PostgreSQL Associations

How to implement relationships

Relationships

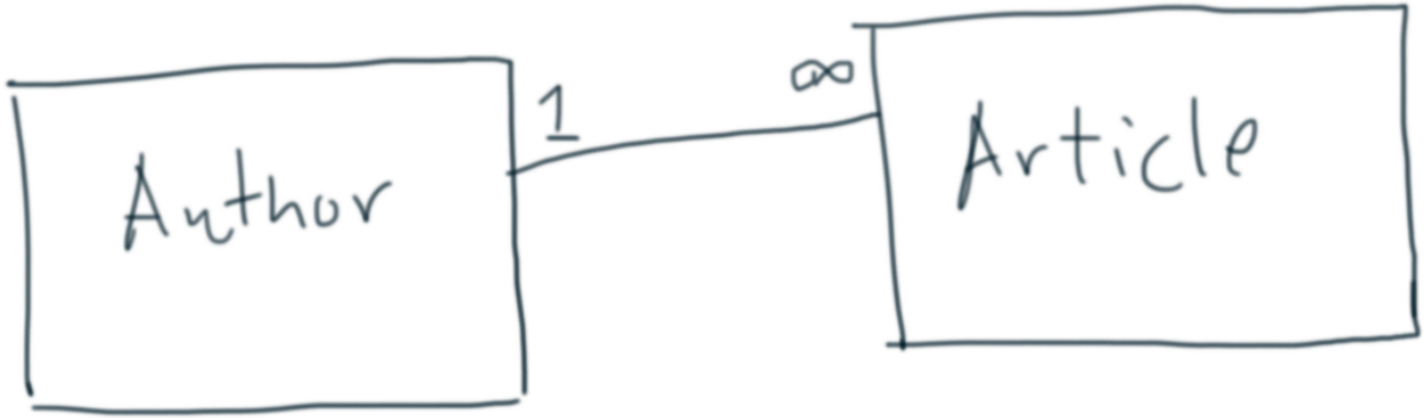
Modeling Relationships

- authors and articles
- owners and cars
- owners and pets
- parent and children
- albums and artists
- projects and contributors
- classes and objects

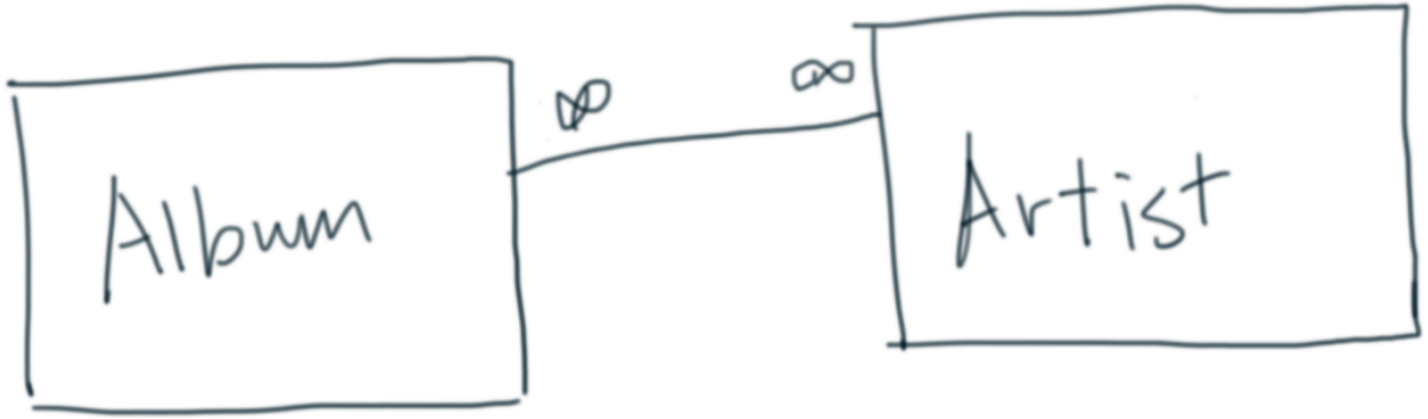
Types of Relationships

- One-to-Many
- Many-to-Many

One-to-Many



Many-to-Many

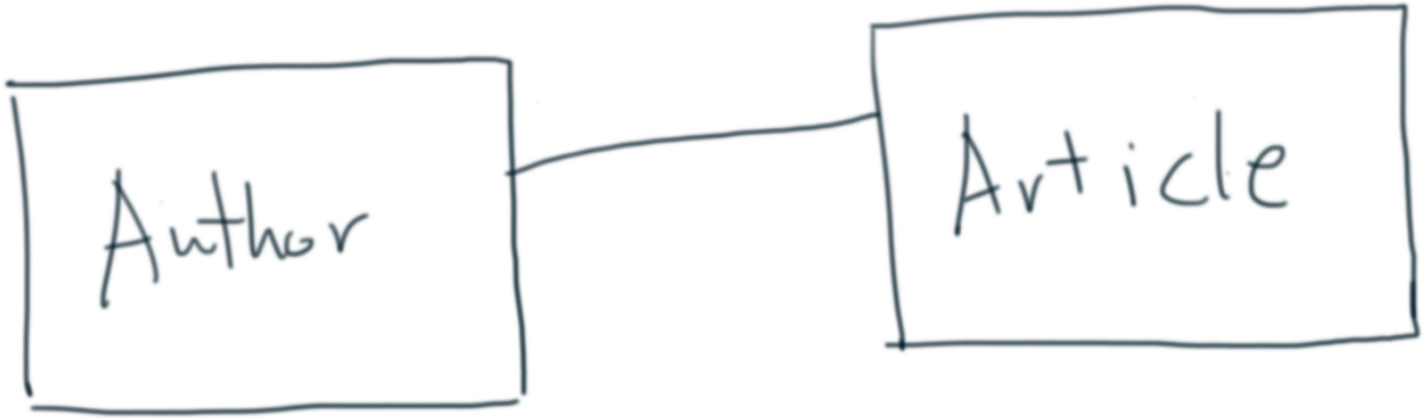


How to determine the relationship?

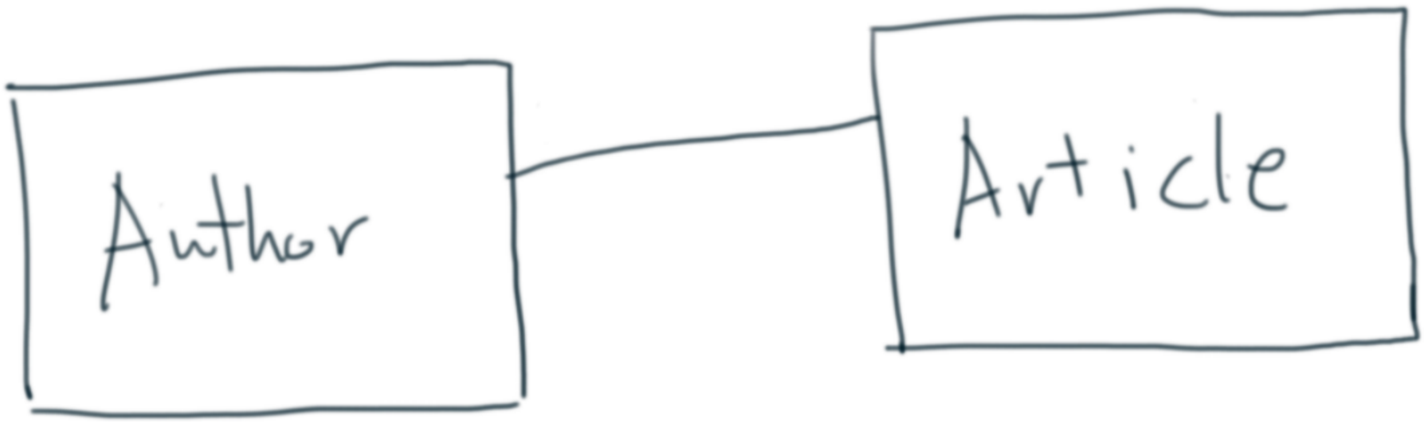
Ask these questions

- Can 1 X have multiple Y’s?
- Can 1 Y have multiple X’s?

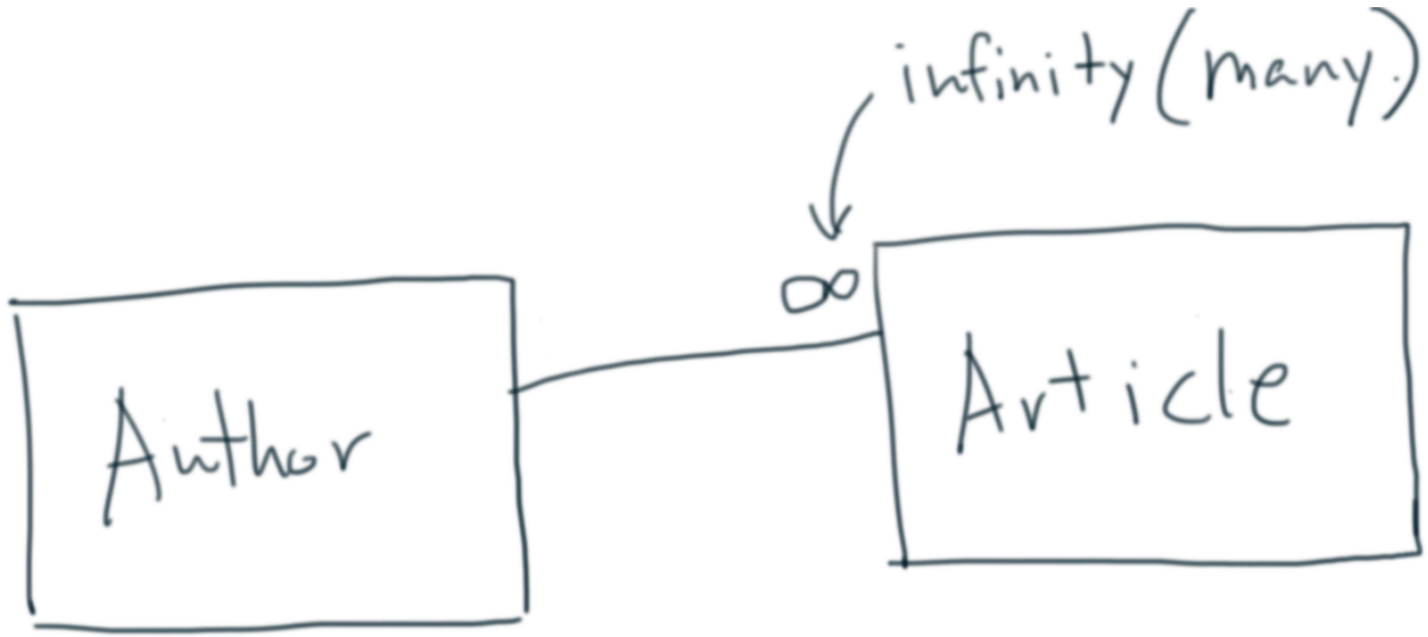
Example


Author
Article

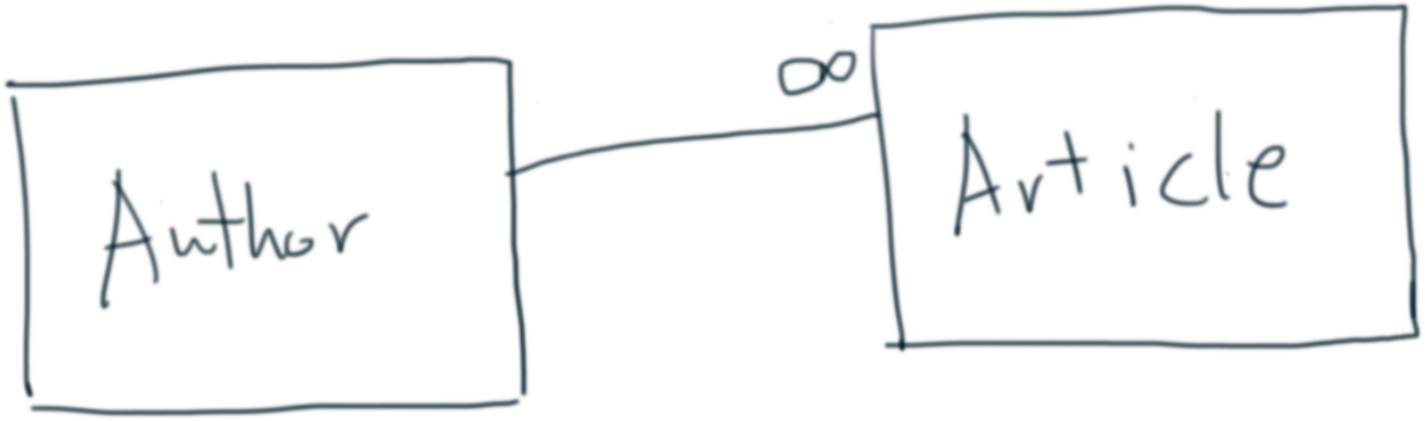
Can 1 Author have multiple articles?



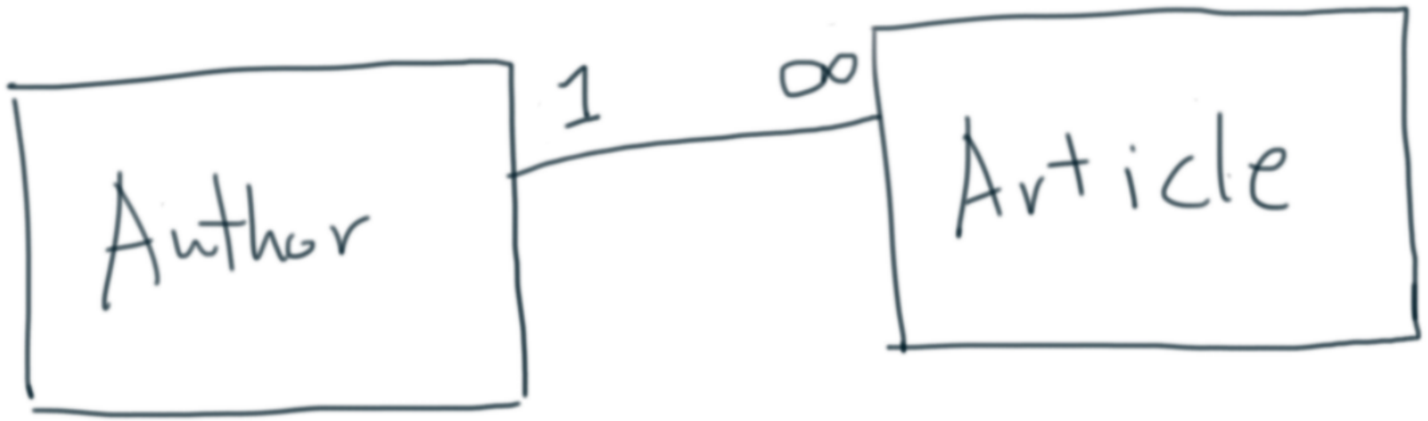
Can 1 Author have multiple articles?
Yes.



Can 1 article have multiple authors?

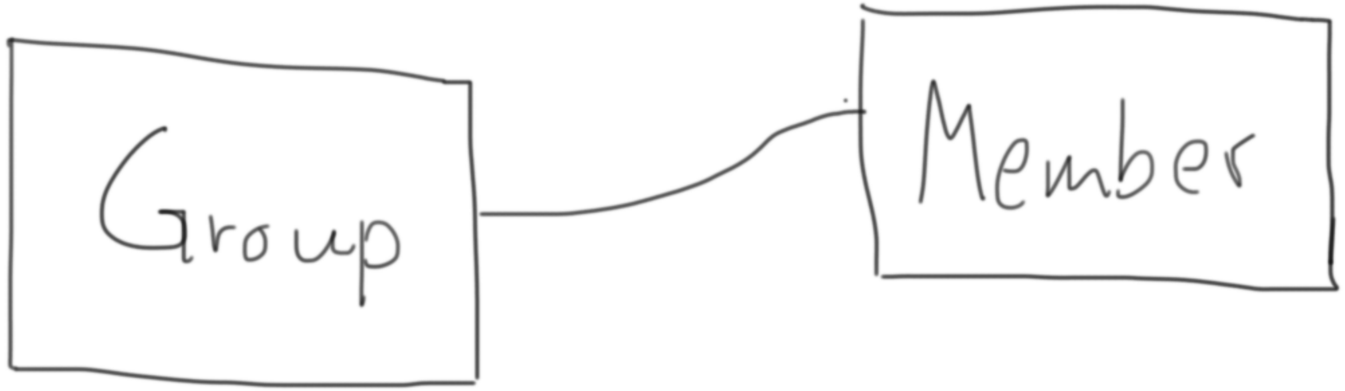


Can 1 article have multiple authors?
No.

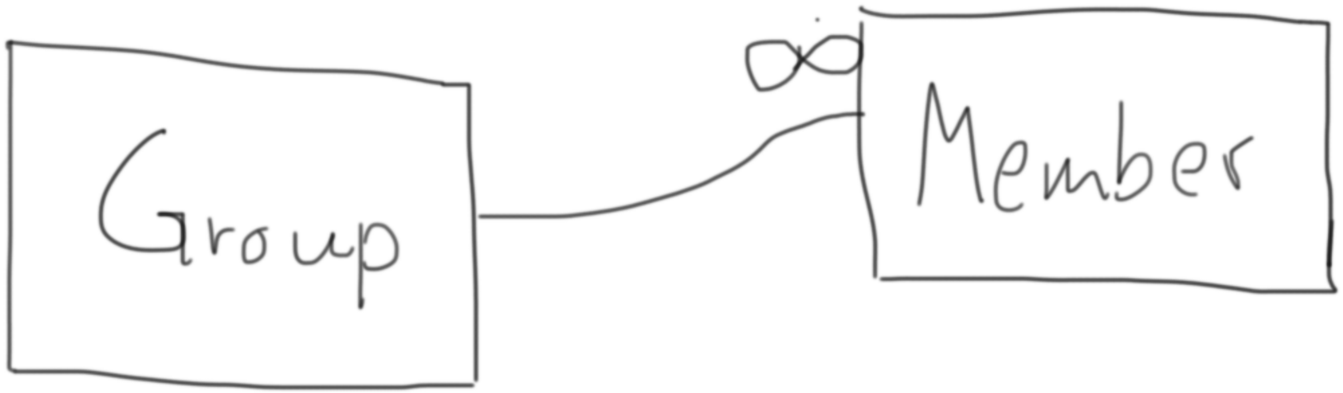


Example

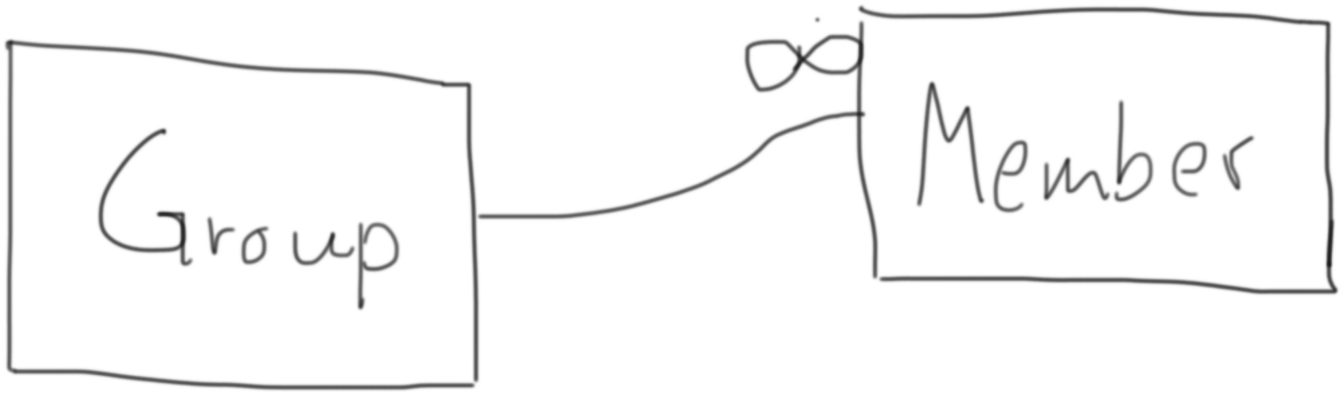
Can 1 group have many members?



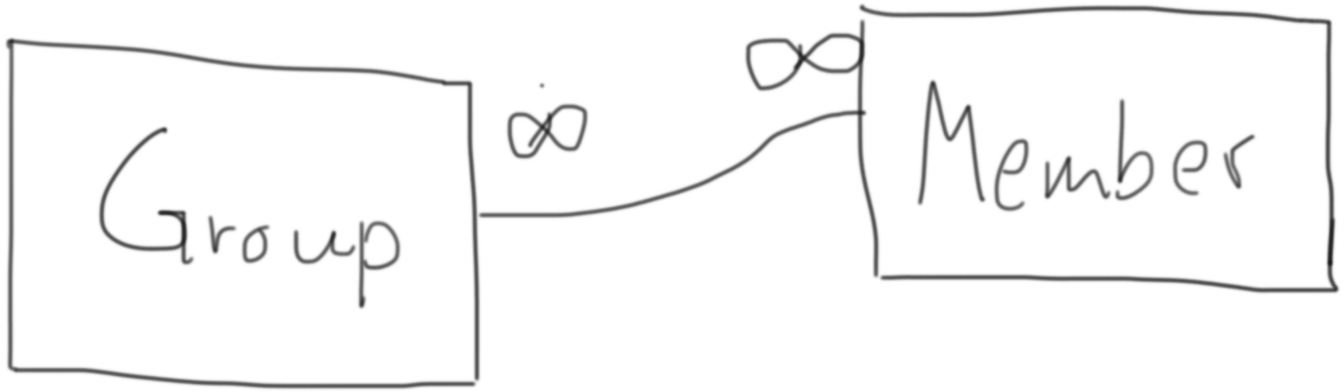
Can 1 group have many members?
Yes.



Can 1 member belong to many groups?



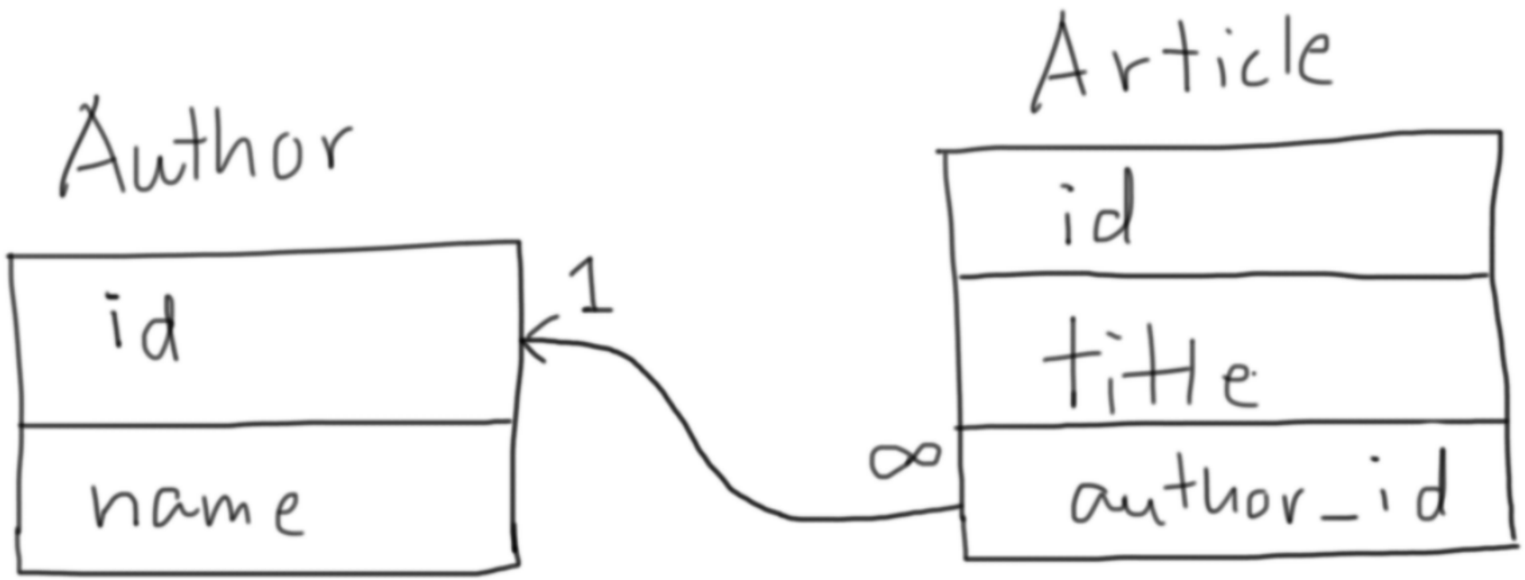
Can 1 member belong to many groups?
Yes.



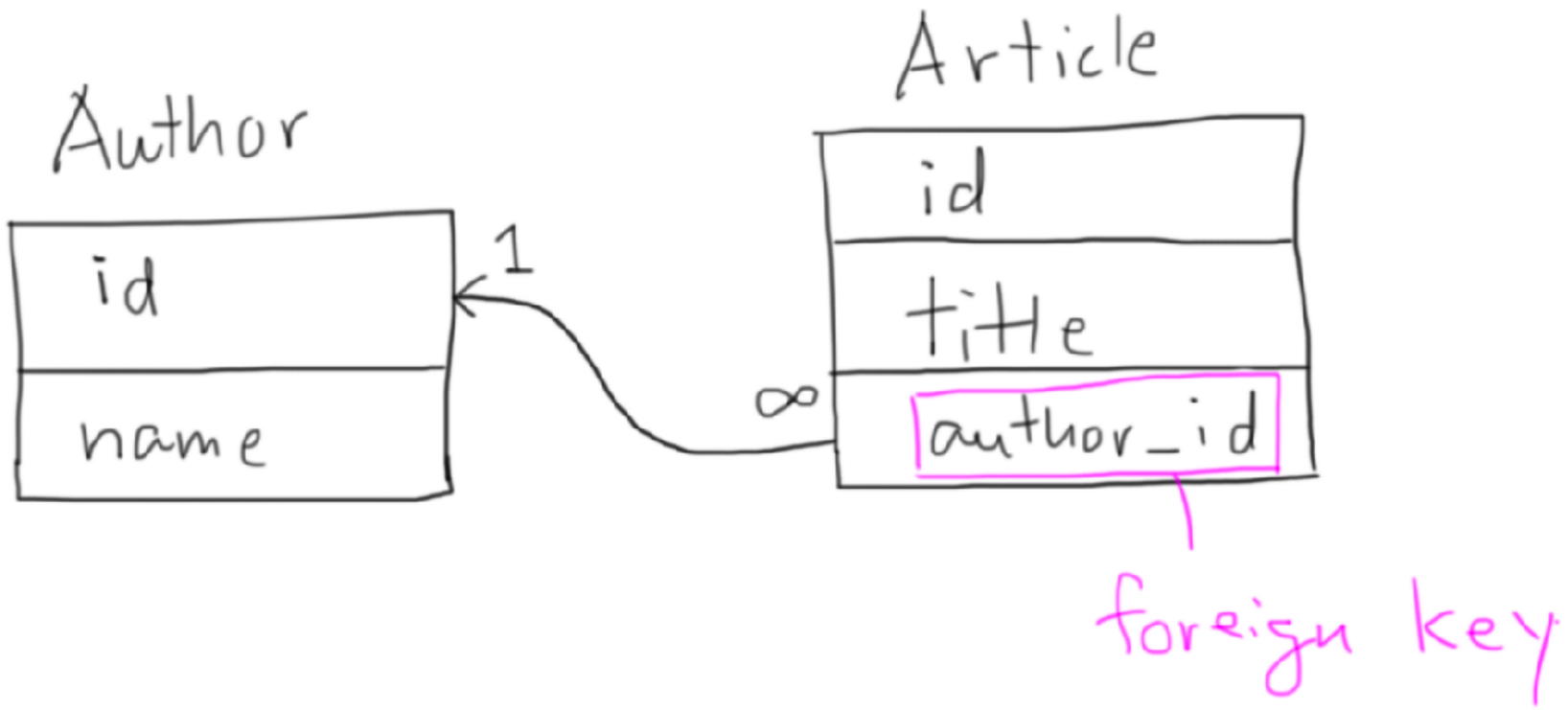
More Examples

- people and legs
- people and homes
- company and employees
- team and players
- Facebook groups and members

Representing One-to-Many



Foreign Key



Many side has a *foreign key* that references the primary key of the one side

Foreign Key

```
CREATE TABLE author (
  id serial PRIMARY KEY,
  name varchar
);

CREATE TABLE article (
  id serial PRIMARY KEY,
  title varchar,
  author_id integer REFERENCES author (id)
);
```

Many side has a *foreign key* that references the primary key of the one side

Foreign Key

Article

id	title	author_id
1	Callbacks. What are they good for?	2
2	Escape from Callback Hell with Promises.	2
3	Where's the Closure?	1
4	What are Higher Order Functions?	NULL

Author

id	name
1	Igor
2	Sofia
3	Alvaro

Foreign key

Foreign Key

Article

id	title	author_id
1	Callbacks. What are they good for?	2
2	Escape from Callback Hell with Promises.	2
3	Where's the Closure?	1
4	What are Higher Order Functions?	NULL

Author

id	name
1	Igor
2	Sofia
3	Alvaro

Foreign key

Joins

Join (Inner)

Associate articles with their authors:

id	title	author_id	id	name
1	Callbacks. What are they good for?	2	2	Sofia
2	Escape from Callback Hell with Promises.	2	2	Sofia
3	Where's the Closure?	1	1	Igor

Inner Join

```
select * from article, author where article.author_id = author.id;
```

id	title	author_id	id	name
1	Callbacks. What are they good for?	2	2	Sofia
2	Escape from Callback Hell with Promises.	2	2	Sofia
3	Where's the Closure?	1	1	Igor

Inner Join

```
select * from article, author where article.author_id = author.id;
```

tables to join

join conditional

The join conditional is essential.
It equates the foreign key in one table to the primary key in the other table.

Inner Join

```
select * from article, author where article.author_id = author.id;
```

tables to join

join conditional

What happens if you left out the join conditional?

Inner Join

```
select * from article, author where article.author_id = author.id;
```

tables to join

join conditional

What happens if you left out the join conditional?

Cartesian Join

```
select * from article, author;
```

The cartesian join pairs every article with every author, regardless if they are related.

id	title	author_id	id	name
1	Callbacks. What are they good for?	2	1	Igor
2	Escape from Callback Hell with Promises.	2	1	Igor
3	Where's the Closure?	1	1	Igor
4	What are Higher Order Functions?	NULL	1	Igor
1	Callbacks. What are they good for?	2	2	Sofia
2	Escape from Callback Hell with Promises.	2	2	Sofia
3	Where's the Closure?	1	2	Sofia
4	What are Higher Order Functions?	NULL	2	Sofia
1	Callbacks. What are they good for?	2	3	Alvaro
2	Escape from Callback Hell with Promises.	2	3	Alvaro
3	Where's the Closure?	1	3	Alvaro
4	What are Higher Order Functions?	NULL	3	Alvaro

Cartesian Join to Inner Join

```
select * from article, author;
```

The join conditional filters it down to only the results that make sense

```
where article.author_id = author.id;
```

id	title	author_id	id	name
1	Callbacks. What are they good for?	2	1	Igor
2	Escape from Callback Hell with Promises.	2	1	Igor
3	Where's the Closure?	1	1	Igor
4	What are Higher Order Functions?	NULL	1	Igor
1	Callbacks. What are they good for?	2	2	Sofia
2	Escape from Callback Hell with Promises.	2	2	Sofia
3	Where's the Closure?	1	2	Sofia
4	What are Higher Order Functions?	NULL	2	Sofia
1	Callbacks. What are they good for?	2	3	Alvaro
2	Escape from Callback Hell with Promises.	2	3	Alvaro
3	Where's the Closure?	1	3	Alvaro
4	What are Higher Order Functions?	NULL	3	Alvaro

Alternate Inner Join Syntax

```
select * from article inner join author on article.author_id = author.id;
```

id	title	author_id	id	name
1	Callbacks. What are they good for?	2	2	Sofia
2	Escape from Callback Hell with Promises.	2	2	Sofia
3	Where's the Closure?	1	1	Igor

Alternate Inner Join Syntax

explicitly joins
this to this

```
select * from article inner join author on article.author_id = author.id;
```

join conditional

Outer Join

```sql
select
    *
from
    article
left outer join
    author on article.author_id = author.id;
```

id	title	author_id	id	name
1	Callbacks. What are they good for?	2	2	Sofia
2	Escape from Callback Hell with Promises.	2	2	Sofia
3	Where's the Closure?	1	1	Igor
4	What are Higher Order Functions?	NULL	NULL	NULL

Outer Join

```
select
    *
from
    article
left outer join
    author on article.author_id = author.id;
```

left table

right table

Like inner join, but ensures all rows in the left table are represented

Outer Join

id	title	author_id	id	name
1	Callbacks. What are they good for?	2	2	Sofia
2	Escape from Callback Hell with Promises.	2	2	Sofia
3	Where's the Closure?	1	1	Igor
4	What are Higher Order Functions?	NULL	NULL	NULL

NULL author_id,

no matching author

Right Outer Join

```sql
select
    *
from
    article
right outer join
    author on article.author_id = author.id;
```

id	title	author_id	id	name
1	Callbacks. What are they good for?	2	2	Sofia
2	Escape from Callback Hell with Promises.	2	2	Sofia
3	Where's the Closure?	1	1	Igor
NULL	NULL	NULL	3	Alvaro

Full Outer Join

```sql
select
    *
from
    article
full outer join
    author on article.author_id = author.id;
```

id	title	author_id	id	name
1	Callbacks. What are they good for?	2	2	Sofia
2	Escape from Callback Hell with Promises.	2	2	Sofia
3	Where's the Closure?	1	1	Igor
4	What are Higher Order Functions?	NULL	NULL	NULL
NULL	NULL	NULL	3	Alvaro

How many articles has each author written?

```
select
    author.id,
    author.name,
    count(article.id)
from
    article
inner join
    author on article.author_id = author.id
group by
    author.id;
```

How many articles has each author written?

```
select
    author.id,
    author.name,
    count(article.id)
from
    article
inner join
    author on article.author_id = author.id
group by
    author.id;
```

selecting is okay because grouping by primary key

How many articles has each author written?

```sql
select
    author.id,
    author.name,
    count(*)
from
    article
inner join
    author on article.author_id = author.id
group by
    author.id;
```

id	name	count
2	Sofia	2
1	Igor	1

Alvaro is not included

How many articles has each author written?

```sql
select
    author.id,
    author.name,
    count(article.id)
from
    article
right outer join
    author on article.author_id = author.id
group by
    author.id;
```

id	name	count
2	Sofia	2
1	Igor	1
3	Alvaro	0

How many articles has each author written?

Before aggregation:

id	title	author_id	id	name
1	Callbacks. What are they good for?	2	2	Sofia
2	Escape from Callback Hell with Promises.	2	2	Sofia
3	Where's the Closure?	1	1	Igor
NULL	NULL	NULL	3	Alvaro

After aggregation:

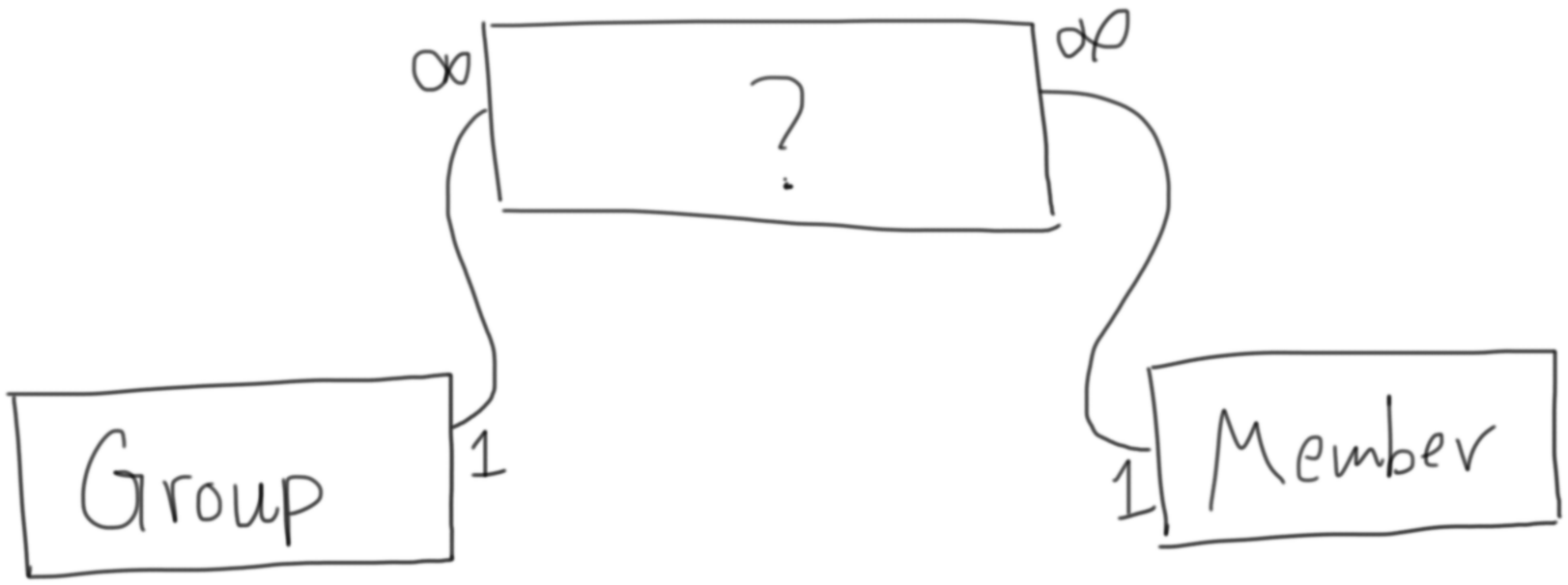
id	name	count
2	Sofia	2
1	Igor	1
3	Alvaro	0

Many-to-Many

Representing Many-to-Many

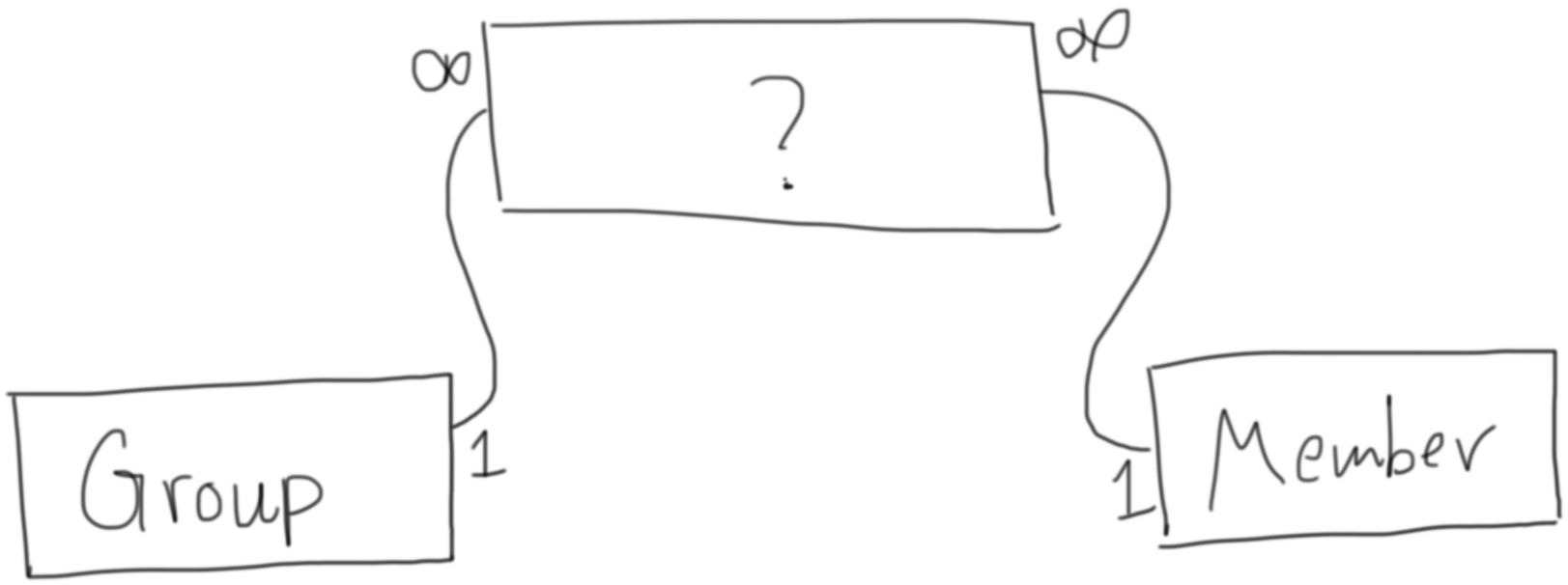
How to represent a many-to-many relationship in SQL?

Representing Many-to-Many



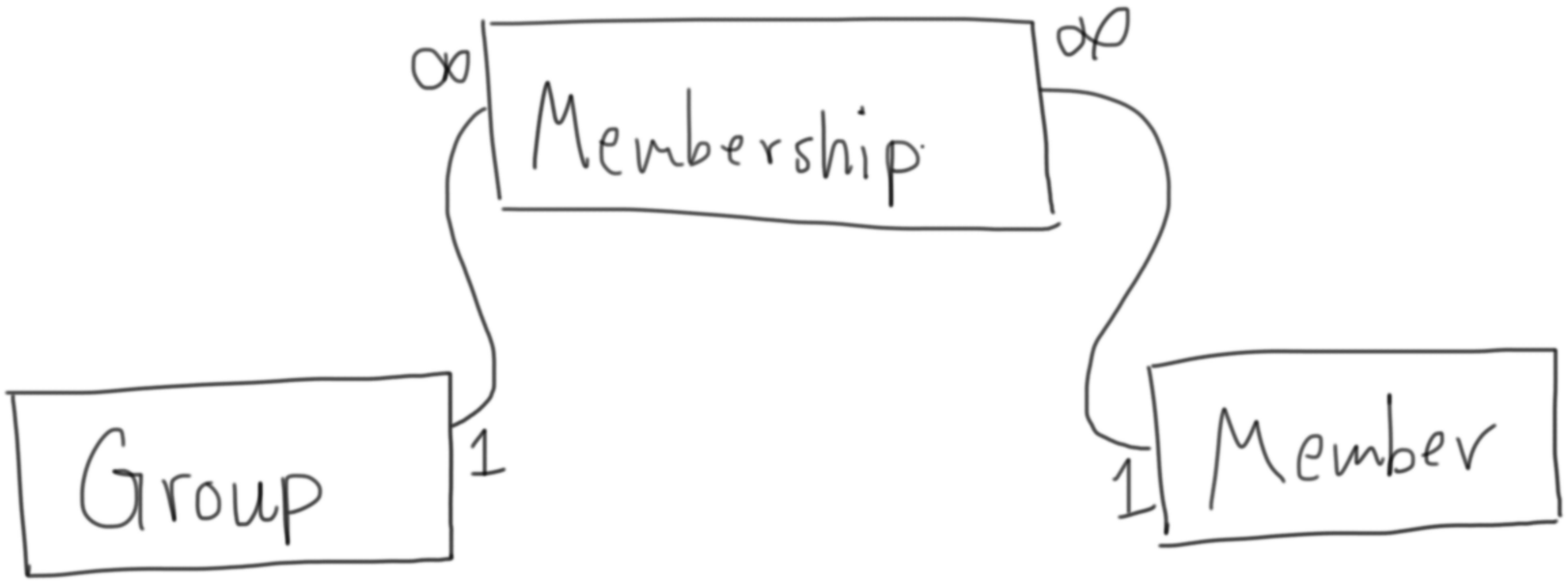
Add an ***associative table*** in the middle,
and it will link to each side as a many-to-one association.

Representing Many-to-Many



What to call the associative table? Group-Member? No!

Representing Many-to-Many



Use a meaningful name for your associative table.

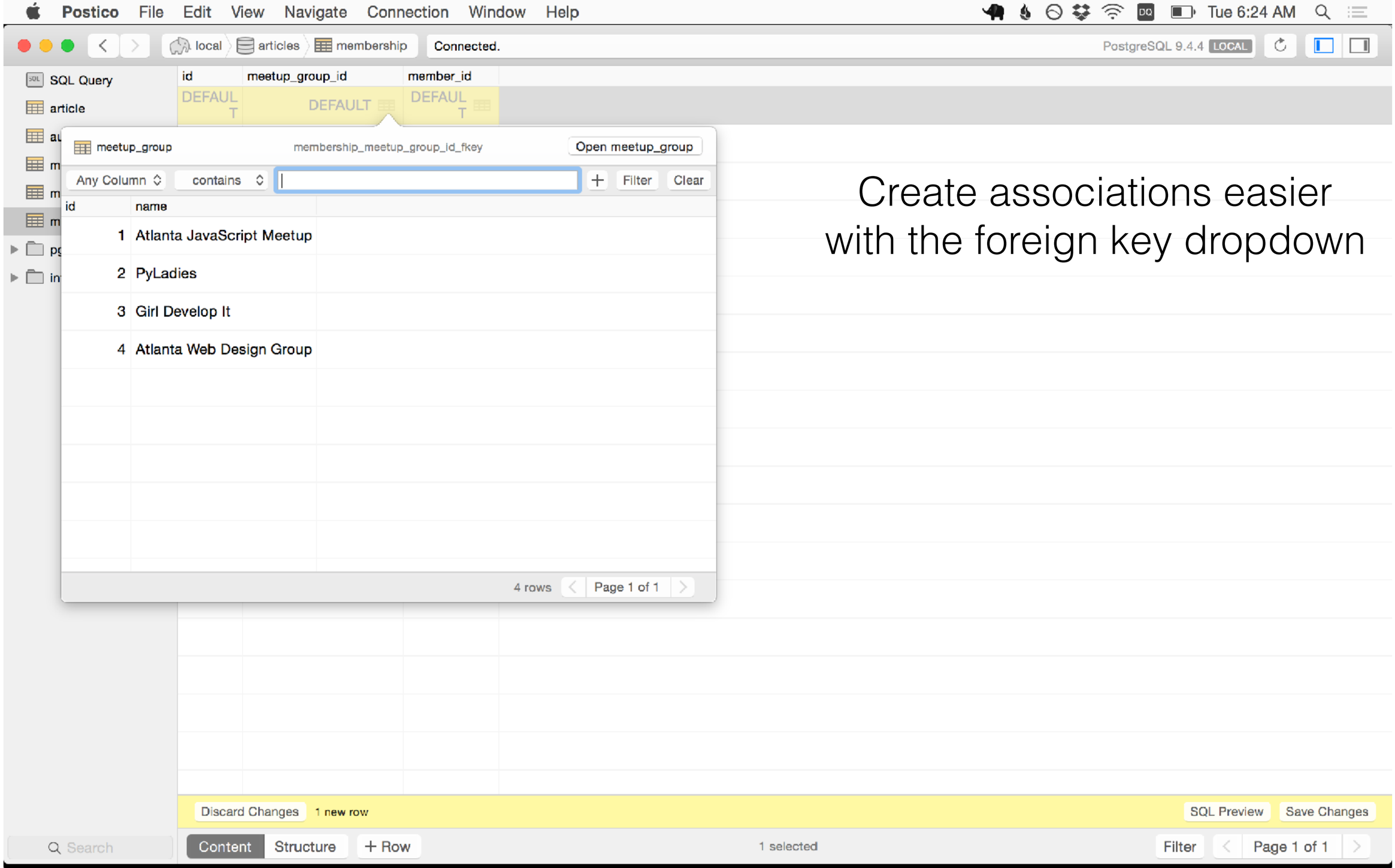
Many-to-Many

Group

id	name
1	Atlanta JavaScript Meetup
2	PyLadies
3	Girl Develop It
4	Atlanta Web Design Group

Member

id	name
1	Ian
2	Carl
3	Andreea
4	Debra
5	Julie
6	James



Create associations easier with the foreign key dropdown

Membership

id	meetup_group_id	member_id
1	1	1
2	1	2
3	1	5
4	1	6
5	2	5
6	2	3
7	3	4
8	3	5

Group

id	name
1	Atlanta JavaScript Meetup
2	PyLadies
3	Girl Develop It
4	Atlanta Web Design Group

Member

id	name
1	Ian
2	Carl
3	Andreea
4	Debra
5	Julie
6	James

Membership

id	meetup_group_id	member_id
1	1	1
2	1	2
3	1	5
4	1	6
5	2	5
6	2	3
7	3	4
8	3	5

Group

id	name
1	Atlanta JavaScript Meetup
2	PyLadies
3	Girl Develop It
4	Atlanta Web Design Group

Member

id	name
1	Ian
2	Carl
3	Andreea
4	Debra
5	Julie
6	James