



INTERMEDIATE SQL

Window Functions

Mona Khalil

Curriculum Lead, DataCamp

Window Functions

```
SELECT season,  
       (SELECT AVG(home_goal + away_goal) AS overall_avg  
FROM match  
GROUP BY season;
```

```
SELECT id,  
       season,  
       (SELECT AVG(home_goal + away_goal) FROM match) AS overall_avg  
FROM match;
```

Window Functions!

- Calculates values based on data in the *window* (table)
- Returns one value per row
- Utilizes the `OVER()` clause with an aggregate function

```
SELECT id,  
       season,  
       AVG(home_goal + away_goal) OVER() AS overall_avg  
FROM match;
```

id	season	overall_avg
757	2011/2012	2.73
758	2011/2012	2.73
759	2011/2012	2.73
760	2011/2012	2.73

Compute with Window Functions

- Compute values with window functions!

```
SELECT id,  
       home_goal,  
       home_goal - AVG(home_goal) OVER() AS diff  
FROM match;
```

id	home_goal	diff
757	2	0.45
758	1	-0.55
759	3	1.45
760	0	-1.55



INTERMEDIATE SQL

Let's Practice!



INTERMEDIATE SQL

Window Partitions

Mona Khalil

Curriculum Lead, DataCamp

OVER and PARTITION BY

- Calculate separate values for different categories
- Contains different calculations in the same column

```
SELECT m.id,  
       c.name,  
       m.season,  
       m.home_goal,  
       m.away_goal,  
       AVG(home_goal + away_goal) OVER(PARTITION BY m.season) AS season_avg  
FROM country AS c  
LEFT JOIN match AS m ON c.id = m.country_id;
```

id	name	season	home_goal	away_goal	season_avg
757	Belgium	2011/2012	2	1	2.72
1479	Belgium	2014/2015	6	1	2.68
760	Belgium	2011/2012	0	1	2.72



PARTITION by Multiple Columns

```
SELECT m.id,  
       c.name,  
       m.season,  
       m.home_goal,  
       m.away_goal,  
       ROUND(AVG(home_goal + away_goal)  
              OVER(PARTITION BY m.season, c.name), 2) AS season_ctry_avg  
FROM country AS c  
LEFT JOIN match AS m ON c.id = m.country_id
```

id	name	season	home_goal	away_goal	season_ctry_avg
760	Belgium	2011/2012	0	1	2.88
15140	Netherlands	2014/2015	0	1	3.08
884	Belgium	2011/2012	0	1	2.88
24063	Spain	2014/2015	0	2	2.66



INTERMEDIATE SQL

Let's Practice!



INTERMEDIATE SQL

Sliding Windows

Mona Khalil

Curriculum Lead, DataCamp



Sliding Windows

- Perform calculations relative to the current row
- Can be used to calculate running totals, sums, averages, etc
- Can be used to calculate partial windows
- Can be partitioned by one or more columns



Sliding Window Keywords

```
ROWS BETWEEN <start> AND <finish>
```

```
PRECEDING  
FOLLOWING  
UNBOUNDED PRECEDING  
UNBOUNDED FOLLOWING  
CURRENT ROW
```



Sliding Window Example

```
/* Manchester City Home Games */  
SELECT date,  
       home_goal,  
       away_goal,  
       SUM(home_goal)  
         OVER(ORDER BY date  
              ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS running_total  
FROM match  
WHERE hometeam_id = 8456 AND season = '2011/2012';
```

date	home_goal	away_goal	running_total
2011-08-15	4	0	4
2011-09-10	3	0	7
2011-09-24	2	0	9
2011-10-15	4	1	13

Sliding Window Frame

```
/* Manchester City Home Games */  
SELECT date,  
       home_goal,  
       away_goal,  
       SUM(home_goal)  
         OVER (ORDER BY date  
              ROWS BETWEEN 1 PRECEDING  
              AND CURRENT ROW) AS last2  
FROM match  
WHERE hometeam_id = 8456  
      AND season = '2011/2012';
```

date	home_goal	away_goal	last2
2011-08-15	4	0	4
2011-09-10	3	0	7
2011-09-24	2	0	5
2011-10-15	4	1	6



INTERMEDIATE SQL

Let's Practice!



INTERMEDIATE SQL

Bringing it all Together

Mona Khalil

Curriculum Lead, DataCamp



Practice Exercises

- Constructing CASE statements
- Creating common table expressions
- Add a window function

Who defeated Manchester United in the 2013/2014 season?



INTERMEDIATE SQL

Let's Practice!