
FROM match
WHERE hometeam_id = 8455 OR awayteam_id = 8455;
```

```
| date       | hometeam_id | awayteam_id | outcome           |
|------------|-------------|-------------|-------------------|
| 2011-08-14 | 10194       | 8455        | Loss or tie :(    |
| 2011-08-20 | 8455        | 8659        | Chelsea home win! |
| 2011-08-27 | 8455        | 9850        | Chelsea home win! |
| 2011-09-10 | 8472        | 8455        | Chelsea away win! |
```

What ELSE is being excluded?

- What's in your `ELSE` clause?

```sql
SELECT date, hometeam_id, awayteam_id,
  CASE WHEN hometeam_id = 8455 AND home_goal > away_goal
            THEN 'Chelsea home win!'
       WHEN awayteam_id = 8455 AND home_goal < away_goal
            THEN 'Chelsea away win!'
       ELSE 'Loss or tie :(' END AS outcome
FROM match;
```

```
| date       | hometeam_id | awayteam_id | outcome        |
|------------|-------------|-------------|----------------|
| 2011-07-29 | 1773        | 8635        | Loss or tie :( |
| 2011-07-30 | 9998        | 9985        | Loss or tie :( |
| 2011-07-30 | 9987        | 9993        | Loss or tie :( |
| 2011-07-30 | 9991        | 9984        | Loss or tie :( |
```

Correctly categorize your data with CASE

```sql
SELECT date, hometeam_id, awayteam_id,
  CASE WHEN hometeam_id = 8455 AND home_goal > away_goal
            THEN 'Chelsea home win!'
       WHEN awayteam_id = 8455 AND home_goal < away_goal
            THEN 'Chelsea away win!'
       ELSE 'Loss or tie :(' END AS outcome
FROM match
WHERE hometeam_id = 8455 OR awayteam_id = 8455;
```

```
| date       | hometeam_id | awayteam_id | outcome           |
|------------|-------------|-------------|-------------------|
| 2011-08-14 | 10194       | **8455**    | Loss or tie :(    |
| 2011-08-20 | **8455**    | 8659        | Chelsea home win! |
| 2011-08-27 | **8455**    | 9850        | Chelsea home win! |
| 2011-09-10 | 8472        | **8455**    | Chelsea away win! |
```

What's NULL?

```sql
SELECT date,
CASE WHEN date > '2015-01-01' THEN 'More Recently'
     WHEN date < '2012-01-01' THEN 'Older'
     END AS date_category
FROM match;
SELECT date,
CASE WHEN date > '2015-01-01' THEN 'More Recently'
     WHEN date < '2012-01-01' THEN 'Older'
     ELSE NULL END AS date_category
FROM match;
```

```
| date       | date_category |
|------------|---------------|
| 2011-11-18 | Older         |
| 2012-02-11 | NULL          |
| 2014-11-07 | NULL          |
| 2015-02-14 | More Recently |
```

What are your NULL values doing?

```
SELECT date, season,
  CASE WHEN hometeam_id = 8455 AND home_goal > away_goal
            THEN 'Chelsea home win!'
       WHEN awayteam_id = 8455 AND home_goal < away_goal
            THEN 'Chelsea away win!'
       END AS outcome
FROM match
WHERE hometeam_id = 8455 OR awayteam_id = 8455;
```

```
| date       | season    | outcome           |
|------------|-----------|-------------------|
| 2011-08-14 | 2011/2012 | NULL              |
| 2011-12-22 | 2011/2012 | NULL              |
| 2012-12-08 | 2012/2013 | Chelsea away win! |
| 2013-03-02 | 2012/2013 | Chelsea home win! |
```

Where to place your CASE?

```sql
SELECT date, season,
  CASE WHEN hometeam_id = 8455 AND home_goal > away_goal
            THEN 'Chelsea home win!'
       WHEN awayteam_id = 8455 AND home_goal < away_goal
            THEN 'Chelsea away win!' END AS outcome
FROM match;
```

Where to place your CASE?

```
SELECT date, season,
  CASE WHEN hometeam_id = 8455 AND home_goal > away_goal
            THEN 'Chelsea home win!'
       WHEN awayteam_id = 8455 AND home_goal < away_goal
            THEN 'Chelsea away win!' END AS outcome
FROM match
WHERE CASE WHEN hometeam_id = 8455 AND home_goal > away_goal
                THEN 'Chelsea home win!'
           WHEN awayteam_id = 8455 AND home_goal < away_goal
                THEN 'Chelsea away win!' END IS NOT NULL;
```

```
| date       | season    | outcome           |
|------------|-----------|-------------------|
| 2011-11-05 | 2011/2012 | Chelsea away win! |
| 2011-11-26 | 2011/2012 | Chelsea home win! |
| 2011-12-03 | 2011/2012 | Chelsea away win! |
```

Let's practice!

DATA MANIPULATION IN SQL

CASE WHEN with aggregate functions

DATA MANIPULATION IN SQL

SQL

Mona Khalil
Data Scientist, Greenhouse Software

In CASE you need to aggregate

- `CASE` statements are great for
  - Categorizing data
  - Filtering data
  - Aggregating data

COUNTing CASES

- How many home and away goals did Liverpool score in each season?

```
| season    | home_wins | away_wins |
|-----------|-----------|-----------|
| 2011/2012 |           |           |
| 2012/2013 |           |           |
| 2013/2014 |           |           |
| 2014/2015 |           |           |
```

CASE WHEN with COUNT

```sql
SELECT
    season,
    COUNT(CASE WHEN hometeam_id = 8650
               AND home_goal > away_goal
               THEN id END) AS home_wins
FROM match
GROUP BY season;
```

CASE WHEN with COUNT

```
SELECT
    season,
    COUNT(CASE WHEN hometeam_id = 8650 AND home_goal > away_goal
          THEN id END) AS home_wins,
    COUNT(CASE WHEN awayteam_id = 8650 AND away_goal > home_goal
          THEN id END) AS away_wins
FROM match
GROUP BY season;
```

```
| season    | home_wins | away_wins |
|-----------|-----------|-----------|
| 2011/2012 | 6         | 8         |
| 2012/2013 | 9         | 7         |
| 2013/2014 | 16        | 10        |
| 2014/2015 | 10        | 8         |
```

CASE WHEN with COUNT

```
SELECT
    season,
    COUNT(CASE WHEN hometeam_id = 8650 AND home_goal > away_goal
          THEN 54321 END) AS home_wins,
    COUNT(CASE WHEN awayteam_id = 8650 AND away_goal > home_goal
          THEN 'Some random text' END) AS away_wins
FROM match
GROUP BY season;
```

```
| season    | home_wins | away_wins |
|-----------|-----------|-----------|
| 2011/2012 | 6         | 8         |
| 2012/2013 | 9         | 7         |
| 2013/2014 | 16        | 10        |
| 2014/2015 | 10        | 8         |
```

CASE WHEN with SUM

```sql
SELECT
    season,
    SUM(CASE WHEN hometeam_id = 8650
          THEN home_goal END) AS home_goals,
    SUM(CASE WHEN awayteam_id = 8650
          THEN away_goal END) AS away_goals
FROM match
GROUP BY season;
```

```
| season    | home_goals | away_goals |
|-----------|------------|------------|
| 2011/2012 | 24         | 23         |
| 2012/2013 | 33         | 38         |
| 2013/2014 | 53         | 48         |
| 2014/2015 | 30         | 22         |
```

The CASE is fairly AVG...

```
SELECT
    season,
    AVG(CASE WHEN hometeam_id = 8650
          THEN home_goal END) AS avg_homegoals,
    AVG(CASE WHEN awayteam_id = 8650
          THEN away_goal END) AS avg_awaygoals
FROM match
GROUP BY season;
```

```
| season    | avg_homegoals    | avg_awaygoals    |
|-----------|------------------|------------------|
| 2011/2012 | 1.26315789473684 | 1.21052631578947 |
| 2012/2013 | 1.73684210526316 | 2                |
| 2013/2014 | 2.78947368421053 | 2.52631578947368 |
| 2014/2015 | 1.57894736842105 | 1.15789473684211 |
```

A ROUNDed AVG

```
ROUND(3.141592653589,2)
```

```
3.14
```

A ROUNDed AVG

```
SELECT
  season,
  ROUND(AVG(CASE WHEN hometeam_id = 8650
        THEN home_goal END),2) AS avg_homegoals,
  ROUND(AVG(CASE WHEN awayteam_id = 8650
        THEN away_goal END),2) AS avg_homegoals
FROM match
GROUP BY season;
```

```
| season    | avg_homegoals     | avg_awaygoals     |
|-----------|-------------------|-------------------|
| 2011/2012 | 1.26              | 1.21              |
| 2012/2013 | 1.73              | 2                 |
| 2013/2014 | 2.78              | 2.52              |
| 2014/2015 | 1.57              | 1.15              |
```

Percentages with CASE and AVG

```
SELECT
  season,
  AVG(CASE WHEN hometeam_id = 8455 AND home_goal > away_goal THEN 1
           WHEN hometeam_id = 8455 AND home_goal < away_goal THEN 0
           END) AS pct_homewins,
  AVG(CASE WHEN awayteam_id = 8455 AND away_goal > home_goal THEN 1
           WHEN awayteam_id = 8455 AND away_goal < home_goal THEN 0
           END) AS pct_awaywins
FROM match
GROUP BY season;
```

```
| season    | pct_homewins     | pct_awaywins     |
|-----------|------------------|------------------|
| 2011/2012 | 0.75             | 0.5              |
| 2012/2013 | 0.85714285714286 | 0.66666666666667 |
| 2013/2014 | 0.9375           | 0.66666666666667 |
| 2014/2015 | 1                | 0.78571428571429 |
```

Percentages with CASE and AVG

```sql
SELECT
  season,
  ROUND(AVG(CASE WHEN hometeam_id = 8455 AND home_goal > away_goal THEN 1
           WHEN hometeam_id = 8455 AND home_goal < away_goal THEN 0
           END),2) AS pct_homewins,
  ROUND(AVG(CASE WHEN awayteam_id = 8455 AND away_goal > home_goal THEN 1
           WHEN awayteam_id = 8455 AND away_goal < home_goal THEN 0
           END),2) AS pct_awaywins
FROM match
GROUP BY season;
```

```
| season    | pct_homewins      | pct_awaywins      |
|-----------|-------------------|-------------------|
| 2011/2012 | 0.75              | 0.5               |
| 2012/2013 | 0.86              | 0.67              |
| 2013/2014 | 0.94              | 0.67              |
| 2014/2015 | 1                 | 0.79              |
```

Let's practice!

DATA MANIPULATION IN SQL