## **EURO 2020 SQL**

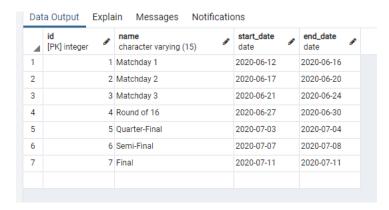
## SELECT \* FROM teams

ı	id [PK] integer	team_short_name character varying (3)	name character varying (100)	match_group character varying (1)
	1	BEL	Belgium	В
	2	ITA	Italy	A
	3	ENG	England	D
	4	GER	Germany	F
	5	ESP	Spain	E
	6	UKR	Ukraine	С
	7	FRA	France	F
	8	POL	Poland	E
	9	SUI	Switzerland	A
	10	CRO	Croatia	D
	11	NED	Netherlands	С
	12	RUS	Russia	В
	13	POR	Portugal	F
	14	TUR	Turkey	A
	15	DFN	Denmark	R

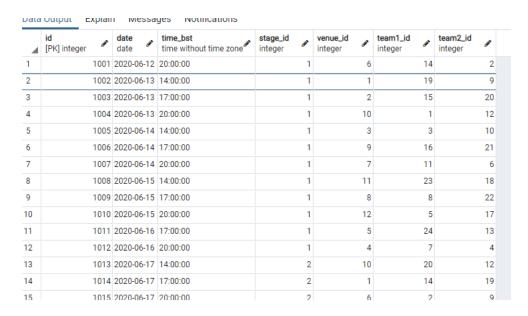
## SELECT \* FROM venues

4	id [PK] integer	name character varying (50)	city character varying (50)	country character varying (50)	capacity integer
1	1	Olympic Stadium	Baku	Azerbaijan	68700
2	2	Parken Stadium	Copenhagen	Denmark	3806
3	3	Wembley Stadium	London	United Kingdom	90000
4	4	Allianz Arena	Munich	Germany	7500
5	5	Puskas Arena	Budapest	Hungary	5600
6	6	Stadio Olimpico	Rome	Italy	7269
7	7	Johan Cruyff Arena	Amsterdam	Netherlands	5499
8	8	Aviva Stadium	Dublin	Republic of Ireland	5170
9	9	Arena Nationala	Bucharest	Romania	5560
10	10	Krestovsky Stadium	Saint Petersburg	Russia	6813
11	11	Hampden Park	Glasgow	United Kingdom	5206
12	12	San Mames	Bilbao	Spain	5333

## SELECT \* FROM stages



## SELECT \* FROM match



#### SELECT \* FROM scores

id [PK] integer	stage_id integer	venue_id integer	team1_id integer	team2_id integer	team1_goal integer	team2_goal integer	notes character varying (100)
1001	1	6	14	2	2	3	[null]
1002	1	1	19	9	0	1	[null]
1003	1	2	15	20	0	0	[null]
1004	1	10	1	12	2	2	[null]
1005	1	3	3	10	4	1	[null]
1006	1	9	16	21	2	2	[null]
1007	1	7	11	6	2	1	[null]
1008	1	11	23	18	0	1	[null]
1009	1	8	8	22	1	0	[null]
1010	1	12	5	17	1	3	[null]
1011	1	5	24	13	0	2	[null]
1012	1	4	7	4	2	2	[null]
1013	2	10	20	12	0	2	[null]
1014	2	1	14	19	1	0	[null]
1015	2	6	2	9	3	1	fnull1

## **Matchlist**

SELECT m.id, date, time\_bst, s.name AS stage, v.name AS venue, t1.name AS Team1, t2.name AS Team2
FROM match m
JOIN stages s
ON m.stage\_id = s.id
JOIN venues v
ON m.venue\_id = v.id
JOIN teams t1
ON m.team1\_id = t1.id
JOIN teams t2
ON m.team2\_id = t2.id
ORDER BY m.id

id in	teger 🖺	date a	time_bst time without time zone	gameweek character varying (15)	venue character varying (50) □	team1 character varying (100)	team2 character varying (100)
	1001	2020-06	20:00:00	Matchday 1	Stadio Olimpico	Turkey	Italy
	1002	2020-06	14:00:00	Matchday 1	Olympic Stadium	Wales	Switzerland
	1003	2020-06	17:00:00	Matchday 1	Parken Stadium	Denmark	Finland
	1004	2020-06	20:00:00	Matchday 1	Krestovsky Stadium	Belgium	Russia
	1005	2020-06	14:00:00	Matchday 1	Wembley Stadium	England	Croatia
	1006	2020-06	17:00:00	Matchday 1	Arena Nationala	Austria	Iceland
	1007	2020-06	20:00:00	Matchday 1	Johan Cruyff Arena	Netherlands	Ukraine
	1008	2020-06	14:00:00	Matchday 1	Hampden Park	Norway	Czech Republic
	1009	2020-06	17:00:00	Matchday 1	Aviva Stadium	Poland	Republic of Ireland
1	1010	2020-06	20:00:00	Matchday 1	San Mames	Spain	Sweden
	1011	2020-06	17:00:00	Matchday 1	Puskas Arena	North Macedonia	Portugal
!	1012	2020-06	20:00:00	Matchday 1	Allianz Arena	France	Germany
	1013	2020-06	14:00:00	Matchday 2	Krestovsky Stadium	Finland	Russia
	1014	2020-06	17:00:00	Matchday 2	Olympic Stadium	Turkey	Wales
	1015	2020-06-	20:00:00	Matchday 2	Stadio Olimpico	Italy	Switzerland

#### Euro 2020 scores

SELECT s.id, date, time\_bst, s.name AS stage, v.name AS venue, t1.name AS Team1, t2.name AS Team2, team1\_goal, team2\_goal FROM scores sc JOIN match m ON sc.id = m.id JOIN stages s ON sc.stage\_id = s.id JOIN venues v ON sc.venue\_id = v.id

JOIN teams t1

 $ON sc.team1_id = t1.id$ 

JOIN teams t2

ON sc.team2\_id = t2.id

ORDER BY sc.id

id integer	date a	time_bst time without time zone	stage character varying (15)	venue character varying (50)	team1 character varying (100)	team2 character varying (100)	team1_goal a	team2_goal a
1	2020-06-12		Matchday 1	Stadio Olimpico	Turkey	Italy	2	3
1	2020-06-13	14:00:00	Matchday 1	Olympic Stadium	Wales	Switzerland	0	1
1	2020-06-13	17:00:00	Matchday 1	Parken Stadium	Denmark	Finland	0	0
1	2020-06-13	20:00:00	Matchday 1	Krestovsky Stadium	Belgium	Russia	2	2
1	2020-06-14	14:00:00	Matchday 1	Wembley Stadium	England	Croatia	4	1
1	2020-06-14	17:00:00	Matchday 1	Arena Nationala	Austria	Iceland	2	2
1	2020-06-14	20:00:00	Matchday 1	Johan Cruyff Arena	Netherlands	Ukraine	2	1
1	2020-06-15	14:00:00	Matchday 1	Hampden Park	Norway	Czech Republic	0	1
1	2020-06-15	17:00:00	Matchday 1	Aviva Stadium	Poland	Republic of Ireland	1	0
1	2020-06-15	20:00:00	Matchday 1	San Mames	Spain	Sweden	1	3
1	2020-06-16	17:00:00	Matchday 1	Puskas Arena	North Macedonia	Portugal	0	2
1	2020-06-16	20:00:00	Matchday 1	Allianz Arena	France	Germany	2	2
2	2020-06-17	14:00:00	Matchday 2	Krestovsky Stadium	Finland	Russia	0	2
2	2020-06-17	17:00:00	Matchday 2	Olympic Stadium	Turkey	Wales	1	0
2	2020-06-17	20:00:00	Matchday 2	Stadio Olimpico	Italy	Switzerland	3	1

#### WHERE FUNCTIONS

# -- Scores from the first matchday SELECT \* FROM scores

WHERE stage\_id = 1

4	[PK] integer	character varying (100)						
	1001	1	6	14	2	2	3	[null]
!	1002	1	1	19	9	0	1	[null]
	1003	1	2	15	20	0	0	[null]
Ļ	1004	1	10	1	12	2	2	[null]
i	1005	1	3	3	10	4	1	[null]
i	1006	1	9	16	21	2	2	[null]
•	1007	1	7	11	6	2	1	[null]
1	1008	1	11	23	18	0	1	[null]
1	1009	1	8	8	22	1	0	[null]
)	1010	1	12	5	17	1	3	[null]
1	1011	1	5	24	13	0	2	[null]
2	1012	1	4	7	4	2	2	[null]

## -- Scores from the group stages

SELECT \* FROM scores
WHERE stage\_id BETWEEN 1 AND 3

4	id [PK] integer	stage_id integer	venue_id integer	team1_id integer	team2_id integer	team1_goal integer	team2_goal integer	notes character varying (100)
1	1001	1	6	14	2	2	3	[null]
2	1002	1	1	19	9	0	1	[null]
3	1003	1	2	15	20	0	0	[null]
4	1004	1	10	1	12	2	2	[null]
5	1005	1	3	3	10	4	1	[null]
6	1006	1	9	16	21	2	2	[null]
7	1007	1	7	11	6	2	1	[null]
8	1008	1	11	23	18	0	1	[null]
9	1009	1	8	8	22	1	0	[null]
10	1010	1	12	5	17	1	3	[null]
11	1011	1	5	24	13	0	2	[null]
12	1012	1	4	7	4	2	2	! [null]
13	1013	2	10	20	12	0	2	[null]
14	1014	2	1	14	19	1	0	[null]
15	1015	2	6	2	0	3	1	Inutil

## -- Scores that took place at Wembley Stadium

SELECT \* FROM scores WHERE venue\_id= 3

_	id [PK] integer	stage_id integer	venue_id integer	team1_id integer	team2_id integer	team1_goal integer	team2_goal integer	notes character varying (100)
1	1005	1	3	3	10	4	1	[null]
2	1021	2	3	3	23	5	0	[null]
3	1031	3	3	18	3	0	2	Czech Republic and England advance to the knockout stages
4	1038	4	3	2	6	2	1	[null]
5	1050	6	3	3	11	0	1	[null]
6	1051	7	3	13	11	0	1	[null]
7	1049	6	3	2	13	2	2	Portugal win 5-3 on penalties

## --Matches that took place within matchday 3

SELECT \* FROM match WHERE stage\_id = 3

4	id [PK] integer	date date	time_bst time without time zone	stage_id integer	venue_id integer	team1_id integer	team2_id integer
1	1025	2020-06-21	17:00:00	3	6	2	19
2	1026	2020-06-21	17:00:00	3	1	9	14
3	1027	2020-06-22	17:00:00	3	9	6	16
4	1028	2020-06-22	17:00:00	3	7	21	11
5	1029	2020-06-22	20:00:00	3	10	20	1
6	1030	2020-06-22	20:00:00	3	2	12	15
7	1031	2020-06-23	20:00:00	3	3	18	3
8	1032	2020-06-23	20:00:00	3	11	10	23
9	1033	2020-06-24	17:00:00	3	8	17	8
10	1034	2020-06-24	17:00:00	3	12	22	5
11	1035	2020-06-24	20:00:00	3	4	4	24
12	1036	2020-06-24	20:00:00	3	5	13	7

## AGGREGATE FUNCTIONS COUNT

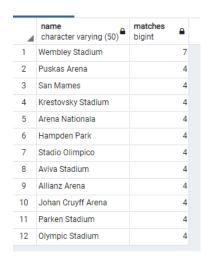
#### -- Number of matches in Euro 2020

SELECT COUNT(id) AS total\_matches FROM scores



#### -- Number of matches taken place each stadium

SELECT v.name, COUNT(\*) AS matches FROM scores JOIN venues AS v ON scores.venue\_id = v.id GROUP BY v.name ORDER BY matches DESC



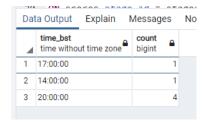
#### -- Number of matches per stage

SELECT stages.name, COUNT(\*) AS matches FROM scores
JOIN stages
ON scores.stage\_id = stages.id
GROUP BY stages.name
ORDER BY matches DESC, stages.name

υai	a Output Explain	messages	N
4	name character varying (15)	matches bigint	•
1	Matchday 1		12
2	Matchday 2		12
3	Matchday 3		12
4	Round of 16		8
5	Quarter-Final		4
6	Semi-Final		2
7	Final		1

#### --What times do England play and how many times?

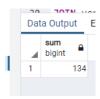
SELECT time\_bst, COUNT(\*)
FROM match
WHERE team1\_id = 3
OR team2\_id = 3
GROUP BY time\_bst
ORDER BY COUNT(\*)



#### SUM

#### -- Number of goals scored in Euro 2020

SELECT SUM(team1\_goal + team2\_goal) FROM scores



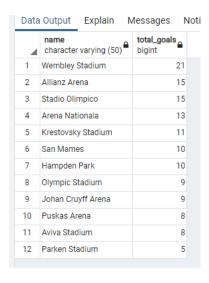
#### -- Number of goals scored in each round

SELECT stages.name, SUM(team1\_goal + team2\_goal) AS total\_goals FROM scores
JOIN stages
ON scores.stage\_id = stages.id
GROUP BY stages.name
ORDER BY total\_goals DESC, stages.name



## --Number of goals scored in each stadium

SELECT venues.name, SUM(team1\_goal + team2\_goal) AS total\_goals FROM scores
JOIN venues
ON scores.venue\_id = venues.id
GROUP BY venues.name
ORDER BY total\_goals DESC



## --Number of goals scored in each stadium capacity of < 55000

SELECT venues.name, SUM(team1\_goal + team2\_goal) AS total\_goals FROM scores

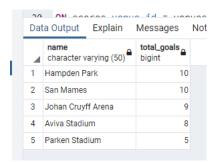
JOIN venues

ON scores.venue id = venues.id

WHERE capacity BETWEEN 30000 AND 55000

GROUP BY venues.name

ORDER BY total\_goals DESC

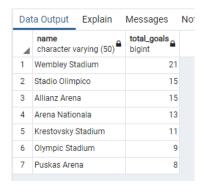


#### --Number of goals scored in each stadium capacity of > 55000

SELECT venues.name, SUM(team1\_goal + team2\_goal) AS total\_goals FROM scores JOIN venues

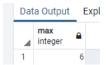
ON scores.venue\_id = venues.id WHERE capacity BETWEEN 55001 AND 90000 GROUP BY venues.name

ORDER BY total\_goals DESC



#### MAX

## -- Maximum number of goals scored in a single match in Euro 2020 SELECT MAX(team1\_goal + team2\_goal) FROM scores



#### -- Maximum number of goals scored in a match per round

SELECT stages.name, MAX(team1\_goal + team2\_goal) AS max\_goals FROM scores
JOIN stages
ON scores.venue\_id = stages.id
GROUP BY stages.name
ORDER BY max\_goals DESC, stages.name



## -- Maximum number of goals scored in a match per stadium

SELECT venues.name, MAX(team1\_goal + team2\_goal) AS max\_goals FROM scores
JOIN venues
ON scores.venue\_id = venues.id
GROUP BY venues.name
ORDER BY max\_goals DESC



ORDER BY m.id

#### **UNION**

## --Displaying all matches and venues

SELECT stage\_id, venue\_id, team1\_id, team2\_id FROM match

**UNION** 

SELECT stage\_id, venue\_id, team1\_id, team2\_id FROM scores

ORDER BY stage\_id, venue\_id

4	stage_id integer	venue_id integer	team1_id integer	team2_id integer
1	1	1	19	9
2	1	2	15	20
3	1	3	3	10
4	1	4	7	4
5	1	5	24	13
6	1	6	14	2
7	1	7	11	6
8	1	8	8	22
9	1	9	16	21
10	1	10	1	12
11	1	11	23	18
12	1	12	5	17
13	2	1	14	19
14	2	2	15	1
15	2	3	3	23

SELECT stage\_id, venue\_id, team1\_id, team2\_id FROM match UNION ALL SELECT stage\_id, venue\_id, team1\_id, team2\_id FROM scores ORDER BY stage\_id, venue\_id

Data (	Julpul Exp	itput Explain		sage	S NOUNCE	auc	10115		
4	stage_id integer		nue_id eger	•	team1_id integer	<u></u>	team2_id integer		
1		1		1		19	g		
2		1		- 1		19	g		
3		1		2		15	20		
4		1		2		15	20		
5		1		3		3	10		
6		1		3		3	10		
7		1		4		7	4		
8		1		4		7	4		
9		1		5		24	13		
10		1		5		24	13		
11		1		6		14	2		
12		1		6		14	2		
13		1		7		11	6		
14		1		7		11	6		
15		1		8		8	22		

#### **CASE STATEMENTS**

## --Identify home wins, losses or draw

SELECT id,

CASE WHEN team1\_goal > team2\_goal THEN 'Home win!' WHEN team1\_goal < team2\_goal THEN 'Home loss' ELSE 'Draw' END AS outcome

FROM scores



#### --Select matches where Netherlands was 'Team 2'

SELECT s.id, t.name,

CASE WHEN team1\_goal > team2\_goal THEN 'Holland lose!'

WHEN team1\_goal < team2\_goal THEN 'Holland win!'

ELSE 'Draw' END AS Outcome

FROM scores s

JOIN teams t

ON s.team1\_id = t.id

WHERE team2\_id = 11

4	id integer	name character varying (100)	outcome text
1	1050	England	Holland win!
2	1051	Portugal	Holland win!
3	1028	Iceland	Holland win!

## --Identify when Belgium won a match in Euro 2020

SELECT m.date, s.name, v.name,

CASE WHEN scores.team1\_id = 1 AND team1\_goal > team2\_goal THEN 'Belgium win' WHEN scores.team2\_id = 1 AND team2\_goal > team1\_goal THEN 'Belgium win'

END AS outcome

FROM scores

JOIN match m

ON scores.id = m.id

JOIN stages s

ON scores.stage\_id = s.id

JOIN venues v

ON scores.venue\_id = v.id

4	date a date	name character varying (15) <sup>△</sup>	name character varying (50)	outcome text
1	2020-06-12	Matchday 1	Stadio Olimpico	[null]
2	2020-06-13	Matchday 1	Olympic Stadium	[null]
3	2020-06-13	Matchday 1	Parken Stadium	[null]
4	2020-06-13	Matchday 1	Krestovsky Stadium	[null]
5	2020-06-14	Matchday 1	Wembley Stadium	[null]
6	2020-06-14	Matchday 1	Arena Nationala	[null]
7	2020-06-14	Matchday 1	Johan Cruyff Arena	[null]
8	2020-06-15	Matchday 1	Hampden Park	[null]
9	2020-06-15	Matchday 1	Aviva Stadium	[null]
10	2020-06-15	Matchday 1	San Mames	[null]
11	2020-06-16	Matchday 1	Puskas Arena	[null]
12	2020-06-16	Matchday 1	Allianz Arena	[null]
13	2020-06-17	Matchday 2	Krestovsky Stadium	[null]
14	2020-06-17	Matchday 2	Olympic Stadium	[null]
15	2020-06-17	Matchday 2	Stadio Olimpico	[null]

#### --Sum the total records in each stages where Team 1 won

SELECT s.name AS Stages,

SUM(CASE WHEN stage\_id = 1 AND team1\_goal > team2\_goal THEN 1 ELSE 0 END) AS Matchday 1,

SUM(CASE WHEN stage\_id = 2 AND team1\_goal > team2\_goal THEN 1 ELSE 0 END) AS Matchday\_2,

SUM(CASE WHEN stage\_id = 3 AND team1\_goal > team2\_goal THEN 1 ELSE 0 END) AS Matchday\_3,

SUM(CASE WHEN stage\_id = 4 AND team1\_goal > team2\_goal THEN 1 ELSE 0 END) AS Matchday 4.

SUM(CASE WHEN stage\_id = 5 AND team1\_goal > team2\_goal THEN 1 ELSE 0 END) AS Matchday 5.

SUM(CASE WHEN stage\_id = 6 AND team1\_goal > team2\_goal THEN 1 ELSE 0 END) AS Matchday 6,

SUM(CASE WHEN stage\_id = 7 AND team1\_goal > team2\_goal THEN 1 ELSE 0 END) AS Matchday\_7

FROM scores

JOIN stages s

ON scores.stage\_id = s.id

**GROUP BY s.name** 

<u>.</u>	stages character varying (15)	matchday_1 bigint	matchday_2 bigint   □	matchday_3 bigint   □	matchday_4 bigint	matchday_5 bigint	matchday_6 bigint	matchday_7 bigint
	Matchday 3	0	0	7	0	0	0	0
2	Round of 16	0	0	0	4	0	0	0
3	Quarter-Final	0	0	0	0	2	0	0
ļ	Semi-Final	0	0	0	0	0	0	0
5	Final	0	0	0	0	0	0	0
5	Matchday 2	0	9	0	0	0	0	0
7	Matchday 1	3	0	0	0	0	0	C

#### -- Count the Team1, Team2 and Draws in each Stadium

SELECT v.name AS venue,

COUNT(CASE WHEN team1\_goal > team2\_goal THEN s.id END) AS team1\_wins,

COUNT(CASE WHEN team1\_goal < team2\_goal THEN s.id END) AS team2\_wins,

COUNT(CASE WHEN team1 goal = team2 goal THEN s.id END) AS Draw

FROM scores s

JOIN venues v

ON s.venue\_id = v.id

**GROUP BY venue** 

4	venue character varying (50) ☐	team1_wins abigint	team2_wins abigint	draw bigint
	Parken Stadium	1	0	3
	Puskas Arena	2	2	0
	San Mames	1	3	0
	Krestovsky Stadium	1	2	1
	Arena Nationala	3	0	1
	Wembley Stadium	3	3	1
	Hampden Park	3	1	0
	Stadio Olimpico	2	2	0
	Aviva Stadium	3	0	1
)	Allianz Arena	2	1	1
	Johan Cruyff Arena	2	2	0
	Olympic Stadium	2	2	0

#### **SUBQUERIES**

## -- Teams who scored 3 or more goals as Team 1

SELECT t.name

FROM teams t

WHERE t.id IN (SELECT team1\_id FROM scores WHERE team1\_goal >= 3)



## --Teams who scored 2 or more goals as Team 2

SELECT t.name

FROM teams t

WHERE t.id IN (SELECT team2\_id FROM scores WHERE team1\_goal >= 3)



#### -- Matches where 5 or more goals were score in a match

SELECT date, team1\_goal, team2\_goal

FROM

(SELECT date, s.name, team1\_goal, team2\_goal, (team1\_goal + team2\_goal) AS

total\_goals

FROM scores

JOIN match m

ON scores.id = m.id

JOIN stages s

ON scores.stage\_id = s.id) AS subq

WHERE total\_goals >= 5

Dat	ta Output	Explain M	essages No	otific
4	date adate □	team1_goal a integer	team2_goal anteger	
1	2020-06-12	2	3	
2	2020-06-14	4	1	
3	2020-06-19	4	2	
4	2020-06-19	5	0	
5	2020-06-20	4	1	
6	2020-06-29	3	2	

#### --Average total of goals per day vs. Overall AVG

SELECT s.name AS stages,

ROUND(AVG(team1\_goal + team2\_goal), 2) AS avg\_goals,

(SELECT ROUND(AVG(team1\_goal + team2\_goal), 2)

FROM scores) AS Overall\_avg

FROM scores

JOIN match AS m

ON scores.id = m.id

JOIN stages s

ON scores.stage\_id = s.id

GROUP BY date, s.name

ORDER BY date

			-			
_	stages character varyin	g (15)	avg_goals numeric	<u></u>	overall_avg numeric	•
1	Matchday 1		5.	.00		2.63
2	Matchday 1		1.	.67		2.63
3	Matchday 1		4.	.00		2.63
4	Matchday 1		2	.00		2.63
5	Matchday 1		3.	.00		2.63
6	Matchday 2		2	.33		2.63
7	Matchday 2		1.	.00		2.63
8	Matchday 2		4.	.33		2.63
9	Matchday 2		3.	.33		2.63
10	Matchday 3		3.	.50		2.63
11	Matchday 3		1.	.50		2.63
12	Matchday 3		2	.50		2.63
13	Matchday 3		2	.00		2.63
14	Round of 16		3.	.00		2.63
15	Round of 16		1	00		2.63

#### --Average total of goals per stage vs Overall AVG

SELECT s.name AS stages,

ROUND(AVG(team1\_goal + team2\_goal), 2) AS avg\_goals,

(SELECT ROUND(AVG(team1\_goal + team2\_goal), 2)

FROM scores) AS Overall\_avg

FROM scores

LEFT JOIN match AS m

ON scores.id = m.id

JOIN stages s

ON scores.stage\_id = s.id

**GROUP BY s.name** 

υaι	a Output Explain	messages	Nonncanons
4	stages character varying (15)	avg_goals numeric	overall_avg numeric
1	Matchday 3	2.17	2.63
2	Round of 16	2.63	2.63
3	Quarter-Final	3.50	2.63
4	Semi-Final	2.50	2.63
5	Final	1.00	2.63
6	Matchday 2	2.75	2.63
7	Matchday 1	2.83	2.63

## --Average total of goals per stage vs difference of overall AVG

SELECT s.name AS stages,

ROUND(AVG(team1\_goal + team2\_goal), 2) AS avg\_goals,

ROUND(AVG(team1\_goal + team2\_goal) -

(SELECT AVG(team1 goal + team2 goal)

FROM scores), 2) AS diff

FROM scores

LEFT JOIN match AS m

ON scores.id = m.id

JOIN stages s

ON scores.stage\_id = s.id

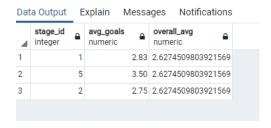
GROUP BY s.name

4	stages character varying (15) <sup>△</sup>	avg_goals numeric	diff numeric
1	Matchday 3	2.17	-0.46
2	Round of 16	2.63	0.00
3	Quarter-Final	3.50	0.87
4	Semi-Final	2.50	-0.13
5	Final	1.00	-1.63
6	Matchday 2	2.75	0.12
7	Matchday 1	2.83	0.21

#### --Stages where average goals > Overall AVG + Overall AVG

**SELECT** 

s.stage\_id, ROUND(s.avg\_goals, 2) AS AVG\_goals,
(SELECT AVG(team1\_goal + team2\_goal) FROM scores) AS overall\_AVG
FROM
(SELECT stage\_id, AVG(team1\_goal + team2\_goal) AS avg\_goals
FROM scores
GROUP BY stage\_id) AS s
WHERE s.avg\_goals > (SELECT AVG(team1\_goal + team2\_goal)
FROM scores)



#### --Correlated subquery (matches where goals > twice the AVG

**SELECT** 

main.stage\_id,
main.team1\_goal,
main.team2\_goal
FROM scores AS main
WHERE
(team1\_goal + team2\_goal) >
(SELECT AVG((sub.team1\_goal + sub.team2\_goal) \* 2)
FROM scores AS sub

WHERE main.stage\_id = sub.stage\_id)

Ĺ	Dat	a Output	Expla	in Message	es Notificatio	ons		
	4	<b>id</b> [PK] integer	ø.	stage_id integer	venue_id integer	team1_goal integer	team2_goal integer	
	1		1019	2	8	4	2	

#### --Scores equal Max number of goals in a match

**SELECT** 

main.id,

main.stage\_id,

main.venue\_id,

main.team1\_goal,

main.team2\_goal

FROM scores AS main

WHERE(team1\_goal + team2\_goal) =

(SELECT MAX(sub.team1\_goal + sub.team2\_goal)

FROM scores AS sub)



## --Scores equal Min number of goals in a match

**SELECT** 

main.id,

main.stage\_id,

main.venue\_id,

main.team1\_goal,

main.team2\_goal

FROM scores AS main

WHERE(team1\_goal + team2\_goal) =

(SELECT MIN(sub.team1\_goal + sub.team2\_goal)

FROM scores AS sub)

4	id [PK] integer	(gar)	stage_id integer	venue_id integer	team1_goal integer	team2_goal integer
1		1003	1	2	0	(
2		1017	2	2	0	
3		1043	4	8	0	

#### -- Comparing Max goals

SELECT st.name, MAX(team1\_goal + team2\_goal) AS max\_goals,
(SELECT MAX(team1\_goal + team2\_goal) FROM scores) AS overall\_max,
(SELECT MAX (team1\_goal + team2\_goal)
FROM scores
WHERE id IN (SELECT id FROM match WHERE EXTRACT(MONTH FROM date) = 06))
AS june\_max\_goals
FROM scores sc
JOIN stages AS st
ON sc.stage\_id = st.id
GROUP BY st.name

4	name character varying (15)   □	max_goals integer	overall_max integer	june_max_goals integer
1	Matchday 3	4	6	6
2	Round of 16	5	6	6
3	Quarter-Final	4	6	6
4	Semi-Final	4	6	6
5	Final	1	6	6
6	Matchday 2	6	6	6
7	Matchday 1	5	6	6

#### --Matches where team1 or team 2 score 2 or more goals

SELECT id, stage\_id, venue\_id FROM scores WHERE team1\_goal >= 2 OR team2\_goal >= 2

id [PK] integer	stage_id integer	venue_id integer	
1001	1	6	
1004	1	10	
1005	1	3	
1006	1	9	
1007	1	7	
1010	1	12	
1011	1	5	
1012	1	4	
1013	2	10	
1015	2	6	
1018	2	7	
1019	2	8	
1020	2	11	
1021	2	3	
1022	2	5	

## --Which stage and venue did those matches take place with > 2 goals?

**SELECT** 

s.name AS stage, v.name AS venue, COUNT(subquery.id) AS matches FROM (

SELECT stage\_id, venue\_id, id

FROM scores

WHERE team1\_goal >= 2 OR team2\_goal >= 2) AS subquery

JOIN stages s

ON s.id = subquery.stage\_id

JOIN venues v

ON v.id = subquery.venue\_id

GROUP BY s.name, v.name

Data	Output Explain Messa	ages Notifications	
_	stage character varying (15)	venue character varying (50)	matches bigint
1	Matchday 2	Krestovsky Stadium	1
2	Matchday 2	Johan Cruyff Arena	1
3	Matchday 1	Wembley Stadium	1
4	Matchday 2	Wembley Stadium	1
5	Matchday 1	Krestovsky Stadium	1
6	Matchday 1	San Mames	1
7	Quarter-Final	Krestovsky Stadium	1
8	Matchday 3	Arena Nationala	1
9	Matchday 3	Hampden Park	1
10	Round of 16	Arena Nationala	1
11	Round of 16	Parken Stadium	1
12	Round of 16	Wembley Stadium	1
13	Matchday 2	Aviva Stadium	1
14	Matchday 1	Johan Cruyff Arena	1
15	Quarter-Final	Stadio Olimpico	1

## --Full scoresheet using a Subquery

SELECT m.date, team1, team2, team1\_goal, team2\_goal

FROM scores s

JOIN match m

ON s.id = m.id

LEFT JOIN (

SELECT match.id, t.name AS team1

FROM match

JOIN teams t

ON match.team1\_id = t.id) AS team1

ON team1.id = s.id

LEFT JOIN (

SELECT match.id, t.name AS team2

FROM match

JOIN teams t

ON match.team2\_id = t.id) AS team2

ON team2.id = s.id

**ORDER BY date** 

Data	Output	Explain Messages	Notifications		
4	date a	team1 character varying (100) <sup>△</sup>	team2 character varying (100) <sup>△</sup>	team1_goal a integer	team2_goal a
1	2020-06-12	Turkey	Italy	2	3
2	2020-06-13	Belgium	Russia	2	2
3	2020-06-13	Wales	Switzerland	0	
4	2020-06-13	Denmark	Finland	0	(
5	2020-06-14	Netherlands	Ukraine	2	
6	2020-06-14	England	Croatia	4	
7	2020-06-14	Austria	Iceland	2	:
8	2020-06-15	Spain	Sweden	1	
9	2020-06-15	Poland	Republic of Ireland	1	(
10	2020-06-15	Norway	Czech Republic	0	
11	2020-06-16	France	Germany	2	
12	2020-06-16	North Macedonia	Portugal	0	:
13	2020-06-17	Turkey	Wales	1	(
14	2020-06-17	Italy	Switzerland	3	
15	2020-06-17	Finland	Russia	0	-

## --Full scoresheet using a correlated subquery

SELECT m.date, team1, team2, team1\_goal, team2\_goal

FROM scores s

JOIN match m

ON s.id = m.id

LEFT JOIN (

SELECT match.id, t.name AS team1

FROM match

JOIN teams t

ON match.team1\_id = t.id) AS team1

ON team1.id = s.id

LEFT JOIN (

SELECT match.id, t.name AS team2

FROM match

JOIN teams t

ON match.team2\_id = t.id) AS team2

ON team2.id = s.id

ORDER BY date

4	date date	team1 character varying (100) ♣	team2 character varying (100) □	team1_goal integer	team2_goal integer
1	2020-06-12	Turkey	Italy	2	3
2	2020-06-13	Belgium	Russia	2	2
3	2020-06-13	Wales	Switzerland	0	1
4	2020-06-13	Denmark	Finland	0	0
5	2020-06-14	Netherlands	Ukraine	2	1
6	2020-06-14	England	Croatia	4	1
7	2020-06-14	Austria	Iceland	2	2
8	2020-06-15	Spain	Sweden	1	3
9	2020-06-15	Poland	Republic of Ireland	1	0
10	2020-06-15	Norway	Czech Republic	0	1
11	2020-06-16	France	Germany	2	2
12	2020-06-16	North Macedonia	Portugal	0	2
13	2020-06-17	Turkey	Wales	1	0
14	2020-06-17	Italy	Switzerland	3	1
15	2020-06-17	Finland	Russia	0	2

#### **COMMON TABLE EXPRESSIONS**

#### --Number of games per stage where total goals are more than / equal to 4

WITH match\_list AS (

**SELECT** 

m.id FROM scores m

WHERE (team1\_goal + team2\_goal) >= 4)

SELECT s.name AS stage, COUNT(match\_list.id) AS matches

FROM scores

JOIN stages AS s

ON scores.stage\_id = s.id

LEFT JOIN match\_list ON scores.id = match\_list.id

**GROUP BY s.name** 

ORDER BY matches DESC



#### -- Matches where 4 or more goals were scored

WITH match\_list AS (

**SELECT** 

scores.stage\_id, m.date, team1\_goal, team2\_goal, (team1\_goal + team2\_goal) AS total\_goals

FROM scores

LEFT JOIN match m ON scores.id = m.id)

SELECT stage\_id, date, team1\_goal, team2\_goal FROM match\_list WHERE total\_goals >= 4

ata Output		Explain I	Messages N	Notifications	
4	stage_id integer	date adate □	team1_goal a integer	team2_goal a integer	
1	1	2020-06-12	2	3	
2	1	2020-06-13	2	2	
3	1	2020-06-14	4	1	
4	1	2020-06-14	2	2	
5	1	2020-06-15	1	3	
6	1	2020-06-16	2	2	
7	2	2020-06-17	3	1	
8	2	2020-06-19	4	2	
9	2	2020-06-19	5	0	
10	2	2020-06-20	0	4	
11	2	2020-06-20	4	1	
12	3	2020-06-21	4	0	
13	3	2020-06-24	0	4	
14	4	2020-06-29	2	2	
15	4	2020-06-29	3	2	

## --Number of goals scored on AVG per venue in June

WITH match\_list AS (

**SELECT** 

venue\_id, (team1\_goal + team2\_goal) AS goals

FROM scores

WHERE id IN (SELECT id FROM match

WHERE EXTRACT(MONTH FROM date) = 06))

SELECT v.name, ROUND(AVG(match\_list.goals), 2)

FROM venues v

LEFT JOIN match\_list

ON v.id = match\_list.venue\_id

GROUP BY v.name

U	154	LEFT JULY MALLI		
	Data	Output Explain I	Messages N	lotificat
	4	name character varying (50)	round numeric	
	1	Hampden Park	2.50	
	2	Johan Cruyff Arena	2.25	
	3	Allianz Arena	3.67	
	4	San Mames	2.50	
	5	Krestovsky Stadium	2.33	
	6	Aviva Stadium	2.00	
	7	Parken Stadium	1.25	
	8	Puskas Arena	2.00	
	9	Arena Nationala	3.25	
	10	Wembley Stadium	3.75	
	11	Stadio Olimpico	4.33	
	12	Olympic Stadium	1.67	

#### --Full scoresheet using CTE

WITH team1 AS (

SELECT s.id, m.date, t.name AS team1, s.team1\_goal

FROM scores s

LEFT JOIN teams t

ON t.id = s.team1 id

LEFT JOIN match m

ON s.id = m.id),

team2 AS (

SELECT s.id, m.date, t.name AS team2, s.team2\_goal

FROM scores s

LEFT JOIN teams t

ON t.id = s.team2\_id

LEFT JOIN match m

On s.id = m.id)

SELECT team1.date, team1.team1, team2.team2, team1.team1\_goal, team2.team2\_goal

FROM team1

JOIN team2

On team1.id = team2.id

ORDER BY date

Data	Ουτρατ	<b>г</b> лрічіі імерридер	Notifications		
4	date adate □	team1 character varying (100) <sup>△</sup>	team2 character varying (100) <sup>△</sup>	team1_goal a integer	team2_goal anteger
1	2020-06-12	Turkey	Italy	2	3
2	2020-06-13	Wales	Switzerland	0	1
3	2020-06-13	Denmark	Finland	0	0
4	2020-06-13	Belgium	Russia	2	2
5	2020-06-14	Netherlands	Ukraine	2	1
6	2020-06-14	England	Croatia	4	1
7	2020-06-14	Austria	Iceland	2	2
8	2020-06-15	Poland	Republic of Ireland	1	0
9	2020-06-15	Spain	Sweden	1	3
10	2020-06-15	Norway	Czech Republic	0	1
11	2020-06-16	North Macedonia	Portugal	0	2
12	2020-06-16	France	Germany	2	2
13	2020-06-17	Finland	Russia	0	2
14	2020-06-17	Italy	Switzerland	3	1
15	2020-06-17	Turkev	Wales	1	0

#### --Full Scoresheet for Netherlands for Euro 2020 CTE

WITH home AS(

SELECT s.id, t.name,

CASE WHEN s.team1\_goal > s.team2\_goal THEN 'Netherlands Win'

WHEN s.team1 goal < s.team2 goal THEN 'Netherlands Lost'

ELSE 'Draw' END AS Netherlands\_Euro\_20

FROM scores s

LEFT JOIN teams t

ON s.team1\_id = t.id),

away AS (

SELECT s.id, t.name,

CASE WHEN s.team1\_goal > s.team2\_goal THEN 'Netherlands Lost'

WHEN s.team1\_goal < s.team2\_goal THEN 'Netherlands Win'

ELSE 'Draw' END AS Netherlands\_Euro\_20

FROM scores s

LEFT JOIN teams t

ON s.team2\_id = t.id)

SELECT DISTINCT m.date, home.name AS team1, away.name AS team2, team1\_goal, team2\_goal,

RANK() OVER(ORDER BY ABS (team1\_goal - team2\_goal) DESC) AS match\_rank FROM scores s

JOIN match m ON s.id = m.id

JOIN home ON s.id = home.id

JOIN away ON s.id = away.id

WHERE home.name = 'Netherlands' OR away.name = 'Netherlands'

4	date a	team1 character varying (100)	team2 character varying (100)	team1_goal integer	team2_goal integer	match_rank bigint   □
1	2020-06	Netherlands	Ukraine	2	1	3
2	2020-06	Netherlands	Austria	2	0	2
3	2020-06	Iceland	Netherlands	0	1	3
4	2020-06	Netherlands	Germany	1	0	3
5	2020-07	Netherlands	Belgium	4	0	1
5	2020-07	England	Netherlands	0	1	3
7	2020-07	Portugal	Netherlands	0	1	3

#### **WINDOW FUNCTIONS**

## --Scorelist with the AVG

SELECT sc.id, v.name AS venue, st.name AS stage, sc.team1\_goal, team2\_goal, AVG(team1\_goal + team2\_goal) OVER() AS overall\_avg FROM scores sc

JOIN venues v ON sc.venue\_id = v.id JOIN stages st ON sc.stage\_id = st.id

4	id integer	venue character varying (50) €	stage character varying (15)	team1_goal integer	team2_goal integer	overall_avg numeric
	1001	Stadio Olimpico	Matchday 1	2	3	2.6274509803921569
	1002	Olympic Stadium	Matchday 1	0	1	2.6274509803921569
	1003	Parken Stadium	Matchday 1	0	0	2.6274509803921569
	1004	Krestovsky Stadium	Matchday 1	2	2	2.6274509803921569
	1005	Wembley Stadium	Matchday 1	4	1	2.6274509803921569
	1006	Arena Nationala	Matchday 1	2	2	2.6274509803921569
	1007	Johan Cruyff Arena	Matchday 1	2	1	2.627450980392156
	1008	Hampden Park	Matchday 1	0	1	2.627450980392156
	1009	Aviva Stadium	Matchday 1	1	0	2.627450980392156
)	1010	San Mames	Matchday 1	1	3	2.627450980392156
	1011	Puskas Arena	Matchday 1	0	2	2.627450980392156
2	1012	Allianz Arena	Matchday 1	2	2	2.627450980392156
3	1013	Krestovsky Stadium	Matchday 2	0	2	2.627450980392156
1	1014	Olympic Stadium	Matchday 2	1	0	2.627450980392156
5	1015	Stadio Olimpico	Matchday 2	3	1	2.6274509803921569

#### --Venue ranking on Average goals

SELECT v.name AS venue,
ROUND(AVG(team1\_goal + team2\_goal), 2) AS avg\_goals,
RANK() OVER(ORDER BY AVG(team1\_goal + team2\_goal)DESC) AS venue\_rank
FROM venues v
JOIN scores ON v.id = scores.venue\_id
GROUP BY v.name
ORDER BY venue\_rank

Data	Output Explain Mess	ages Notificat	ions
4	venue character varying (50)	avg_goals numeric	venue_rank bigint  □
1	Allianz Arena	3.75	1
2	Stadio Olimpico	3.75	1
3	Arena Nationala	3.25	3
4	Wembley Stadium	3.00	4
5	Krestovsky Stadium	2.75	5
6	San Mames	2.50	6
7	Hampden Park	2.50	6
8	Olympic Stadium	2.25	8
9	Johan Cruyff Arena	2.25	8
10	Puskas Arena	2.00	10
11	Aviva Stadium	2.00	10
12	Parken Stadium	1.25	12

## --Stage ranking on Average goals

SELECT s.name AS stage,
ROUND(AVG(team1\_goal + team2\_goal), 2) AS avg\_goals,
RANK() OVER(ORDER BY AVG(team1\_goal + team2\_goal)DESC) AS stage\_rank
FROM stages s
JOIN scores ON s.id = scores.stage\_id
GROUP BY s.name
ORDER BY stage\_rank

4	stage character varying (15)   □	avg_goals numeric	stage_rank bigint
1	Quarter-Final	3.50	1
2	Matchday 1	2.83	2
3	Matchday 2	2.75	3
4	Round of 16	2.63	4
5	Semi-Final	2.50	5
6	Matchday 3	2.17	6
7	Final	1.00	7

#### --Partitioning by a column

SELECT m.date, scores.venue\_id, scores.team1\_goal, scores.team2\_goal, CASE WHEN scores.team1\_id = 1 THEN 'home' ELSE 'away' END AS France, AVG(team1\_goal) OVER(PARTITION BY scores.venue\_id) AS homeavg, AVG(team2\_goal) OVER(PARTITION BY scores.venue\_id) AS awayavg FROM scores
JOIN match m ON scores.id = m.id
JOIN stages s ON scores.stage\_id = s.id
WHERE scores.team1\_id = 1
OR scores.team2\_id = 1
ORDER BY(team1\_goal + team2\_goal) DESC

Da	ta Output	Explain	Messages	Notifications			
4	date ate	venue_id integer	team1_goal a integer	team2_goal anteger	france text	homeavg numeric	awayavg numeric
1	2020-07-04	1	4	0	away	4.00000000000000000	0.0000000000000000000000000000000000000
2	2020-06-13	10	2	2	home	1.0000000000000000000000000000000000000	1.50000000000000000
3	2020-06-27	7	1	2	away	1.0000000000000000000000000000000000000	2.00000000000000000
4	2020-06-22	10	0	1	away	1.0000000000000000000000000000000000000	1.50000000000000000
5	2020-06-18	2	0	0	away	0.0000000000000000000000000000000000000	0.0000000000000000000000000000000000000

## --Assessing Running total of goals and AVG from France when they are team1 SELECT m.date, s.team1\_goal, team2\_goal,

SUM(team1\_goal) OVER(ORDER BY m.date ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS running\_total,

AVG(team1\_goal) OVER(ORDER BY m.date ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS running\_avg

FROM scores s

JOIN match m ON s.id = m.id

WHERE s.team1\_id = 1

Dat	ta Output	Explain	Messages	Notifications			
4	date adate □	venue_id integer	team1_goal a integer	team2_goal anteger	france text	homeavg numeric	awayavg numeric
1	2020-07-04	1	4	0	away	4.00000000000000000	0.0000000000000000000000000000000000000
2	2020-06-13	10	2	2	home	1.0000000000000000000000000000000000000	1.500000000000000000
3	2020-06-27	7	1	2	away	1.0000000000000000000000000000000000000	2.0000000000000000000000000000000000000
4	2020-06-22	10	0	1	away	1.0000000000000000000000000000000000000	1.5000000000000000000000000000000000000
5	2020-06-18	2	0	0	away	0.0000000000000000000000000000000000000	0.0000000000000000000000000000000000000

--Assessing Running total of goals and AVG from France when they are team1 SELECT m.date, s.team1\_goal, team2\_goal,

SUM(team2\_goal) OVER(ORDER BY m.date ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS running\_total,

AVG(team2\_goal) OVER(ORDER BY m.date ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS running\_avg

FROM scores s

JOIN match m ON s.id = m.id

WHERE s.team2\_id = 1

	4	date a	team1_goal a integer	team2_goal a integer	running_total bigint	running_avg numeric	
	1	2020-06-18	0	0	0	0.0000000000000000000000000000000000000	
	2	2020-06-22	0	1	1	0.500000000000000000000	
	3	2020-06-27	1	2	3	1.0000000000000000000000000000000000000	
	4	2020-07-04	4	0	3	0.750000000000000000000	