

# LEFT and RIGHT JOINS

JOINING DATA IN SQL

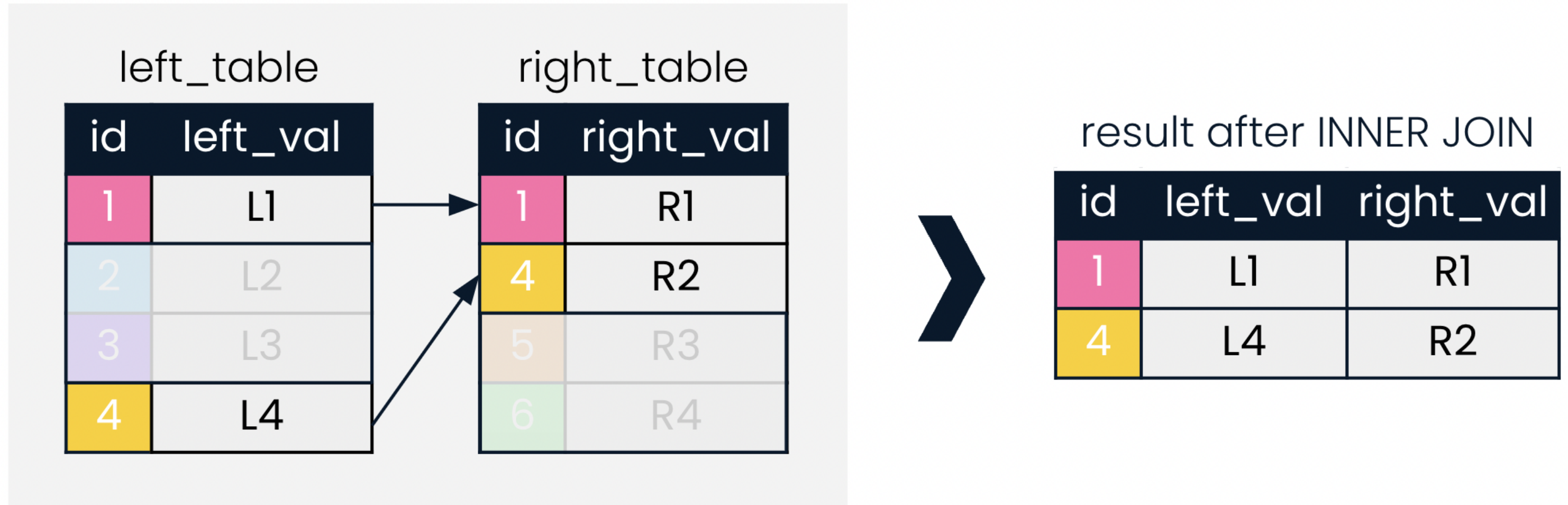


**Maham Faisal Khan**

Senior Content Developer, Data Science

# INNER JOIN diagram

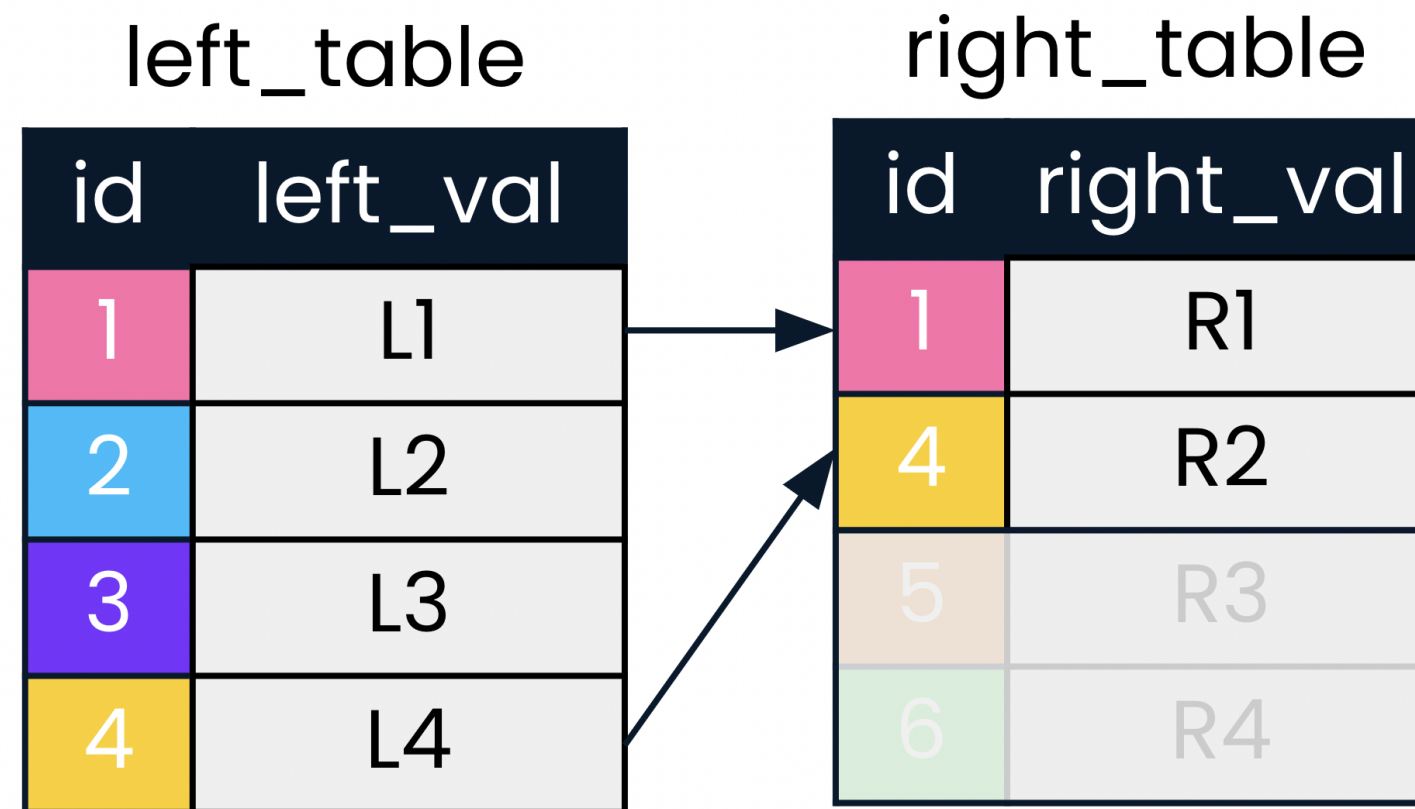
Diagram for an **INNER JOIN** **ON** the **id** field



# LEFT JOIN initial diagram

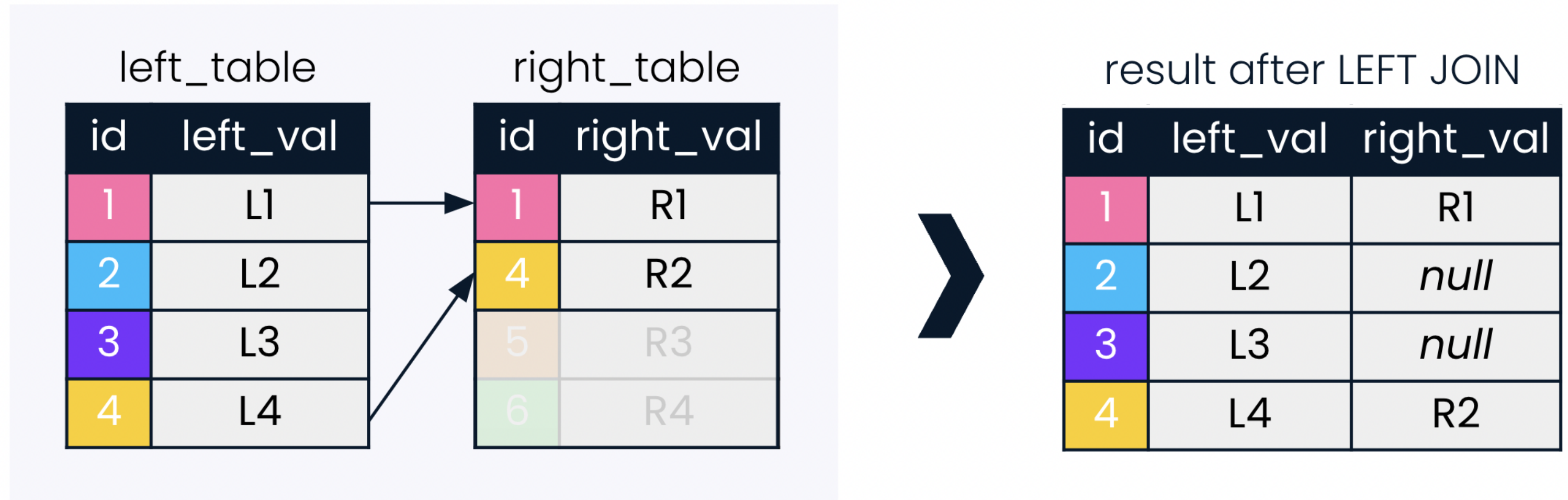
- **LEFT JOIN** will return all records in the left table, and those records in the right table that match on the joining field provided

Diagram for a **LEFT JOIN ON the id field**



# LEFT JOIN diagram

Diagram for a **LEFT JOIN** **ON** the **id** field



# LEFT JOIN syntax

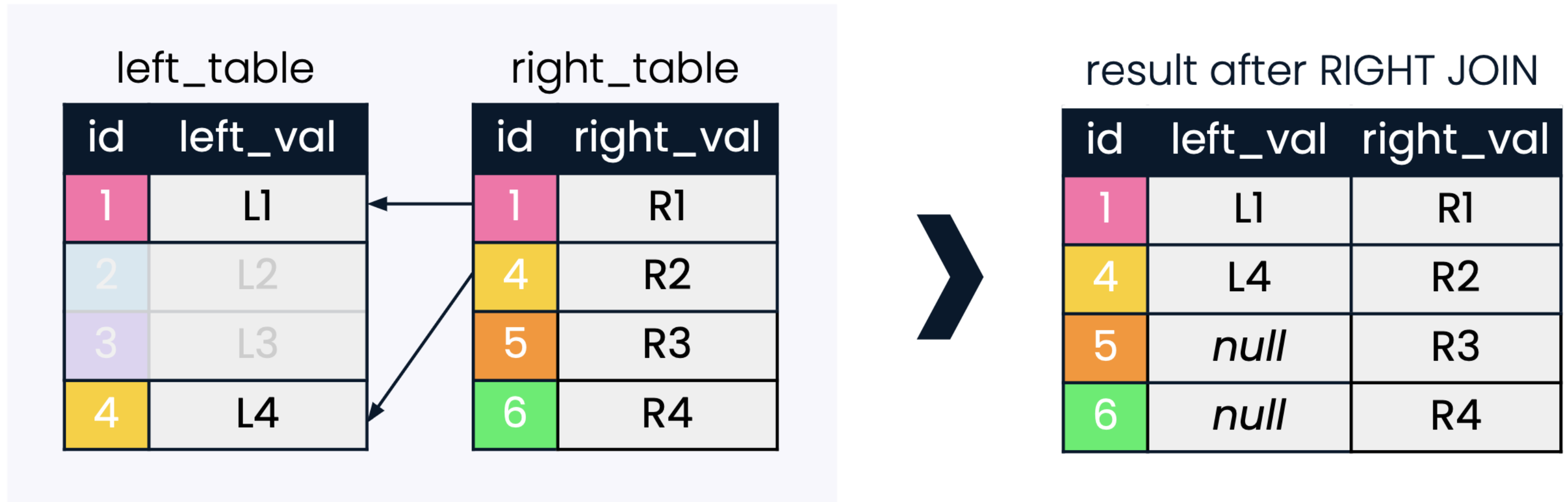
```
SELECT p1.country, prime_minister, president
FROM prime_ministers AS p1
LEFT JOIN presidents AS p2
USING(country);
```

**Note.** `LEFT JOIN` can also be written as `LEFT OUTER JOIN`.

country	prime_minister	president
Egypt	Mostafa Madbouly	Abdel Fattah el-Sisi
Portugal	António Costa	Marcelo Rebelo de Sousa
Pakistan	Shehbaz Sharif	Arif Alvi
United Kingdom	Boris Johnson	null
India	Narendra Modi	Ram Nath Kovind
Australia	Scott Morrison	null
Norway	Jonas Gahr Støre	null
Brunei	Hassanal Bolkiah	null
Oman	Haitham bin Tarik	null

# RIGHT JOIN

Diagram for a **RIGHT JOIN** **ON** the **id** field



# RIGHT JOIN syntax

```
SELECT *  
FROM left_table  
RIGHT JOIN right_table  
ON left_table.id = right_table.id;
```

**Note.** `RIGHT JOIN` can also be written as `RIGHT OUTER JOIN`.

# RIGHT JOIN with presidents and prime ministers

```
SELECT p1.country, prime_minister, president
FROM prime_ministers AS p1
RIGHT JOIN presidents AS p2
USING(country);
```

country	prime_minister	president
Egypt	Mostafa Madbouly	Abdel Fattah el-Sisi
Portugal	António Costa	Marcelo Rebelo de Sousa
USA	null	Joe Biden
Uruguay	null	Luis Lacalle Pou
Pakistan	Shehbaz Sharif	Arif Alvi
Chile	null	Gabriel Boric
India	Narendra Modi	Ram Nath Kovind



# LEFT JOIN or RIGHT JOIN?

- `RIGHT JOIN` is less commonly used than `LEFT JOIN`
- Any `RIGHT JOIN` can be re-written as a `LEFT JOIN`
- `LEFT JOIN` feels more intuitive to users when typing from left to right

# Let's practice!

JOINING DATA IN SQL

# FULL JOINS

JOINING DATA IN SQL



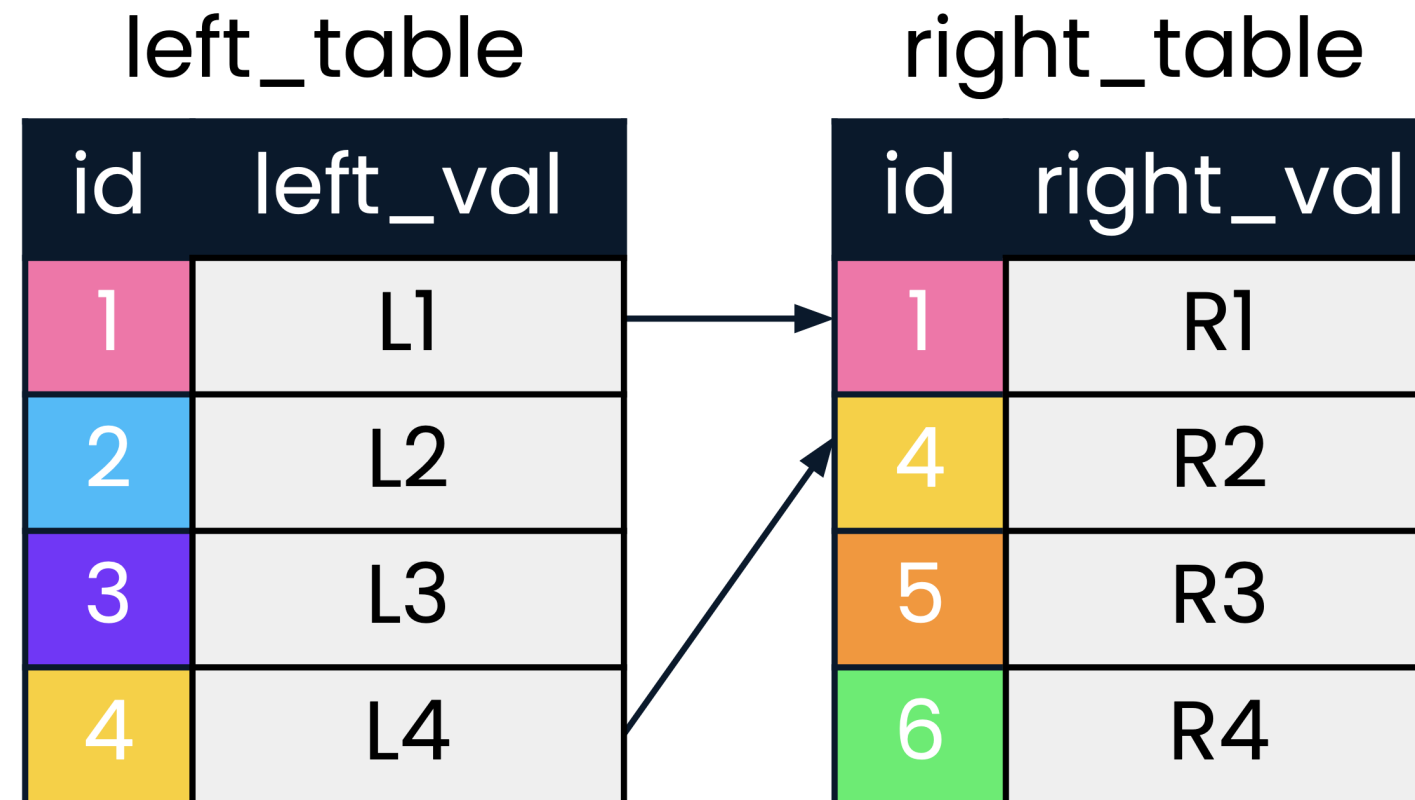
**Maham Faisal Khan**

Senior Content Developer, DataCamp

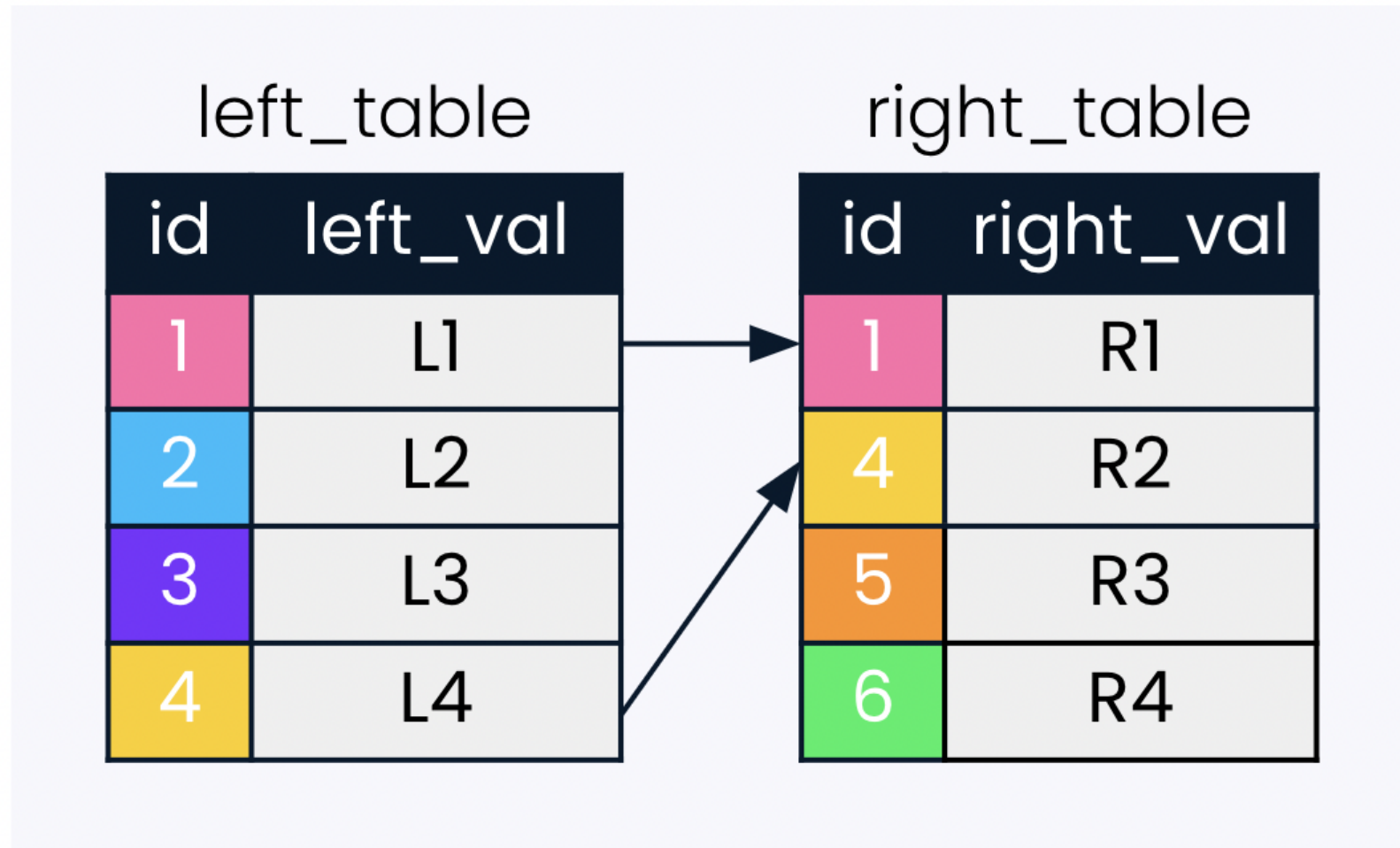
# FULL JOIN initial diagram

- A **FULL JOIN** combines a **LEFT JOIN** and a **RIGHT JOIN**

Diagram for a **FULL JOIN ON the id field**



# FULL JOIN diagram



result after FULL JOIN

id	left_val	right_val
1	L1	R1
2	L2	null
3	L3	null
4	L4	R2
5	null	R3
6	null	R4

# FULL JOIN syntax

```
SELECT left_table.id AS L_id,  
       right_table.id AS R_id,  
       left_table.val AS L_val,  
       right_table.val AS R_val  
FROM left_table  
FULL JOIN right_table  
USING (id);
```

**Note.** The keyword `FULL OUTER JOIN` can also be used.

# FULL JOIN example using leaders database

```
SELECT p1.country AS country, prime_minister, president
```

# FULL JOIN example using leaders database

```
SELECT p1.country AS country, prime_minister, president  
FROM prime_ministers AS p1
```



# FULL JOIN example using leaders database

```
SELECT p1.country AS country, prime_minister, president  
FROM prime_ministers AS p1  
FULL JOIN presidents AS p2
```

# FULL JOIN example using leaders database

```
SELECT p1.country AS country, prime_minister, president
FROM prime_ministers AS p1
FULL JOIN presidents AS p2
ON p1.country = p2.country
LIMIT 10;
```

# FULL JOIN example using leaders database

country	prime_minister	president
-----	-----	-----
Egypt	Mostafa Madbouly	Abdel Fattah el-Sisi
Portugal	António Costa	Marcelo Rebelo de Sousa
Pakistan	Shehbaz Sharif	Arif Alvi
United Kingdom	Boris Johnson	null
USA	null	Joe Biden
Uruguay	null	Luis Lacalle Pou
Chile	null	Gabriel Boric
India	Narendra Modi	Ram Nath Kovind
Norway	Jonas Gahr Støre	null
Brunei	Hassanal Bolkiah	null

# Let's practice!

JOINING DATA IN SQL

# Crossing into CROSS JOIN

JOINING DATA IN SQL

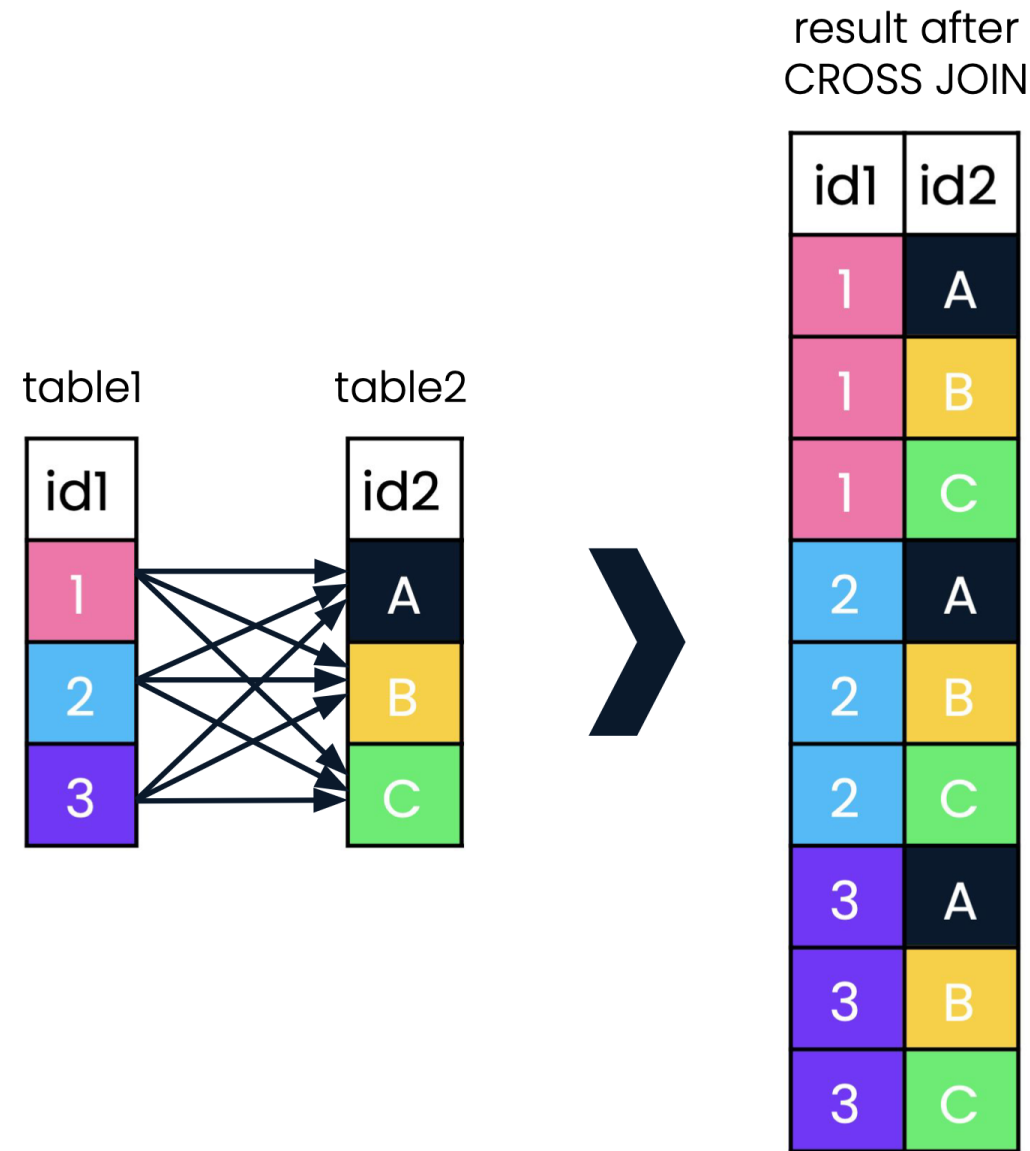


**Maham Faisal Khan**

Senior Content Developer, DataCamp

# CROSS JOIN diagram

**CROSS JOIN** creates all possible combinations of two tables.



# CROSS JOIN syntax

```
SELECT id1, id2  
FROM table1  
CROSS JOIN table2;
```

# Pairing prime ministers with presidents

```
SELECT prime_minister, president
FROM prime_ministers AS p1
CROSS JOIN presidents AS p2
WHERE p1.continent IN ('Asia')
      AND p2.continent IN ('South America');
```

prime_minister	president
Shehbaz Sharif	Luis Lacalle Pou
Narendra Modi	Luis Lacalle Pou
Hassanal Bolkiah	Luis Lacalle Pou
Haitham bin Tarik	Luis Lacalle Pou
Shehbaz Sharif	Gabriel Boric
Narendra Modi	Gabriel Boric
Hassanal Bolkiah	Gabriel Boric
Haitham bin Tarik	Gabriel Boric



# Let's practice!

JOINING DATA IN SQL

# Self joins

## JOINING DATA IN SQL



**Maham Faisal Khan**

Senior Content Developer, DataCamp

# Self joins

- **Self joins** are tables joined with themselves
- They can be used to compare parts of the same table

The `prime_ministers` table from the world leaders database

country	continent	prime_minister
Egypt	Africa	Mostafa Madbouly
Portugal	Europe	António Costa
Pakistan	Asia	Shehbaz Sharif
United Kingdom	Europe	Boris Johnson
India	Asia	Narendra Modi
Australia	Oceania	Scott Morrison
Norway	Europe	Jonas Gahr Støre
Brunei	Asia	Hassanal Bolkiah
Oman	Asia	Haitham bin Tarik
New Zealand	Oceania	Jacinda Ardern

# Prime minister, meet prime minister

```
SELECT
    p1.country AS country1,
    p2.country AS country2,
    p1.continent
FROM prime_ministers AS p1
INNER JOIN prime_ministers AS p2
ON p1.continent = p2.continent
LIMIT 10;
```

# Prime minister, meet prime minister

country1	country2	continent
-----	-----	-----
Egypt	Egypt	Africa
Portugal	Spain	Europe
Portugal	Norway	Europe
Portugal	Portugal	Europe
Vietnam	Oman	Asia
Vietnam	Brunei	Asia
Vietnam	India	Asia
Vietnam	Vietnam	Asia
Haiti	Haiti	North America
India	Oman	Asia
India	Brunei	Asia

# Prime minister, meet prime minister

```
SELECT
    p1.country AS country1,
    p2.country AS country2,
    p1.continent
FROM prime_ministers AS p1
INNER JOIN prime_ministers AS p2
ON p1.continent = p2.continent
   AND p1.country <> p2.country;
```

# The self joined prime ministers table

-----	-----	-----
country1	country2	continent
-----	-----	-----
Portugal	Spain	Europe
Portugal	Norway	Europe
Vietnam	Oman	Asia
Vietnam	Brunei	Asia
Vietnam	India	Asia
India	Oman	Asia
India	Brunei	Asia
India	Vietnam	Asia
Norway	Spain	Europe
Norway	Portugal	Europe
-----	-----	-----

# Let's practice!

JOINING DATA IN SQL