PostgreSQL Exercises

Previous Exercise

Home / Joins / Self

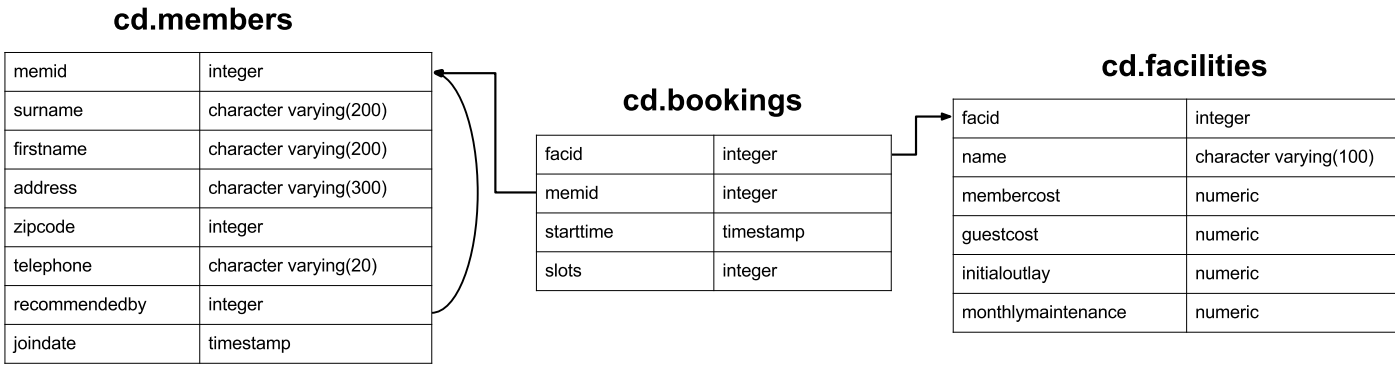
Next Exerci

Produce a list of all members who have recommended another member

Question

How can you output a list of all members who have recommended another member? Ensure that there are no duplicat in the list, and that results are ordered by (surname, firstname).

Schema reminder ▲



Expected Results

firstname	surname
Florence	Bader
Timothy	Baker
Gerald	Butters
Jemima	Farrell
Matthew	Genting
David	Jones
Janice	Joplette
Millicent	Purview
Tim	Rownam
Darren	Smith

Your Answer

Hint Help Save Run Query

```
select distinct m1.firstname, m1.surname
from cd.members m1
join cd.members m2 on m1.memid = m2.recommendedby
order by surname, firstname;
```

firstname	surname
Florence	Bader
Timothy	Baker
Gerald	Butters
Jemima	Farrell
Matthew	Genting
David	Jones

Answers and Discussion Hide

```
select distinct recs.firstname as firstname, recs.surname as surname
        from
                cd.members mems
                inner join cd.members recs
                        on recs.memid = mems.recommendedby
order by surname, firstname;
```

Here's a concept that some people find confusing: you can join a table to itself! This is really useful if you have columns that reference data in the same table, like we do with recommendedby in cd.members.

If you're having trouble visualising this, remember that this works just the same as any other inner join. Our join takes each row in members that has a recommendedby value, and looks in members again for the row which has a matching member id. It then generates an output row combining the two members entries. This looks like the diagram below:

SELECT

mems.memid	mems.surname	mems.firstname	mems.addre	mems.zipcod	mems.teleph	mems.recommendedby	mems.joinda	recs.memid	recs.surname	recs.firstname	recs.address	recs.zipcode	recs.telepho	recs.recomm	recs.jo
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
7	Dare	Nancy	6 Hunting Lo	10383	(833) 776-40	4	2012-07-25 0	4	Joplette	Janice	20 Crossing Roac	234	(833) 942-47	1	2012-0
11	Jones	David	976 Gnats Cl	33862	(844) 536-80	4	2012-08-06 1	4	Joplette	Janice	20 Crossing Roac	234	(833) 942-47	1	2012-0
35	Hunt	John	5 Bullington	54333	(899) 720-69	30	2012-09-19 1	30	Purview	Millicent	641 Drudgery Cl	34232	(855) 941-97	2	2012-0
8	Boothe	Tim	3 Bloomsbur	234	(811) 433-25	3	2012-07-25 1	3	Rownham	Tim	23 Highway Way	23423	(844) 693-0723		2012-0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

Note that while we might have two 'surname' columns in the output set, they can be distinguished by their table aliase Once we've selected the columns that we want, we simply use `DISTINCT` to ensure that there are no duplicates.

Previous Exercise

Home / Joins / Self

Next Exerci