What are JOINS?

JOINS

- ✓ **Combine information from multiple tables in one query**

	payment_id integer	customer_id smallint	staff_id smallint	rental_id integer	amount numeric (5,2)	payment_date timestamp with time zone
1	1605	269	2	7	1.99	2020-01-24 22:40:19.996577+01
2	1605	269	1	98	0.99	2020-01-25 16:16:50.996577+01
3	1605	269	2	678	6.99	2020-01-28 22:44:14.996577+01

customer_id [PK] integer	store_id smallint	first_name text	last_name text	email text
269	1	CASSANDRA	WALTERS	CASSANDRA.WALTERS@sakilacustomer.org

	payment_id	customer_id	staff_id	rental_id	amount numeric (5,2)	payment_date timestamp with time zone	first_name text	last_name text
				7	1.99	2020-01-24 22:40:19.996577+01	CASSANDRA	WALTERS
				98	0.99	2020-01-25 16:16:50.996577+01	CASSANDRA	WALTERS
3	16052	269	2	678	6.99	2020-01-28 22:44:14.996577+01	CASSANDRA	WALTERS

	customer_id [PK] integer	store_id smallint	first_name text	last_name text	email text
1	1	1	MARY	SMITH	MARY.SMITH@sakilacustomer.org
2	2	1	PATRICIA	JOHNSON	PATRICIA.JOHNSON@sakilacustomer.org
3	3	1	LINDA	WILLIAMS	LINDA.WILLIAMS@sakilacustomer.org

What are Joins

- ✓ How Joins work and how we use them practically

INNER JOIN

OUTER JOIN

LEFT JOIN

RIGHT JOIN

What are Joins

- ✓ Very important technique in SQL
- ✓ Can seem a bit complicated at first
- ✓ Theory + practice
- ✓ Many examples + practical tips + challenges

Inner Join - Theory

INNER JOIN

- ✓ **Inner Join easiest join type**
- ✓ **Helps to understand the general concept of joins**

INNER JOIN

sales table

employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	Munich	100
Frank	Frankfurt	100

bonus table

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	NO

- ✓ Combine the two tables in one query
- ✓ One common column – join column

INNER JOIN

sales table

employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
~~Michael~~	~~Munich~~	~~100~~
~~Frank~~	~~Frankfurt~~	~~100~~

bonus table

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	NO

Do not appear in the bonus table

INNER JOIN

sales table

employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
~~Michael~~	~~Munich~~	~~100~~
~~Frank~~	~~Frankfurt~~	~~100~~

Do not appear in the bonus table

bonus table

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
~~Simon~~	~~NO~~

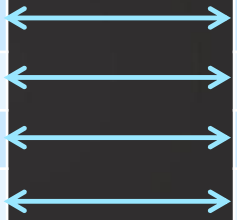
Do not appear in the sales table

INNER JOIN

sales table

employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
~~Michael~~	~~Munich~~	~~100~~
~~Frank~~	~~Frankfurt~~	~~100~~

Do not appear in the bonus table



bonus table

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
~~Simon~~	~~NO~~

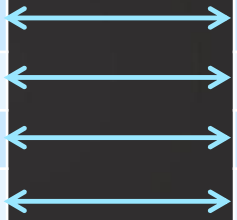
Do not appear in the sales table

INNER JOIN

sales table

employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
~~Michael~~	~~Munich~~	~~100~~
~~Frank~~	~~Frankfurt~~	~~100~~

Do not appear in the bonus table



bonus table

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
~~Simon~~	~~NO~~

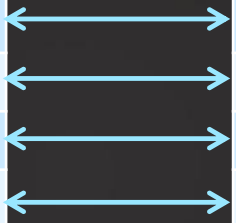
Do not appear in the sales table

✓ **INNER JOIN: Only rows appear in <u>both</u> tables**

INNER JOIN

sales table

employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
~~Michael~~	~~Munich~~	~~100~~
~~Frank~~	~~Frankfurt~~	~~100~~



bonus table

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
~~Simon~~	~~NO~~

employee	city	sales	bonus

INNER JOIN

sales table

employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
~~Michael~~	~~Munich~~	~~100~~
~~Frank~~	~~Frankfurt~~	~~100~~

bonus table

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
~~Simon~~	~~NO~~

employee	city	sales	bonus
Sandra	Frankfurt	500	

INNER JOIN

sales table

employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
~~Michael~~	~~Munich~~	~~100~~
~~Frank~~	~~Frankfurt~~	~~100~~

bonus table

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
~~Simon~~	~~NO~~

employee	city	sales	bonus
Sandra	Frankfurt	500	YES

INNER JOIN

sales table

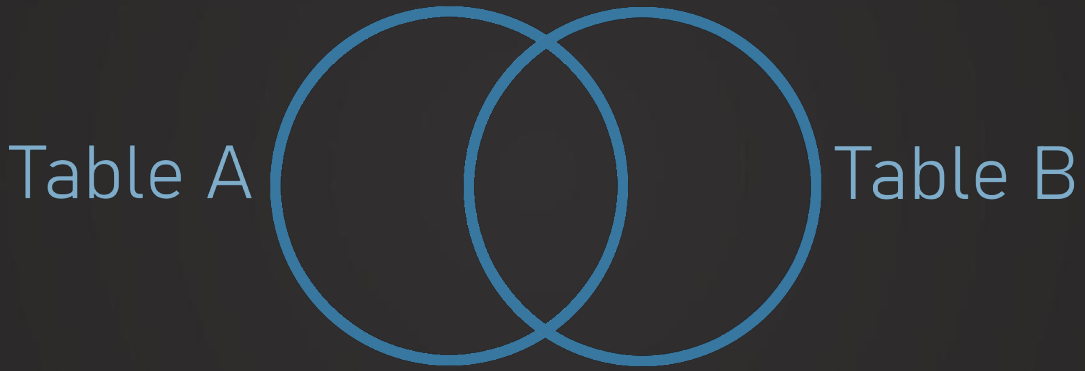
employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
~~Michael~~	~~Munich~~	~~100~~
~~Frank~~	~~Frankfurt~~	~~100~~

bonus table

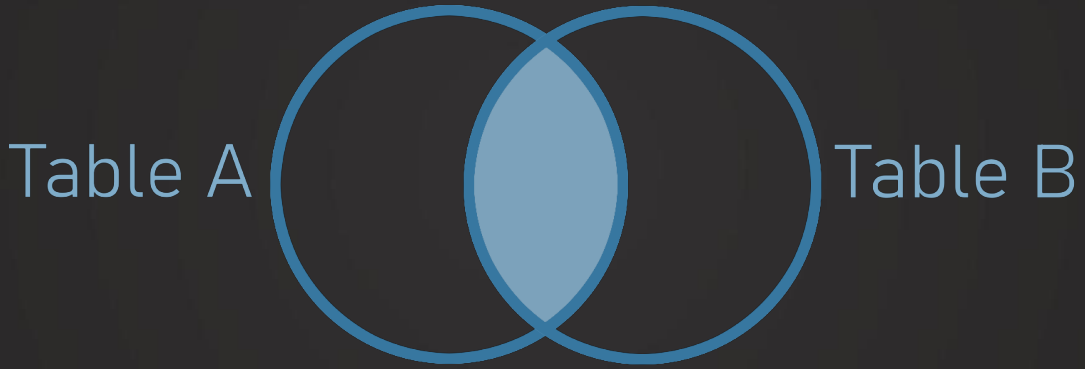
employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
~~Simon~~	~~NO~~

employee	city	sales	bonus
Sandra	Frankfurt	500	YES
Sabine	Munich	300	YES
Peter	Hamburg	200	NO
Manuel	Hamburg	400	YES

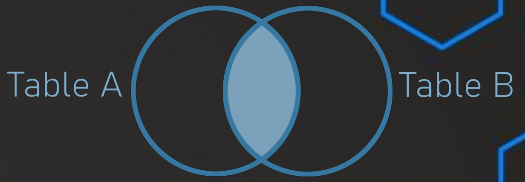
INNER JOIN



INNER JOIN



INNER JOIN

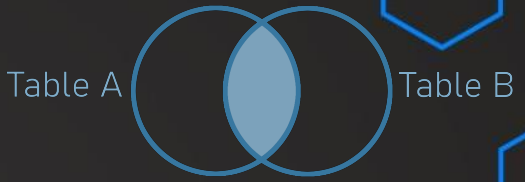


employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
~~Michael~~	~~Munich~~	~~100~~
~~Frank~~	~~Frankfurt~~	~~100~~

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
~~Simon~~	~~NO~~

employee	city	sales	bonus
Sandra	Frankfurt	500	YES
Sabine	Munich	300	YES
Peter	Hamburg	200	NO
Manuel	Hamburg	400	YES

INNER JOIN

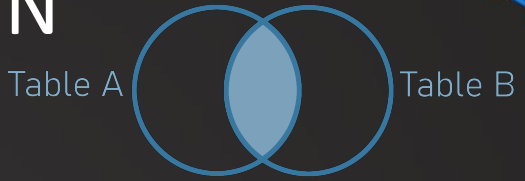


employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	**Hamburg**	**200**
Peter	**Munich**	**250**
Manuel	Hamburg	400
~~Michael~~	~~Munich~~	~~100~~
~~Frank~~	~~Frankfurt~~	~~100~~

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
~~Simon~~	~~NO~~

employee	city	sales	bonus
Sandra	Frankfurt	500	YES
Sabine	Munich	300	YES
Peter	Hamburg	200	NO
Manuel	Hamburg	400	YES

INNER JOIN

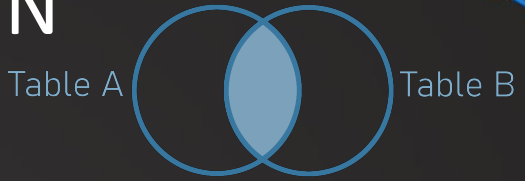


employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	**Hamburg**	**200**
Peter	**Munich**	**250**
Manuel	Hamburg	400
~~Michael~~	~~Munich~~	~~100~~
~~Frank~~	~~Frankfurt~~	~~100~~

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
~~Simon~~	~~NO~~

employee	city	sales	bonus
Sandra	Frankfurt	500	YES
Sabine	Munich	300	YES
Peter	**Hamburg**	**200**	**NO**
Peter	**Munich**	**250**	**NO**
Manuel	Hamburg	400	YES

INNER JOIN

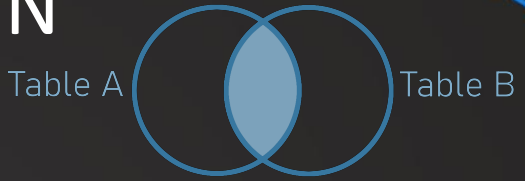


employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	**Hamburg**	**200**
Peter	**Munich**	**250**
Manuel	Hamburg	400
~~Michael~~	~~Munich~~	~~100~~
~~Frank~~	~~Frankfurt~~	~~100~~

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
~~Simon~~	~~NO~~

employee	**-**	**-**	**bonus**
Sandra	-	-	YES
Sabine	-	-	YES
Peter	**-**	**-**	**NO**
Peter	**-**	**-**	**NO**
Manuel	-	-	YES

INNER JOIN

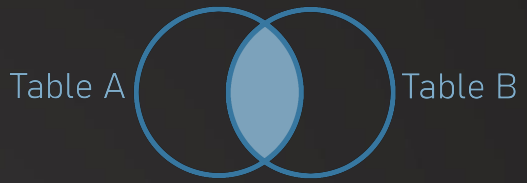


employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	**Hamburg**	**200**
Peter	**Munich**	**250**
Manuel	Hamburg	400
~~Michael~~	~~Munich~~	~~100~~
~~Frank~~	~~Frankfurt~~	~~100~~

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
~~Simon~~	~~NO~~

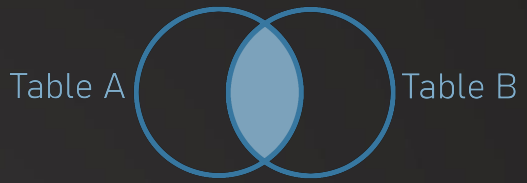
employee	city	sales	bonus
Sandra	Frankfurt	500	YES
Sabine	Munich	300	YES
Peter	**Hamburg**	**200**	**NO**
Peter	**Munich**	**250**	**NO**
Manuel	Hamburg	400	YES

SYNTAX



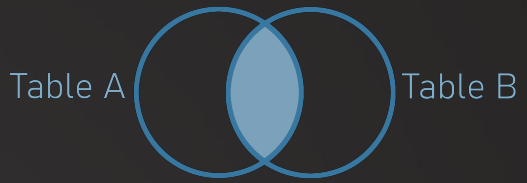

```
SELECT * FROM TableA
INNER JOIN TableB
ON TableA.employee = TableB.employee
```

SYNTAX




```
SELECT * FROM TableB
INNER JOIN TableA
ON TableA.employee = TableB.employee
```

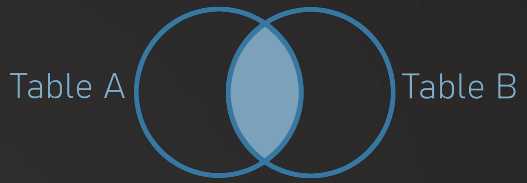
SYNTAX




```
SELECT * FROM TableA AS A
INNER JOIN TableB AS B
ON A.employee = B.employee
```

✓ **Aliases help with writing & reading the code more easily**

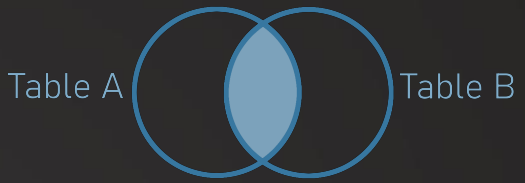
SYNTAX




```
SELECT * FROM TableA A
INNER JOIN TableB B
ON A.employee = B.employee
```

✓ **Aliases help with writing & reading the code more easily**

SYNTAX

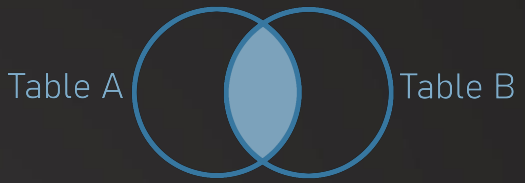



```
SELECT employee FROM TableA A
INNER JOIN TableB B
ON A.employee = B.employee
```

employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	Munich	100
Frank	Frankfurt	100

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	NO

SYNTAX

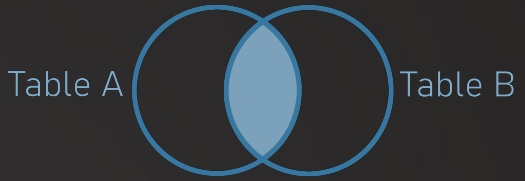



```
SELECT A.employee FROM TableA A
INNER JOIN TableB B
ON A.employee = B.employee
```

employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	Munich	100
Frank	Frankfurt	100

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	NO

SYNTAX




```
SELECT A.employee, sales FROM TableA A
INNER JOIN TableB B
ON A.employee = B.employee
```

employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	Munich	100
Frank	Frankfurt	100

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	NO

SYNTAX

employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	Munich	100
Frank	Frankfurt	100

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	NO

- ✓ Always need a common column / reference
- ✓ INNER JOIN: Only rows where reference column value is in both tables
- ✓ Order of tables (A and B / B and A) does not matter
- ✓ Repeated values in either table will also be repeated

Challenge

The airline company wants to understand in which category they sell most tickets.

How many people choose seats in the category
- Business
- Economy or
- Comfort?

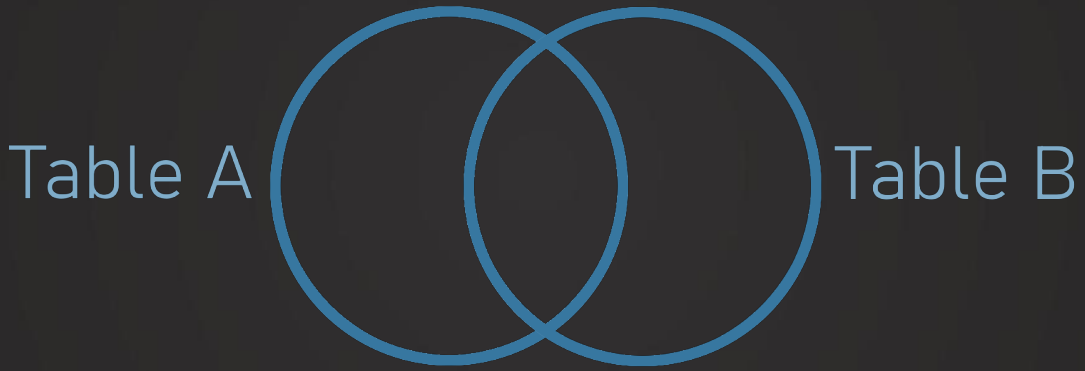
You need to work on the seats table and the boarding_passes table.

Result

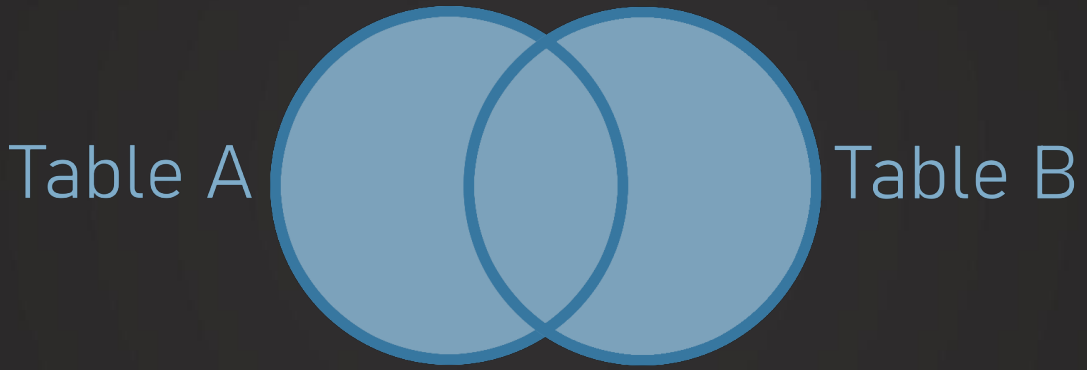
	fare_conditions character varying (10)	count bigint
1	Business	621092
2	Comfort	112098
3	Economy	2476771

FULL OUTER JOIN

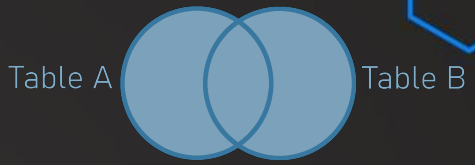
FULL OUTER JOIN



FULL OUTER JOIN



FULL OUTER JOIN



sales table

employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	**Munich**	**100**
Frank	**Frankfurt**	**100**

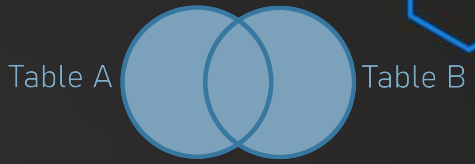
Do not appear in the bonus table

bonus table

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	**NO**

Do not appear in the sales table

FULL OUTER JOIN

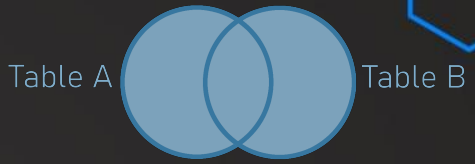


employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	**Munich**	**100**
Frank	**Frankfurt**	**100**

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	**NO**

employee	city	sales	bonus
Sandra	Frankfurt	500	YES
Sabine	Munich	300	YES
Peter	Hamburg	200	NO
Manuel	Hamburg	400	YES

FULL OUTER JOIN

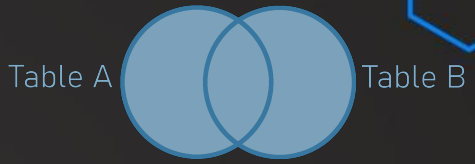


employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	**Munich**	**100**
Frank	**Frankfurt**	**100**

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	**NO**

employee	city	sales	bonus
Sandra	Frankfurt	500	YES
Sabine	Munich	300	YES
Peter	Hamburg	200	NO
Manuel	Hamburg	400	YES
Michael	**Munich**	**100**	
Frank	**Frankfurt**	**100**	

FULL OUTER JOIN

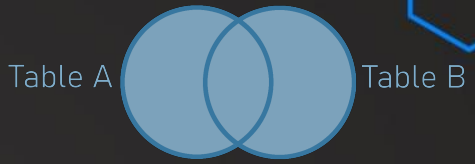


employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	**Munich**	**100**
Frank	**Frankfurt**	**100**

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	**NO**

employee	city	sales	bonus
Sandra	Frankfurt	500	YES
Sabine	Munich	300	YES
Peter	Hamburg	200	NO
Manuel	Hamburg	400	YES
Michael	**Munich**	**100**	*null*
Frank	**Frankfurt**	**100**	*null*

FULL OUTER JOIN

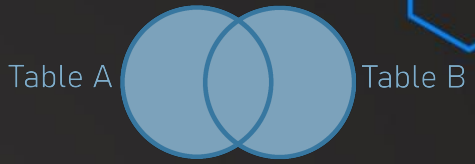


employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	**Munich**	**100**
Frank	**Frankfurt**	**100**

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	**NO**

employee	city	sales	bonus
Sandra	Frankfurt	500	YES
Sabine	Munich	300	YES
Peter	Hamburg	200	NO
Manuel	Hamburg	400	YES
Michael	**Munich**	**100**	*null*
Frank	**Frankfurt**	**100**	*null*
	null	*null*	**NO**

FULL OUTER JOIN

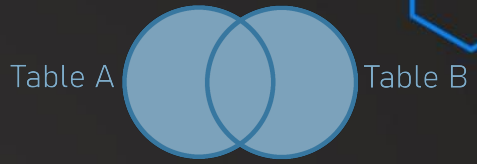


employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	**Munich**	**100**
Frank	**Frankfurt**	**100**

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	**NO**

employee	city	sales	bonus
Sandra	Frankfurt	500	YES
Sabine	Munich	300	YES
Peter	Hamburg	200	NO
Manuel	Hamburg	400	YES
Michael	**Munich**	**100**	*null*
Frank	**Frankfurt**	**100**	*null*
null	*null*	*null*	**NO**

FULL OUTER JOIN

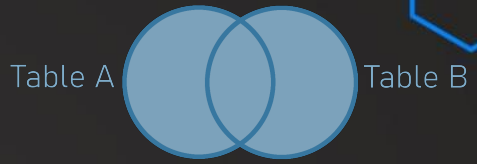


employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	**Munich**	**100**
Frank	**Frankfurt**	**100**

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	**NO**

bonus.employee	sales.employee	city	sales	bonus
Sandra	Sandra	Frankfurt	500	YES
Sabine	Sabine	Munich	300	YES
Peter	Peter	Hamburg	200	NO
Manuel	Manuel	Hamburg	400	YES
null	**Michael**	**Munich**	**100**	*null*
null	**Frank**	**Frankfurt**	**100**	*null*
Simon	*null*	*null*	*null*	**NO**

FULL OUTER JOIN

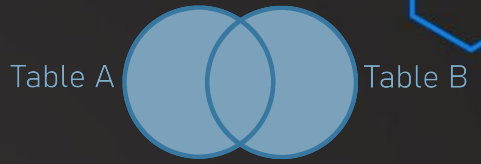


employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	**Munich**	**100**
Frank	**Frankfurt**	**100**

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	**NO**

bonus.employee	sales.employee	city	sales	bonus
null	Sandra	Frankfurt	500	YES
null	Sabine	Munich	300	YES
null	Peter	Hamburg	200	NO
null	Manuel	Hamburg	400	YES
null	**Michael**	**Munich**	100	*null*
null	**Frank**	**Frankfurt**	100	*null*
Simon	*null*	*null*	*null*	**NO**

FULL OUTER JOIN

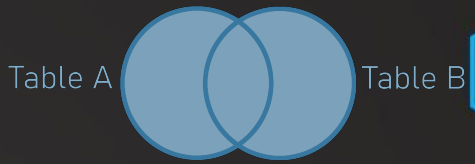


employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	**Munich**	**100**
Frank	**Frankfurt**	**100**

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	**NO**

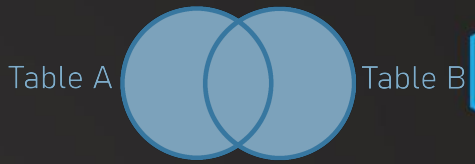
bonus.employee	sales.employee	city	sales	bonus
null	Sandra	Frankfurt	500	YES
null	Sabine	Munich	300	YES
null	Peter	Hamburg	200	NO
null	Manuel	Hamburg	400	YES
null	**Michael**	**Munich**	100	*null*
null	**Frank**	**Frankfurt**	100	*null*
Simon	*null*	*null*	*null*	**NO**

SYNTAX



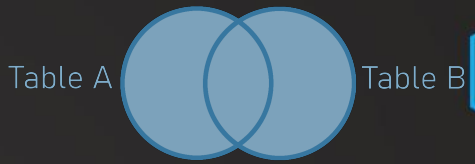

```
SELECT * FROM TableA
FULL OUTER JOIN TableB
ON TableA.employee = TableB.employee
```

SYNTAX



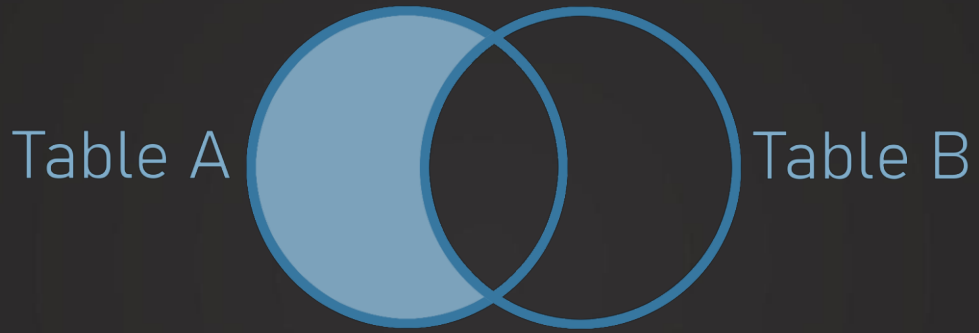

```
SELECT * FROM TableB
FULL OUTER JOIN TableA
ON TableA.employee = TableB.employee
```

SYNTAX




```
SELECT * FROM TableB
FULL OUTER JOIN TableA
ON TableA.employee = TableB.employee
WHERE condition
```

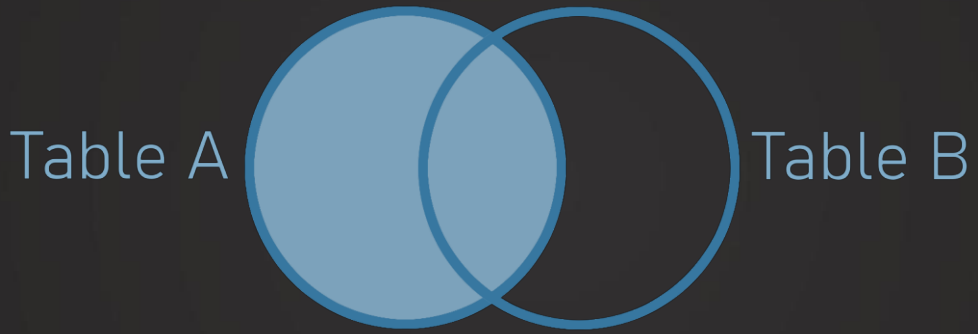
FULL OUTER JOIN



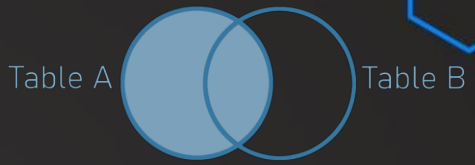

```
SELECT * FROM TableB
FULL OUTER JOIN TableA
ON TableA.employee = TableB.employee
WHERE TableB.anycolumn IS null
```

LEFT OUTER JOIN

LEFT OUTER JOIN



LEFT OUTER JOIN



sales table

employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	**Munich**	**100**
Frank	**Frankfurt**	**100**

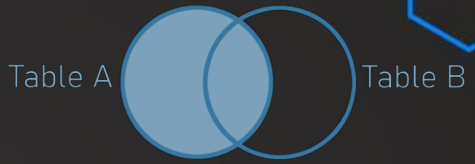
Do not appear in the bonus table

bonus table

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	**NO**

Do not appear in the sales table

LEFT OUTER JOIN



employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	**Munich**	**100**
Frank	**Frankfurt**	**100**

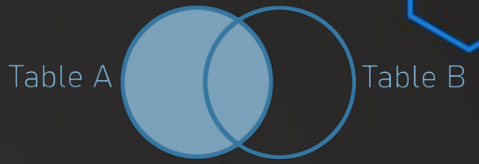
LEFT

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	**NO**

RIGHT

b.employee	s.employee	city	sales	bonus
Sandra	Sandra	Frankfurt	500	YES
Sabine	Sabine	Munich	300	YES
Peter	Peter	Hamburg	200	NO
Manuel	Manuel	Hamburg	400	YES
null	**Michael**	**Munich**	**100**	*null*
null	**Frank**	**Frankfurt**	**100**	*null*

LEFT OUTER JOIN



employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	**Munich**	**100**
Frank	**Frankfurt**	**100**

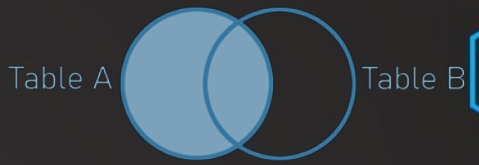
LEFT

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	**NO**

RIGHT

b.employee	s.employee	city	sales	bonus
Sandra	Sandra	Frankfurt	500	YES
Sabine	Sabine	Munich	300	YES
Peter	Peter	Hamburg	200	NO
Manuel	Manuel	Hamburg	400	YES
null	**Michael**	**Munich**	**100**	*null*
null	**Frank**	**Frankfurt**	**100**	*null*
~~**Simon**~~	~~*null*~~	~~*null*~~	~~*null*~~	~~**NO**~~

SYNTAX




```
SELECT * FROM TableA
LEFT OUTER JOIN TableB
ON TableA.employee = TableB.employee
```

Challenge

The flight company is trying to find out what their most popular seats are.
Try to find out which seat has been chosen most frequently.
Make sure all seats are included even if they have never been booked.

Are there seats that have never been booked?

Result

	seat_no character varying (4)	count bigint
1	1A	53559
2	4A	53181
3	2A	53145

Challenge

Try to find out which line (A, B, …, H) has been chosen most frequently.

	seat_no character varying (4)	count bigint
1	1A	53559
2	4A	53181
3	2A	53145

Result

1	A	751618
2	D	652188
3	C	596921

Challenge

You want to create a tier list in the following way:

1. Rating is 'PG' or 'PG-13' or length is more then 210 min:
   'Great rating or long (tier 1)
2. Description contains 'Drama' and length is more than 90min:
   'Long drama (tier 2)'
3. Description contains 'Drama' and length is not more than 90min:
   'Shcity drama (tier 3)'
4. Rental_rate less than $1:
   'Very cheap (tier 4)'

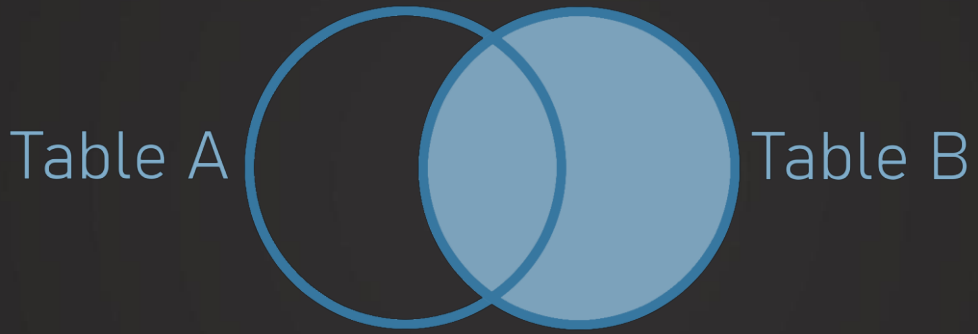
If one movie can be in multiple categories it gets the higher tier assigned.
How can you filter to only those movies that appear in one of these 4 tiers?

Result

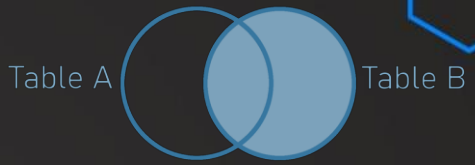
	title text	case text
1	ACADEMY DINOSAUR	Great rating or very long (tier 1)
2	AGENT TRUMAN	Great rating or very long (tier 1)
3	AIRPLANE SIERRA	Great rating or very long (tier 1)
4	ALABAMA DEVIL	Great rating or very long (tier 1)
5	ALAMO VIDEOTAPE	Very cheap (tier 4)

RIGHT OUTER JOIN

RIGHT OUTER JOIN



FULL OUTER JOIN



sales table

employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	**Munich**	**100**
Frank	**Frankfurt**	**100**

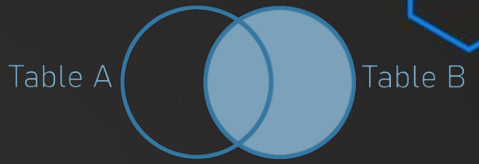
Do not appear in the bonus table

bonus table

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	**NO**

Do not appear in the sales table

FULL OUTER JOIN



employee	city	sales
Sandra	Frankfurt	500
Sabine	Munich	300
Peter	Hamburg	200
Manuel	Hamburg	400
Michael	**Munich**	**100**
Frank	**Frankfurt**	**100**

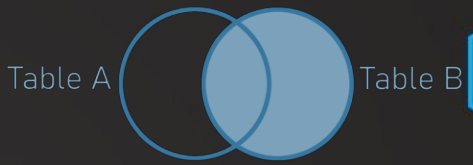
LEFT

employee	bonus
Sandra	YES
Sabine	YES
Peter	NO
Manuel	YES
Simon	**NO**

RIGHT

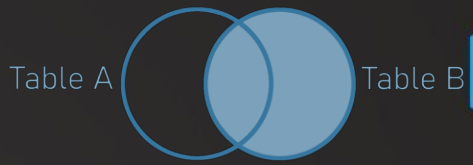
b.employee	s.employee	city	sales	bonus
Sandra	Sandra	Frankfurt	500	YES
Sabine	Sabine	Munich	300	YES
Peter	Peter	Hamburg	200	NO
Manuel	Manuel	Hamburg	400	YES
Simon	*null*	*null*	*null*	**NO**
~~*null*~~	~~**Michael**~~	~~**Munich**~~	~~**100**~~	~~*null*~~
~~*null*~~	~~**Frank**~~	~~**Frankfurt**~~	~~**100**~~	~~*null*~~

SYNTAX




```
SELECT * FROM TableA
RIGHT OUTER JOIN TableB
ON TableA.employee = TableB.employee
```

SYNTAX




```
SELECT * FROM TableB
LEFT OUTER JOIN TableA
ON TableA.employee = TableB.employee
```

Challenge

The company wants to run a phone call campaing on all customers in Texas (=district).

What are the customers (first_name, last_name, phone number and their district) from Texas?

Are there any (old) addresses that are not related to any customer?

Result

first_name text	last_name text	phone text	district text
JENNIFER	DAVIS	860452626434	Texas
KIM	CRUZ	909029256431	Texas
RICHARD	MCCRARY	262088367001	Texas
BRYAN	HARDISON	775235029633	Texas
IAN	STILL	239357986667	Texas

address_id integer	address text	a t
1	47 MySakila Drive	[
2	28 MySQL Boulevard	[
3	23 Workhaven Lane	[
4	1411 Lillydale Drive	[

Multiple join conditions

Multiple join conditions

sales table

first_name	last_name	city	sales
Sandra	Jones	Frankfurt	500
Sandra	Williams	Munich	300
Peter	Davis	Hamburg	200
Manuel	White	Hamburg	400
Michael	Anderson	Munich	100
Frank	Wilson	Frankfurt	100

bonus table

first_name	last_name	bonus
Sandra	Jones	YES
Sandra	Williams	YES
Peter	Davis	NO
Manuel	White	YES
Simon	Taylor	NO

SYNTAX

```
SELECT * FROM TableA a
INNER JOIN TableB b
ON a.first_name = b.first_name
AND a.last_name = b.last_name
```

Expert tip

```
SELECT * FROM TableA a
INNER JOIN TableB b
ON a.first_name = b.first_name
AND a.last_name = 'Jones'
```

Expert tip

```
SELECT * FROM TableA a
INNER JOIN TableB b
ON a.first_name = b.first_name
AND a.last_name = 'Jones'
```

More performance-efficient!

```
SELECT * FROM TableA a
INNER JOIN TableB b
ON a.first_name = b.first_name
WHERE a.last_name = 'Jones'
```

Challenge

The company wants to run a phone call campaing on all customers in Texas (=district).

What are the customers (first_name, last_name, phone number and their district) from Texas?

Are there any (old) addresses that are not related to any customer?

Result

first_name text	last_name text	phone text	district text
JENNIFER	DAVIS	860452626434	Texas
KIM	CRUZ	909029256431	Texas
RICHARD	MCCRARY	262088367001	Texas
BRYAN	HARDISON	775235029633	Texas
IAN	STILL	239357986667	Texas

address_id integer	address text	a t
1	47 MySakila Drive	[
2	28 MySQL Boulevard	[
3	23 Workhaven Lane	[
4	1411 Lillydale Drive	[

Two join tables

sales table

first_name	last_name	city	sales
Sandra	Jones	Frankfurt	500
Sandra	Williams	Munich	300
Peter	Davis	Hamburg	200
Manuel	White	Hamburg	400
Michael	Anderson	Munich	100
Frank	Wilson	Frankfurt	100

bonus table

first_name	last_name	bonus
Sandra	Jones	YES
Sandra	Williams	YES
Peter	Davis	NO
Manuel	White	YES
Simon	Taylor	NO

Joining multiple tables

sales table

employee	city_id	sales
Sandra	1	500
Sabine	2	300
Peter	3	200
Manuel	3	400
Michael	3	100
Frank	1	100

city table

city_id	city	country_id
1	Frankfurt	1
2	Munich	1
3	New York	2

country table

country_id	country
1	Germany
2	USA

Joining multiple tables

sales table

employee	city_id	sales
Sandra	1	500
Sabine	2	300
Peter	3	200
Manuel	3	400
Michael	3	100
Frank	1	100

city table

city_id	city	country_id
1	Frankfurt	1
2	Munich	1
3	New York	2

country table

country_id	country
1	Germany
2	USA

employee	country
Sandra	Germany
Sabine	Germany
Peter	USA
Manuel	USA
Michael	USA
Frank	Germany

Joining multiple tables

sales table

employee	city_id	sales
Sandra	1	500
Sabine	2	300
Peter	3	200
Manuel	3	400
Michael	3	100
Frank	1	100

city table

city_id	city	country_id
1	Frankfurt	1
2	Munich	1
3	New York	2

country table

country_id	country
1	Germany
2	USA

employee	country
Sandra	Germany
Sabine	Germany
Peter	USA
Manuel	USA
Michael	USA
Frank	Germany

Joining multiple tables

sales table

employee	city_id	sales
Sandra	1	500
Sabine	2	300
Peter	3	200
Manuel	3	400
Michael	3	100
Frank	1	100

city table

city_id	city	country_id
1	Frankfurt	1
2	Munich	1
3	New York	2

country table

country_id	country
1	Germany
2	USA

employee	country_id
Sandra	1
Sabine	1
Peter	2
Manuel	2
Michael	2
Frank	1

employee	country
Sandra	Germany
Sabine	Germany
Peter	USA
Manuel	USA
Michael	USA
Frank	Germany

SYNTAX

```
SELECT employee, ci.country_id FROM sales s
INNER JOIN city ci
ON s.city_id = ci.city_id
```

employee	country_id
Sandra	1
Sabine	1
Peter	2
Manuel	2
Michael	2
Frank	1

SYNTAX

```
SELECT employee, co.country FROM sales s
INNER JOIN city ci
ON s.city_id = ci.city_id
INNER JOIN country co
ON ci.country_id = co.country_id
```

employee	country
Sandra	Germany
Sabine	Germany
Peter	USA
Manuel	USA
Michael	USA
Frank	Germany

SYNTAX

```
SELECT employee, co.country FROM sales s
INNER JOIN country co
ON ci.country_id = co.country_id
INNER JOIN city ci
ON s.city_id = ci.city_id
```

INNER JOIN:

Table order doesn't matter!

employee	country
Sandra	Germany
Sabine	Germany
Peter	USA
Manuel	USA
Michael	USA
Frank	Germany

Joining multiple tables

sales table

employee	city_id	sales
Sandra	1	500
Sabine	2	300
Peter	3	200
Manuel	3	400
Michael	3	100
Frank	4	100

city table

city_id	city	country_id
1	Frankfurt	1
2	Munich	1
3	New York	2

country table

country_id	country
1	Germany
2	USA

employee	country
Sandra	Germany
Sabine	Germany
Peter	USA
Manuel	USA
Michael	USA
Frank	*null*

SYNTAX

```
SELECT employee, co.country FROM sales s
LEFT JOIN city ci
ON s.city_id = ci.city_id
LEFT JOIN country co
ON ci.country_id = co.country_id
```

employee	country
Sandra	Germany
Sabine	Germany
Peter	USA
Manuel	USA
Michael	USA
Frank	*null*

SYNTAX

```
SELECT employee, co.country FROM city ci
LEFT JOIN sales s
ON s.city_id = ci.city_id
LEFT JOIN country co
ON ci.country_id = co.country_id
```

employee	country
Sandra	Germany
Sabine	Germany
Peter	USA
Manuel	USA
Michael	USA
Frank	*null*

Challenge

The company wants customize their campaigns to customers depending on the country they are from.

Which customers are from Brazil?

Write a query to get first_name, last_name, email and the country from all customers from Brazil.

Result

first_name text	last_name text	email text	country text
CLAYTON	BARBEE	CLAYTON.BARBEE@sakilacustomer.org	Brazil
JOSEPH	JOY	JOSEPH.JOY@sakilacustomer.org	Brazil
TAMARA	NGUYEN	TAMARA.NGUYEN@sakilacustomer.org	Brazil