Banco de dados II

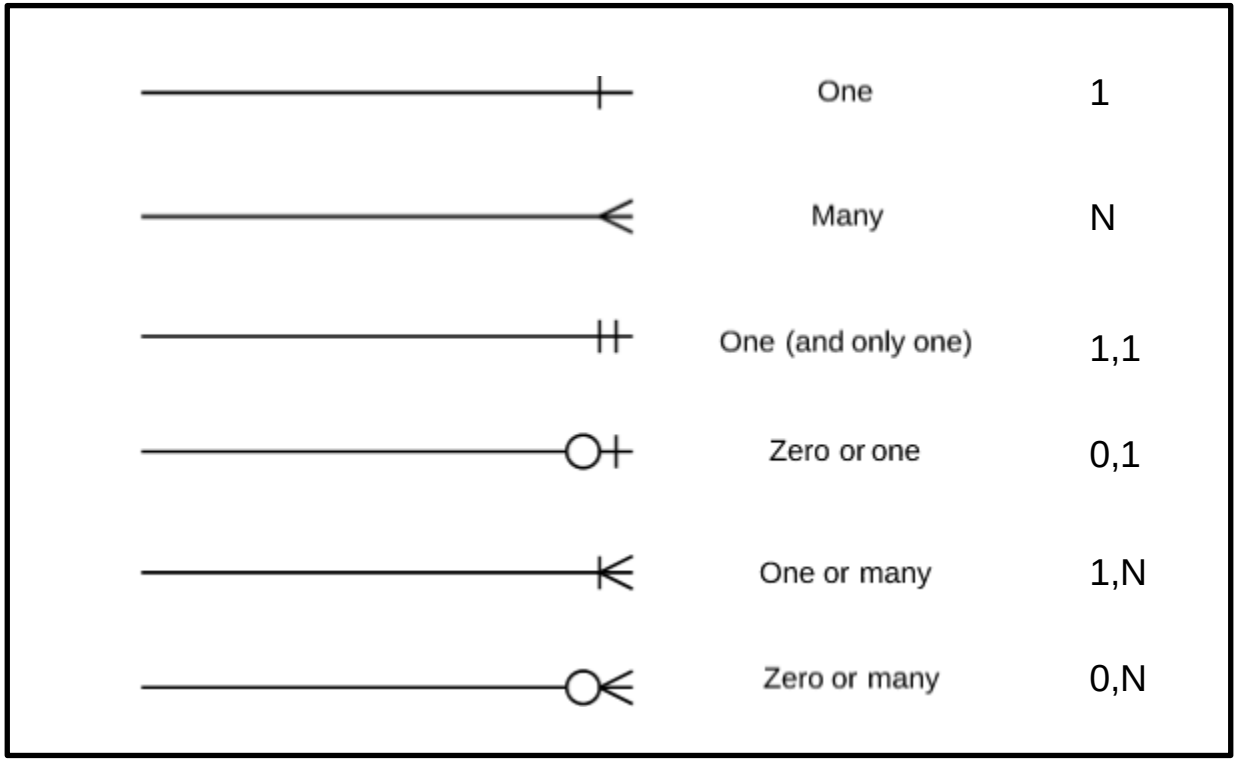
IFSP - Guarulhos

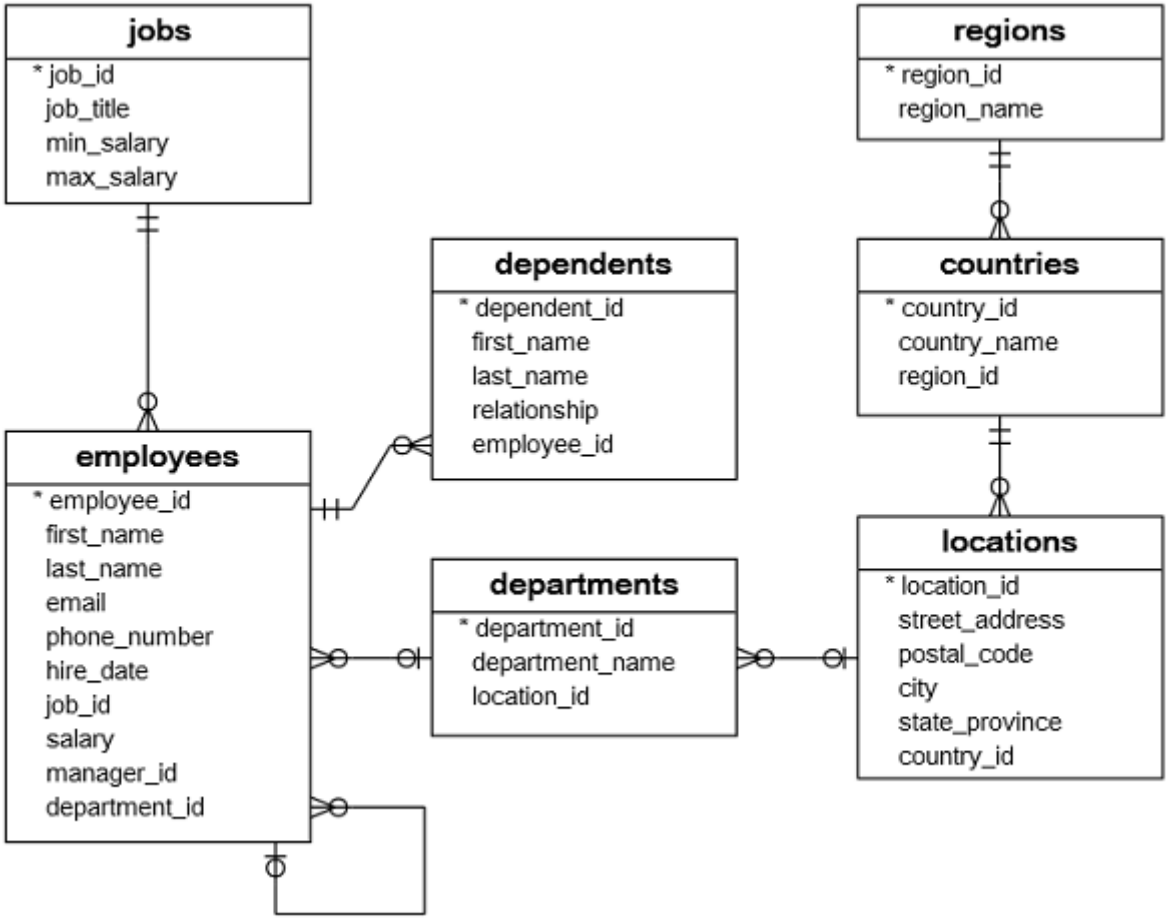
Revisão MySQL

https://www.mysqltutorial.org/mysql-basics/

https://www.sqltutorial.org/sql-sample-database/

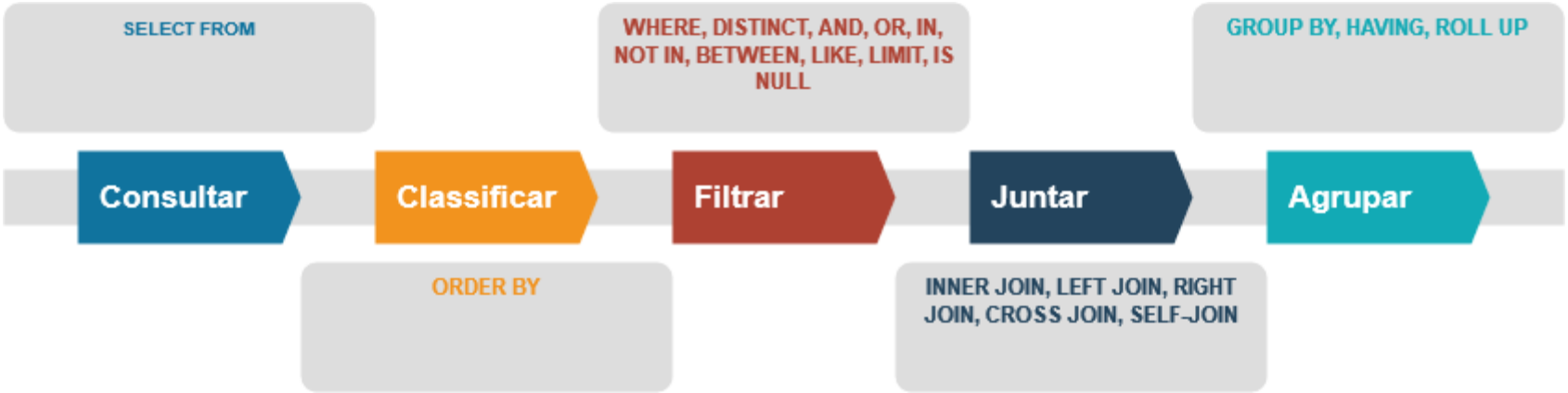
Revisão cardinalidades





https://www.sqltutorial.org/sql-sample-database/

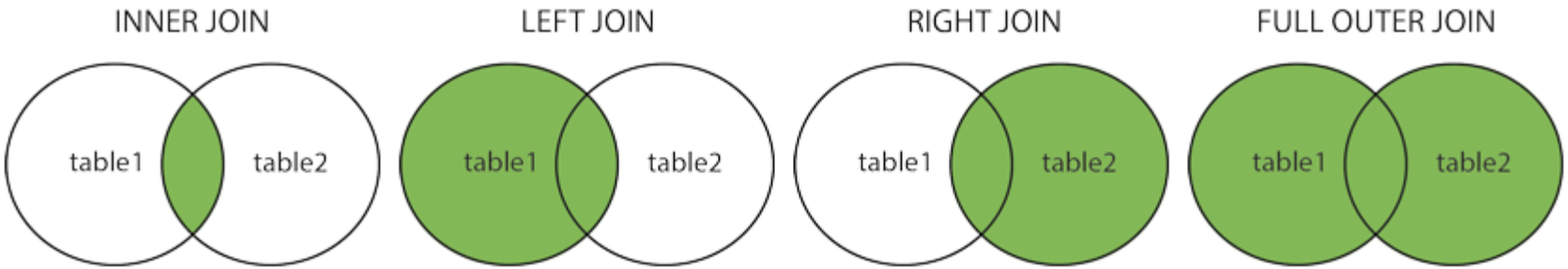
Revisão do básico

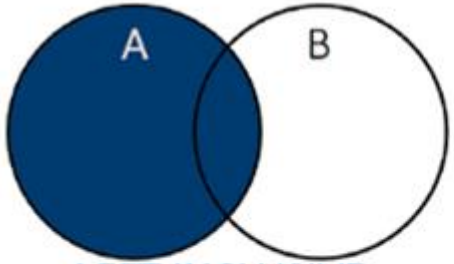


https://www.mysqltutorial.org/mysql-basics/

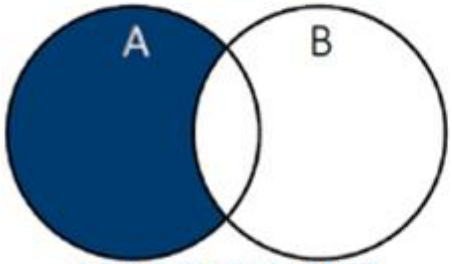
JOIN

(INNER) JOIN	Retorna registros que possuem valores correspondentes em ambas as tabelas
LEFT (OUTER) JOIN	Retorna todos os registros da tabela da esquerda e os registros correspondentes da tabela da direita
RIGHT (OUTER) JOIN	Retorna todos os registros da tabela da direita e os registros correspondentes da tabela da esquerda
FULL (OUTER) JOIN	Retorna todos os registros quando há uma correspondência na tabela esquerda ou direita

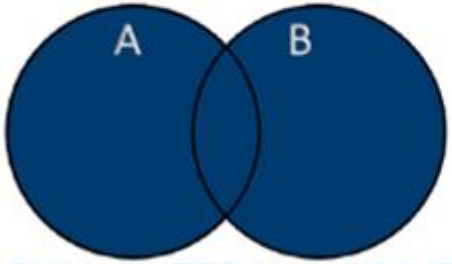




LEFT INCLUSIVE



LEFT EXCLUSIVE



FULL OUTER INCLUSIVE

SQL JOINS

LEFT INCLUSIVE

```
SELECT [Select List]
FROM TableA A
LEFT OUTER JOIN TableB B
ON A.Key= B.Key
```

RIGHT INCLUSIVE

```
SELECT [Select List]
FROM TableA A
RIGHT OUTER JOIN TableB B
ON A.Key= B.Key
```

LEFT EXCLUSIVE

```
SELECT [Select List]
FROM TableA A
LEFT OUTER JOIN TableB B
ON A.Key= B.Key
WHERE B.Key IS NULL
```

RIGHT EXCLUSIVE

```
SELECT [Select List]
FROM TableA A
LEFT OUTER JOIN TableB B
ON A.Key= B.Key
WHERE A.Key IS NULL
```

FULL OUTER INCLUSIVE

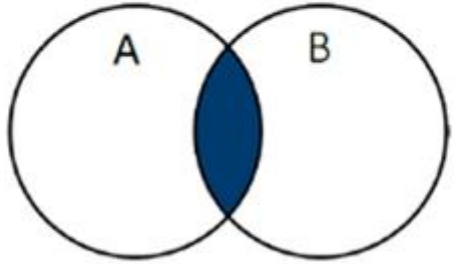
```
SELECT [Select List]
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
```

FULL OUTER EXCLUSIVE

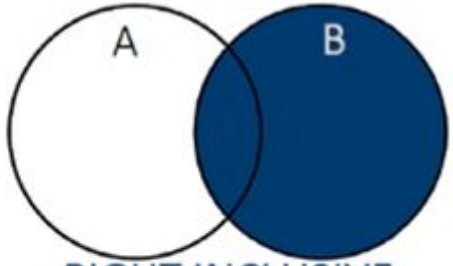
```
SELECT [Select List]
FROM TableA A
FULL OUTER JOIN TableB B
ON A.Key = B.Key
WHERE A.Key IS NULL OR B.Key IS NULL
```

INNER JOIN

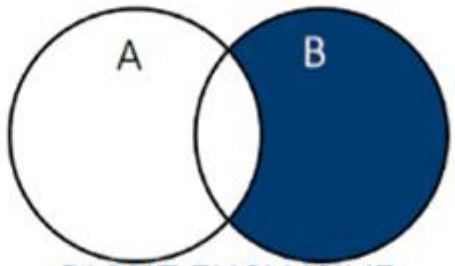
```
SELECT [Select List]
FROM TableA A
INNER JOIN TableB B
ON A.Key = B.Key
```



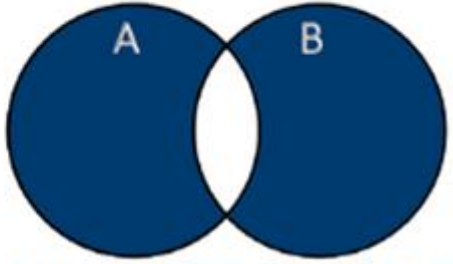
INNER JOIN



RIGHT INCLUSIVE



RIGHT EXCLUSIVE

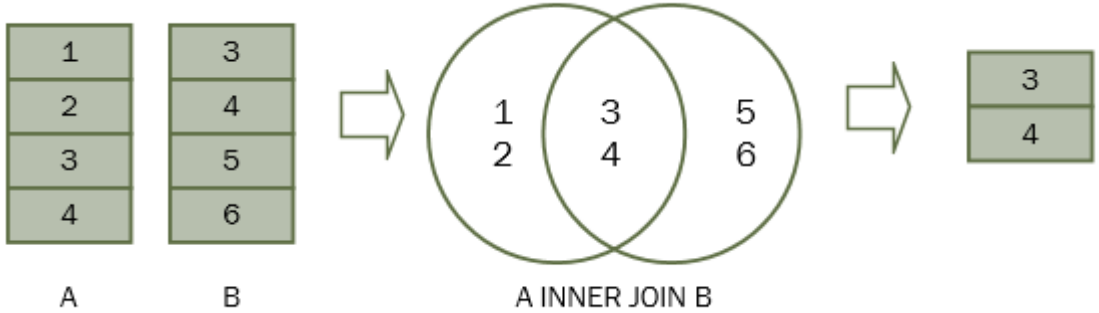


FULL OUTER EXCLUSIVE

INNER JOIN	1 2 3	INNER JOIN	A B C	=	1 B 2 A		Only returns rows that meet the join condition
RIGHT OUTER JOIN	1 2 3	RIGHT OUTER JOIN	A B C	=	1 B 2 A C		Returns all rows from the table on the right side of JOIN and matched rows from the left side of the JOIN
LEFT OUTER JOIN	1 2 3	LEFT OUTER JOIN	A B C	=	1 B 2 A 3		Returns all rows from the table on the left side of JOIN and matched rows from the right side of the JOIN
FULL OUTER JOIN	1 2 3	FULL OUTER JOIN	A B C	=	1 B 2 A 3 C		Returns all rows from both sides even if join condition is not met
CROSS JOIN	1 2 3	CROSS JOIN	A B C	=	1 A 1 B 1 C 2 A 2 B 2 C 3 A 3 B 3 C		Cartesian product between the two sides is a join but without a join condition. Returns all rows joined from both sides

https://cloud.google.com/blog/topics/developers-practitioners/bigquery-explained-working-joins-nested-repeated-data

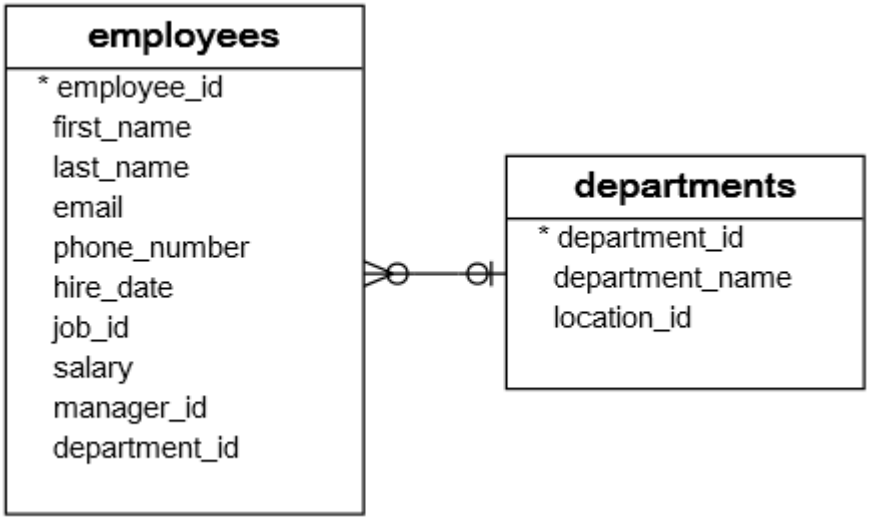
INNER JOIN




```
SELECT a FROM A
        INNER JOIN B
                ON b = a
```

https://www.sqltutorial.org/sql-inner-join/

INNER JOIN



Vamos analisar as cardinalidades do relacionamento entre as duas tabelas...

https://www.sqltutorial.org/sql-inner-join/

INNER JOIN

first_name	last_name	department_id
Jennifer	Whalen	1
Michael	Hartstein	2
Pat	Fay	2
Den	Raphaely	3
Alexander	Khoo	3
Shelli	Baida	3
Sigal	Tobias	3
Guy	Himuro	3
Karen	Colmenares	3

N

department_id	department_name
1	Administration
2	Marketing
3	Purchasing

1

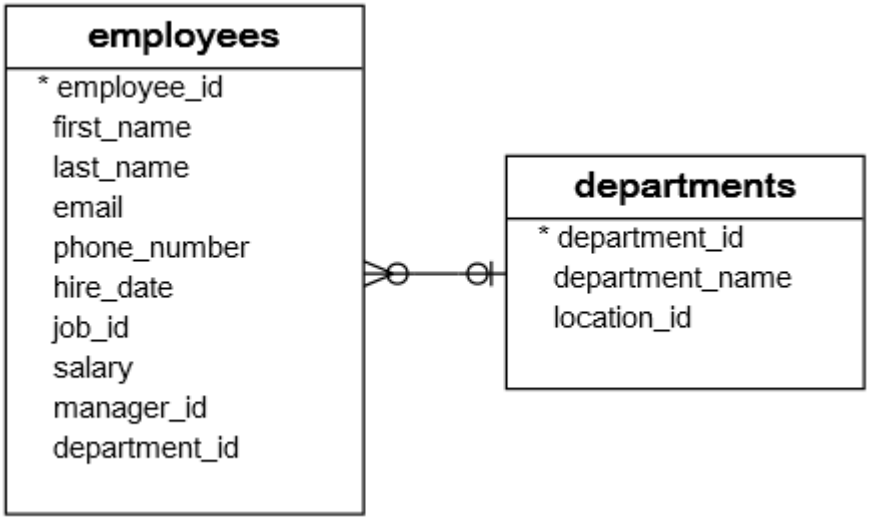
Como obtemos o resultado acima?

Como obtemos o resultado acima?

Vamos analisar as cardinalidades do relacionamento entre as duas tabelas...

https://www.sqltutorial.org/sql-inner-join/

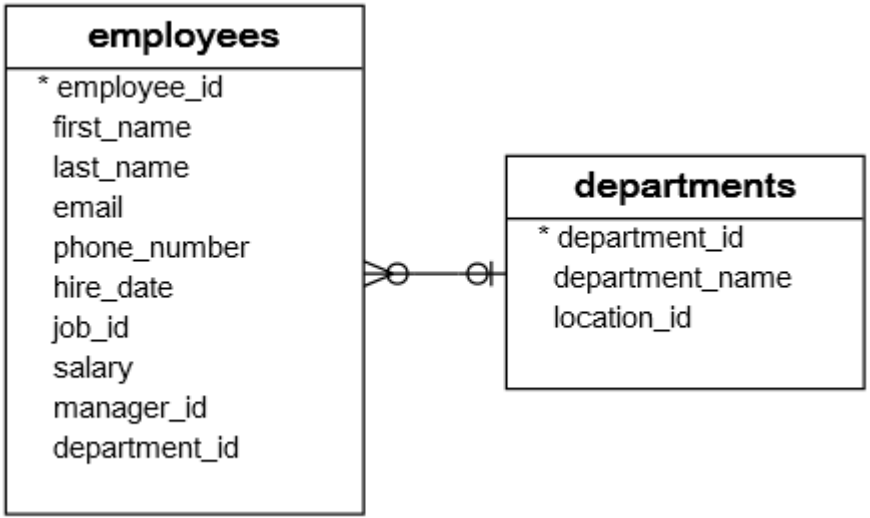
INNER JOIN



Como combinamos dados das 2 tabelas?

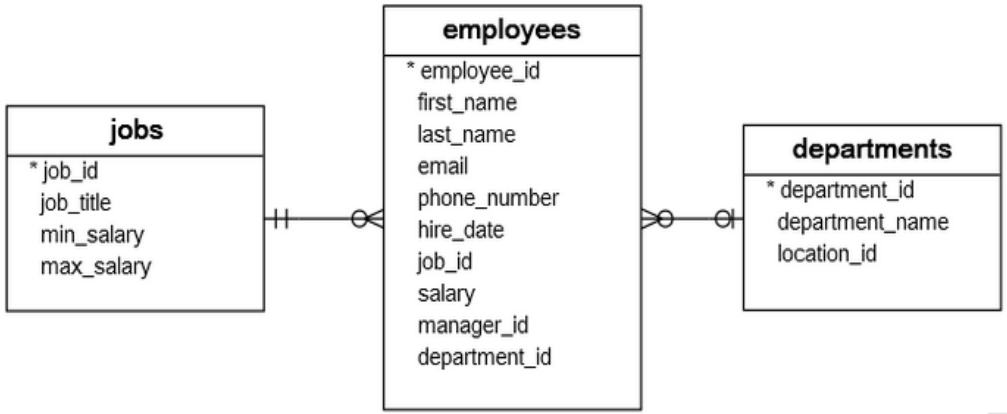
https://www.sqltutorial.org/sql-inner-join/

INNER JOIN




```
SELECT    first_name,    last_name,
          employees.department_id,
          departments.department_id,
          department_name
  FROM    employees
    INNER JOIN    departments
      ON departments.department_id =
          employees.department_id
  WHERE
    employees.department_id IN (1 , 2, 3);
```

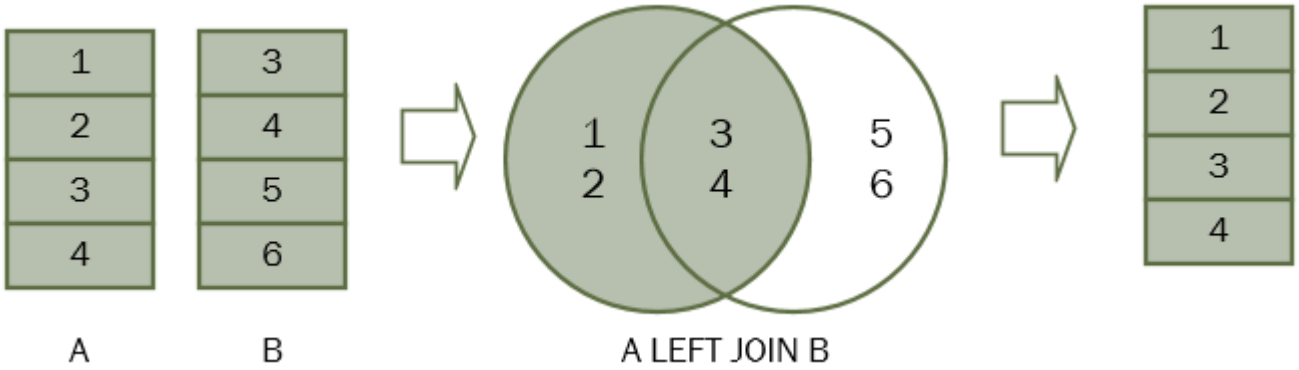
INNER JOIN com mais de 2 tabelas



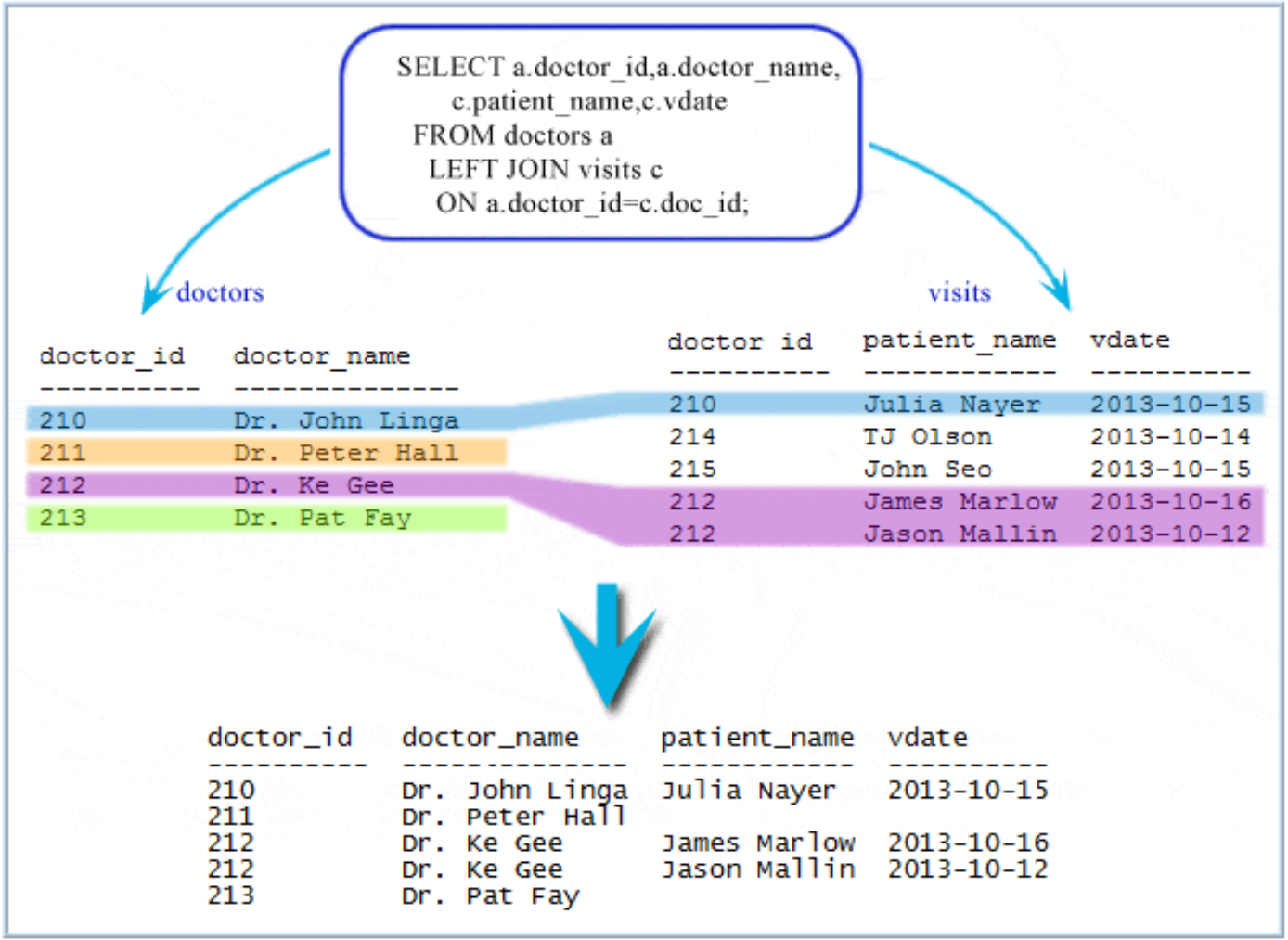
Como obtemos o resultado abaixo?

first_name	last_name	job_title	department_name
Jennifer	Whalen	Administration Assistant	Administration
Michael	Hartstein	Marketing Manager	Marketing
Pat	Fay	Marketing Representative	Marketing
Den	Raphaely	Purchasing Manager	Purchasing
Alexander	Khoo	Purchasing Clerk	Purchasing
Shelli	Baida	Purchasing Clerk	Purchasing
Sigal	Tobias	Purchasing Clerk	Purchasing
Guy	Himuro	Purchasing Clerk	Purchasing
Karen	Colmenares	Purchasing Clerk	Purchasing

LEFT JOIN

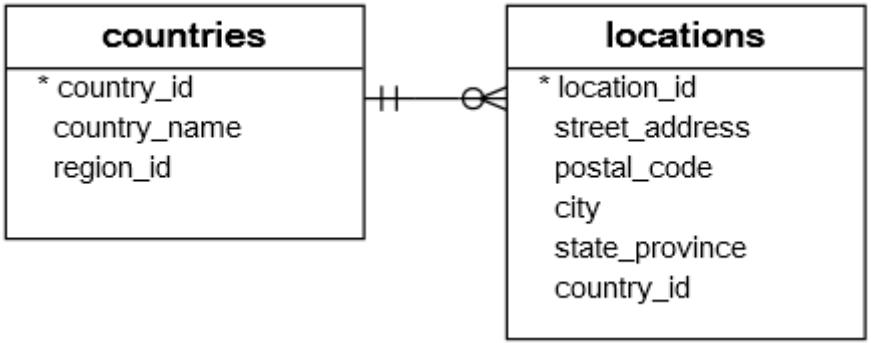



```
SELECT    A.n
  FROM    A
    LEFT JOIN B
  ON
    B.n = A.n
```



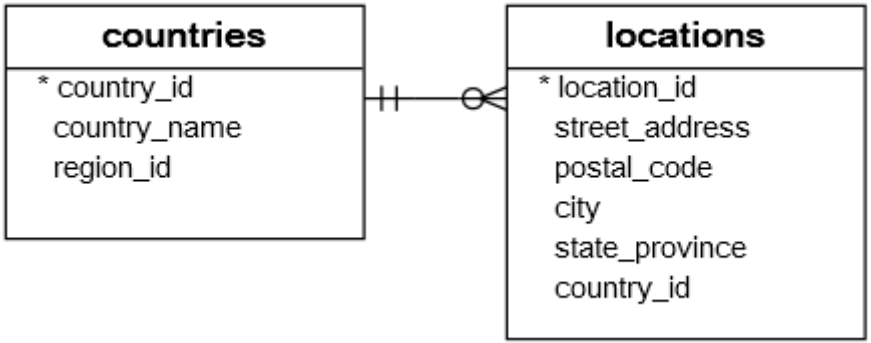
https://www.w3resource.com/sqlite/sqlite-left-join.php

LEFT JOIN



Vamos analisar as cardinalidades? O que ela nos informa?

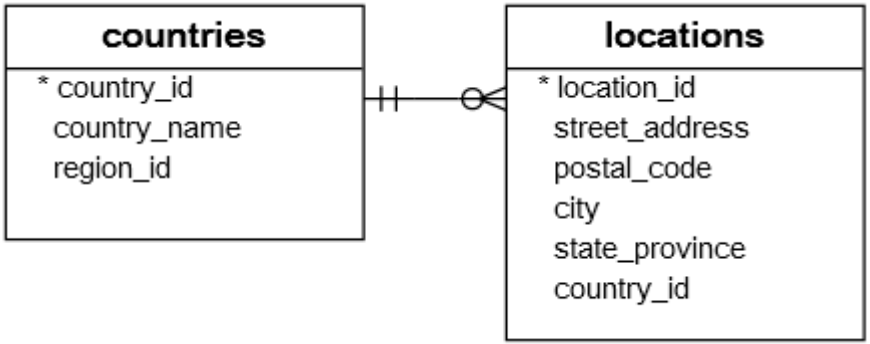
LEFT JOIN



Existem países sem localizações cadastradas na tabela locations?

https://www.sqltutorial.org/sql-left-join/

LEFT JOIN

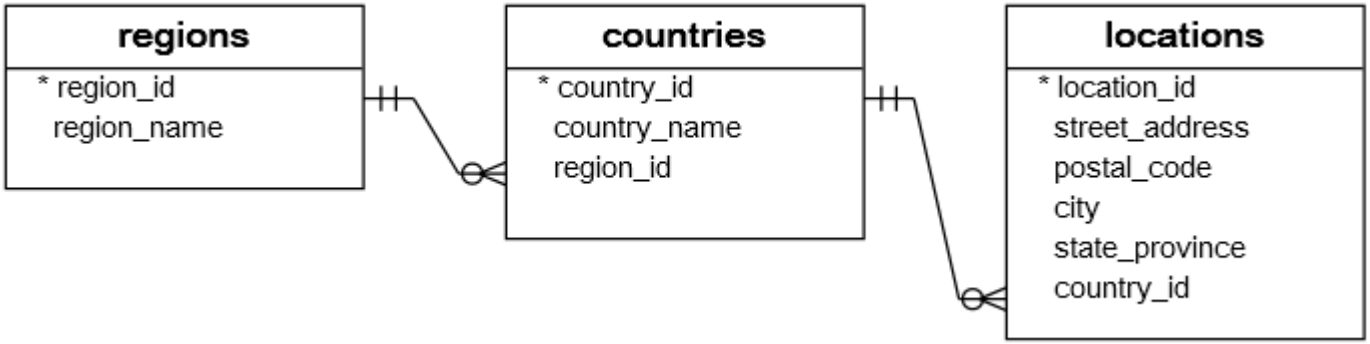


Como obter o resultado abaixo?

country_name	country_id	country_id	street_address	city
United States of America	US	US	2014 Jabberwocky Rd	Southlake
United States of America	US	US	2011 Interiors Blvd	South San Francisco
United States of America	US	US	2004 Charade Rd	Seattle
United Kingdom	UK	UK	8204 Arthur St	London
United Kingdom	UK	UK	Magdalen Centre, The Oxford Science Park	Oxford
China	CN	NULL	NULL	NULL

https://www.sqltutorial.org/sql-left-join/

LEFT JOIN com mais de 2 tabelas...



Como obter o resultado abaixo?

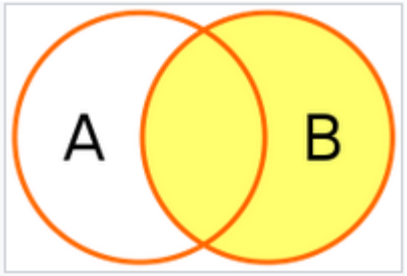
	region_name	country_name	street_address	city
▶	Americas	United States of America	2014 Jabberwocky Rd	Southlake
	Americas	United States of America	2011 Interiors Blvd	South San Francisco
	Americas	United States of America	2004 Charade Rd	Seattle
	Europe	United Kingdom	8204 Arthur St	London
	Europe	United Kingdom	Magdalen Centre, The Oxford Science Park	Oxford
	Asia	China	NULL	NULL

1 2 3

https://www.sqltutorial.org/sql-left-join/

RIGHT JOIN

Existe departamento sem funcionário?

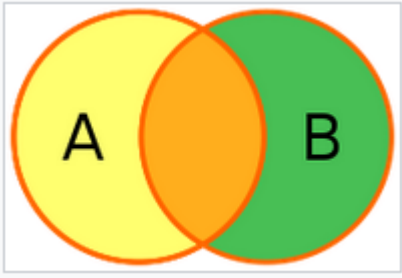



```
SELECT *
FROM empregado RIGHT OUTER JOIN departamento
  ON empregado.IDDepartamento = departamento.IDDepartamento;
```

Empregado.ÚltimoNome	Empregado.IDDepartamento	Departamento.NomeDepartamento	Departamento.IDDepartamento
Smith	34	Administrativo	34
Jones	33	Engenharia	33
Robinson	34	Administrativo	34
Heisenberg	33	Engenharia	33
Rafferty	31	Vendas	31
NULO	NULO	*Marketing*	*35*

https://pt.wikipedia.org/wiki/Join_(SQL)#Jun%C3%A7%C3%A3o_interna_(inner_join)

FULL OUTER JOIN




```
SELECT *
FROM empregado FULL OUTER JOIN departamento
  ON empregado.IDDepartamento = departamento.IDDepartamento;
```

Empregado.ÚltimoNome	Empregado.IDDepartamento	Departamento.NomeDepartamento	Departamento.IDDepartamento
Smith	34	Administrativo	34
Jones	33	Engenharia	33
Robinson	34	Administrativo	34
Williams	NULO	NULO	NULO
Heisenberg	33	Engenharia	33
Rafferty	31	Administrativo	31
NULO	NULO	*Marketing*	*35*

https://pt.wikipedia.org/wiki/Join_(SQL)#Jun%C3%A7%C3%A3o_interna_(inner_join)

Mais alguns exemplos

https://www.devmedia.com.br/inner-cross-left-rigth-e-full-joins/21016
https://www.sqltutorial.org/sql-sample-database/
https://www.mysqltutorial.org/mysql-basics/

Referências Bibliográficas

https://www.sqltutorial.org/sql-sample-database/
https://www.sqltutorial.org/sql-left-join/
https://www.sqltutorial.org/wp-content/uploads/2020/04/mysql.txt
https://www.sqltutorial.org/wp-content/uploads/2020/04/mysql-data.txt

https://www.mysqltutorial.org/mysql-join/

https://www.w3schools.com/sql/sql_join.asp
https://www.w3resource.com/sqlite/sqlite-left-join.php

http://www.cadcobol.com.br/db2_join_tech_agilist_inner_joins_and_outer_joins.htm
https://pt.wikipedia.org/wiki/Join_(SQL)
https://www.devmedia.com.br/inner-cross-left-rigth-e-full-joins/21016

https://dev.mysql.com/doc/index-other.html
https://www.javatpoint.com/mysql-common-table-expression