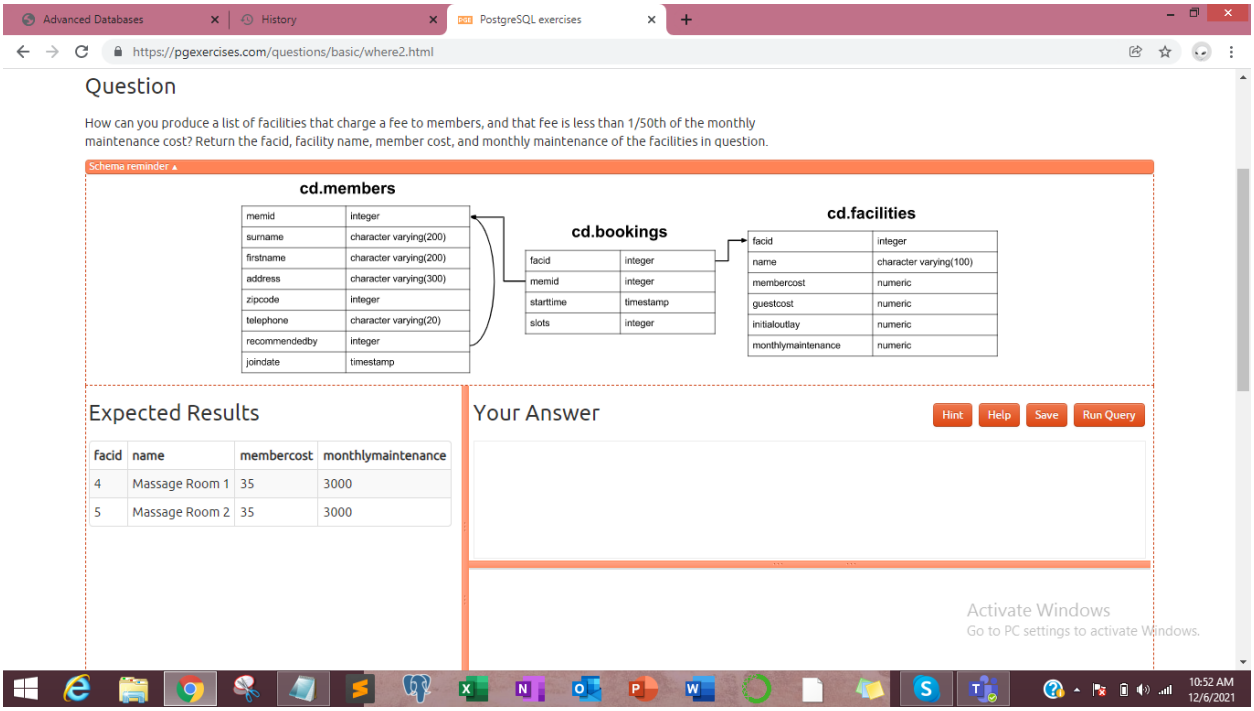
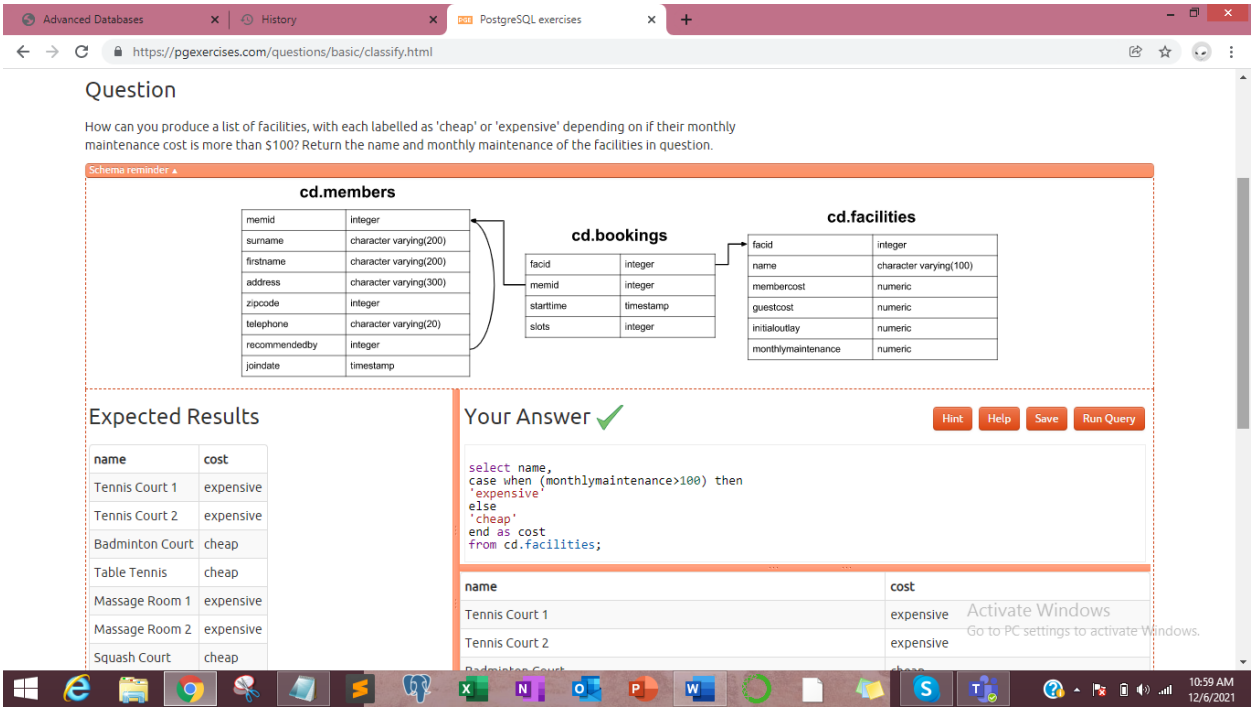
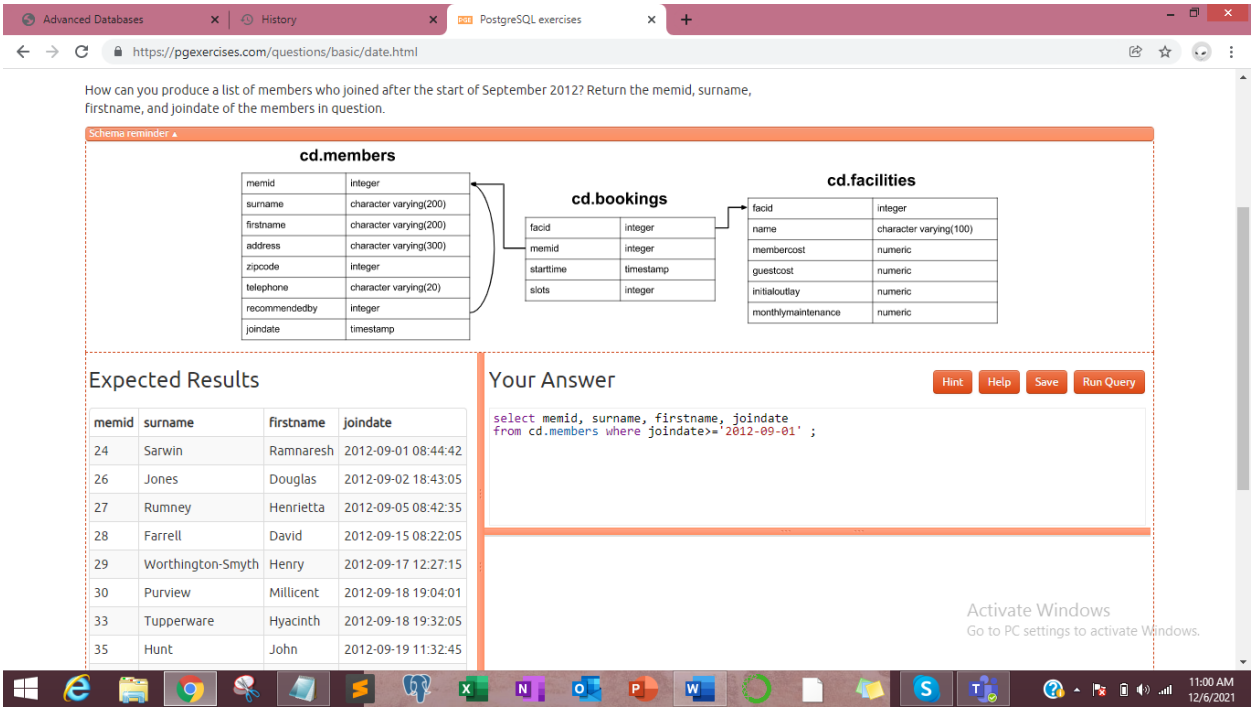
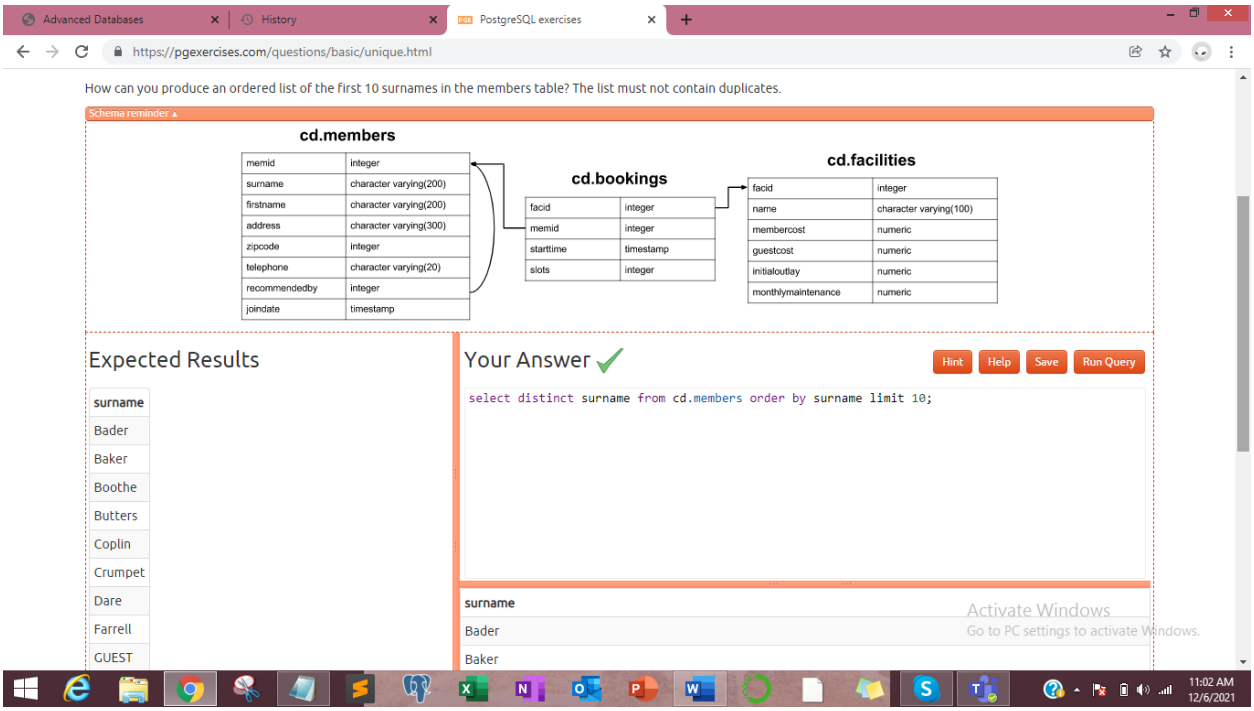
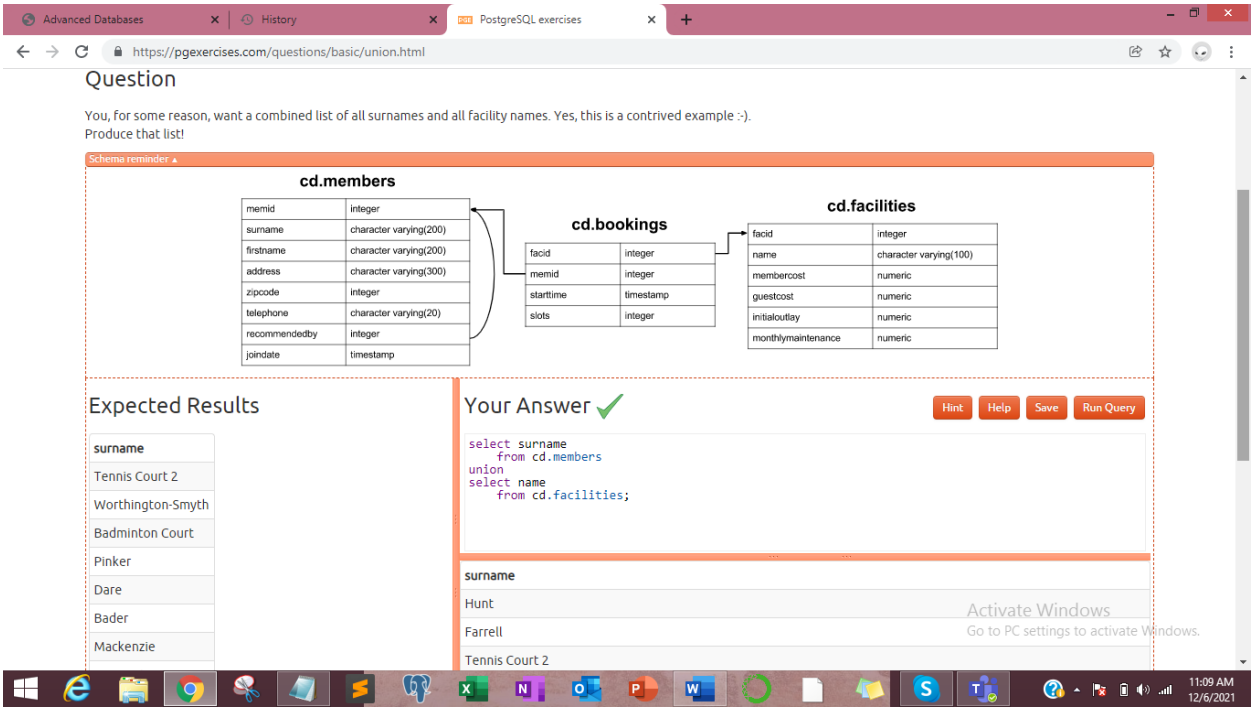


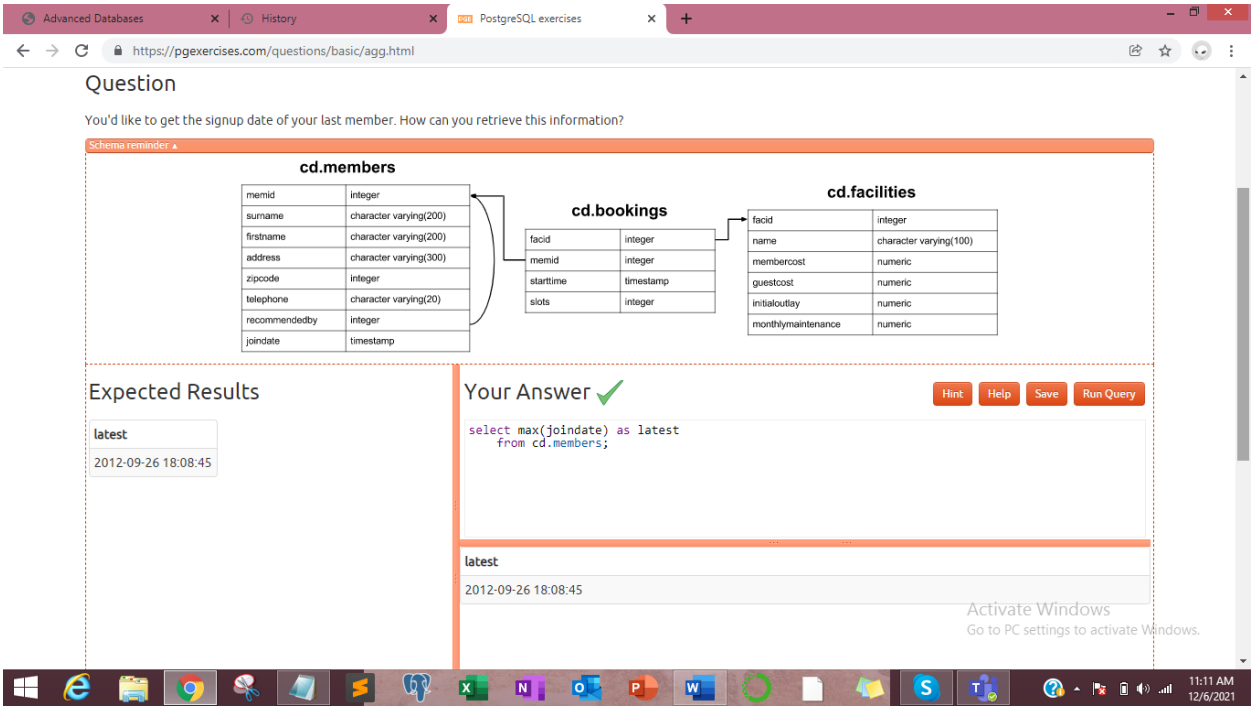

```
select facid, name, membercost, monthlymaintenance from cd.facilities where
membercost > 0 and (membercost < monthlymaintenance/50.0);
```




Advanced Databases
History
PostgreSQL exercises
https://pgexercises.com/questions/basic/date.html
How can you produce a list of members who joined after the start of September 2012? Return the memid, surname, firstname, and joindate of the members in question.
Schema reminder
cd.members
memid integer
surname character varying(200)
firstname character varying(200)
address character varying(300)
zipcode integer
telephone character varying(20)
recommendedby integer
joindate timestamp
cd.bookings
facid integer
memid integer
starttime timestamp
slots integer
cd.facilities
facid integer
name character varying(100)
membercost numeric
guestcost numeric
initialoutlay numeric
monthlymaintenance numeric
Expected Results
memid surname firstname joindate
24 Sarwin Ramnaresh 2012-09-01 08:44:42
26 Jones Douglas 2012-09-02 18:43:05
27 Rumney Henrietta 2012-09-05 08:42:35
28 Farrell David 2012-09-15 08:22:05
29 Worthington-Smyth Henry 2012-09-17 12:27:15
30 Purview Millicent 2012-09-18 19:04:01
33 Tupperware Hyacinth 2012-09-18 19:32:05
35 Hunt John 2012-09-19 11:32:45
Your Answer
Hint Help Save Run Query
select memid, surname, firstname, joindate
from cd.members where joindate>='2012-09-01' ;
Activate Windows
Go to PC settings to activate Windows.
11:00 AM
12/6/2021


Advanced Databases
History
PostgreSQL exercises
https://pgexercises.com/questions/basic/unique.html
How can you produce an ordered list of the first 10 surnames in the members table? The list must not contain duplicates.
Schema reminder
cd.members
memid integer
surname character varying(200)
firstname character varying(200)
address character varying(300)
zipcode integer
telephone character varying(20)
recommendedby integer
joindate timestamp
cd.bookings
facid integer
memid integer
starttime timestamp
slots integer
cd.facilities
facid integer
name character varying(100)
membercost numeric
guestcost numeric
initialoutlay numeric
monthlymaintenance numeric
Expected Results
surname
Bader
Baker
Boothe
Butters
Coplin
Crumpet
Dare
Farrell
GUEST
Your Answer
Hint Help Save Run Query
select distinct surname from cd.members order by surname limit 10;
surname
Bader
Baker
Activate Windows
Go to PC settings to activate Windows.
11:02 AM
12/6/2021


Advanced Databases
History
PostgreSQL exercises
https://pgexercises.com/questions/basic/union.html
Question
You, for some reason, want a combined list of all surnames and all facility names. Yes, this is a contrived example :-).
Produce that list!
Schema reminder
cd.members
memid integer
surname character varying(200)
firstname character varying(200)
address character varying(300)
zipcode integer
telephone character varying(20)
recommendedby integer
joindate timestamp
cd.bookings
facid integer
memid integer
starttime timestamp
slots integer
cd.facilities
facid integer
name character varying(100)
membercost numeric
guestcost numeric
initialoutlay numeric
monthlymaintenance numeric
Expected Results
surname
Tennis Court 2
Worthington-Smyth
Badminton Court
Pinker
Dare
Bader
Mackenzie
Your Answer
Hint
Help
Save
Run Query
select surname
from cd.members
union
select name
from cd.facilities;
surname
Hunt
Farrell
Tennis Court 2
Activate Windows
Go to PC settings to activate Windows.
11:09 AM
12/6/2021


Advanced Databases
History
PostgreSQL exercises
https://pgexercises.com/questions/basic/agg.html
Question
You'd like to get the signup date of your last member. How can you retrieve this information?
Schema reminder
cd.members
memid integer
surname character varying(200)
firstname character varying(200)
address character varying(300)
zipcode integer
telephone character varying(20)
recommendedby integer
joindate timestamp
cd.bookings
facid integer
memid integer
starttime timestamp
slots integer
cd.facilities
facid integer
name character varying(100)
membercost numeric
guestcost numeric
initialoutlay numeric
monthlymaintenance numeric
Expected Results
latest
2012-09-26 18:08:45
Your Answer
Hint
Help
Save
Run Query
select max(joindate) as latest
from cd.members;
latest
2012-09-26 18:08:45
Activate Windows
Go to PC settings to activate Windows.
11:11 AM
12/6/2021

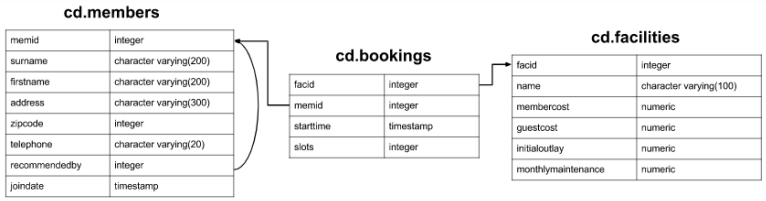
Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/basic/agg2.html

Question

You'd like to get the first and last name of the last member(s) who signed up - not just the date. How can you do that?

Schema reminder



Expected Results

firstname	surname	joindate
Darren	Smith	2012-09-26 18:08:45

Your Answer

Hint Help Save Run Query

```
select firstname, surname ,joindate from cd.members
where joindate = (select max(joindate)from cd.members);
```

firstname	surname	joindate
Darren	Smith	2012-09-26 18:08:45

11:18 AM 12/6/2021

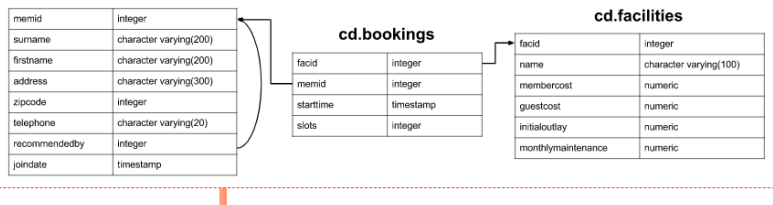
Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/joins/simplejoin.html

Question

How can you produce a list of the start times for bookings by members named 'David Farrell'?

Schema reminder



Expected Results

starttime
2012-09-18 09:00:00
2012-09-18 17:30:00
2012-09-18 13:30:00
2012-09-18 20:00:00
2012-09-19 09:30:00
2012-09-19 15:00:00
2012-09-19 12:00:00
2012-09-20 15:30:00

Your Answer

Hint Help Save Run Query

```
select bks.starttime
    from
        cd.bookings bks
        inner join cd.members mems
            on mems.memid = bks.memid
    where
        mems.firstname='David'
        and mems.surname='Farrell';
```


11:20 AM 12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/joins/simplejoin2.html

Question

How can you produce a list of the start times for bookings for tennis courts, for the date '2012-09-21'? Return a list of start time and facility name pairings, ordered by the time.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

start	name
2012-09-21 08:00:00	Tennis Court 1
2012-09-21 08:00:00	Tennis Court 2
2012-09-21 09:30:00	Tennis Court 1
2012-09-21 10:00:00	Tennis Court 2
2012-09-21 11:30:00	Tennis Court 2
2012-09-21 12:00:00	Tennis Court 1
2012-09-21 13:30:00	Tennis Court 1

Your Answer

Hint Help Save Run Query

```
select bks.starttime as start, facs.name as name
    from
        cd.facilities facs
        inner join cd.bookings bks
            on facs.facid = bks.facid
    where
        facs.name in ('Tennis Court 2','Tennis Court 1') and
        bks.starttime >= '2012-09-21' and
        bks.starttime < '2012-09-22'
order by bks.starttime;
```

start	name
2012-09-21 08:00:00	Tennis Court 1

11:23 AM 12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/joins/self.html

Question

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

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

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

Your Answer

Hint Help Save Run Query

```
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;
```

firstname	surname
Florence	Bader
Timothy	Baker
Gerald	Butters

11:24 AM 12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/joins/self2.html

Question

How can you output a list of all members, including the individual who recommended them (if any)? Ensure that results are ordered by (surname, firstname).

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

memfname	memsname	recfname	recsname
Florence	Bader	Ponder	Stibbons
Anne	Baker	Ponder	Stibbons
Timothy	Baker	Jemima	Farrell
Tim	Boothe	Tim	Rownam
Gerald	Butters	Darren	Smith
Joan	Coplin	Timothy	Baker
Erica	Crumpet	Tracy	Smith

Your Answer

Hint | Help | Save | Run Query

```
select mems.firstname as memfname, mems.surname as memsname, recs.firstname as recfname,
recs.surname as recsname
    from
        cd.members mems
        left outer join cd.members recs
            on recs.memid = mems.recommendedby
order by memsname, memfname;
```

memfname	memsname	recfname	recsname
Florence	Bader	Ponder	Stibbons

11:25 AM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/joins/threejoin.html

Question

How can you produce a list of all members who have used a tennis court? Include in your output the name of the court, and the name of the member formatted as a single column. Ensure no duplicate data, and order by the member name followed by the facility name.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

member	facility
Anne Baker	Tennis Court 1
Anne Baker	Tennis Court 2
Burton Tracy	Tennis Court 1
Burton Tracy	Tennis Court 2
Charles Owen	Tennis Court 1
Charles Owen	Tennis Court 2
Darren Smith	Tennis Court 2

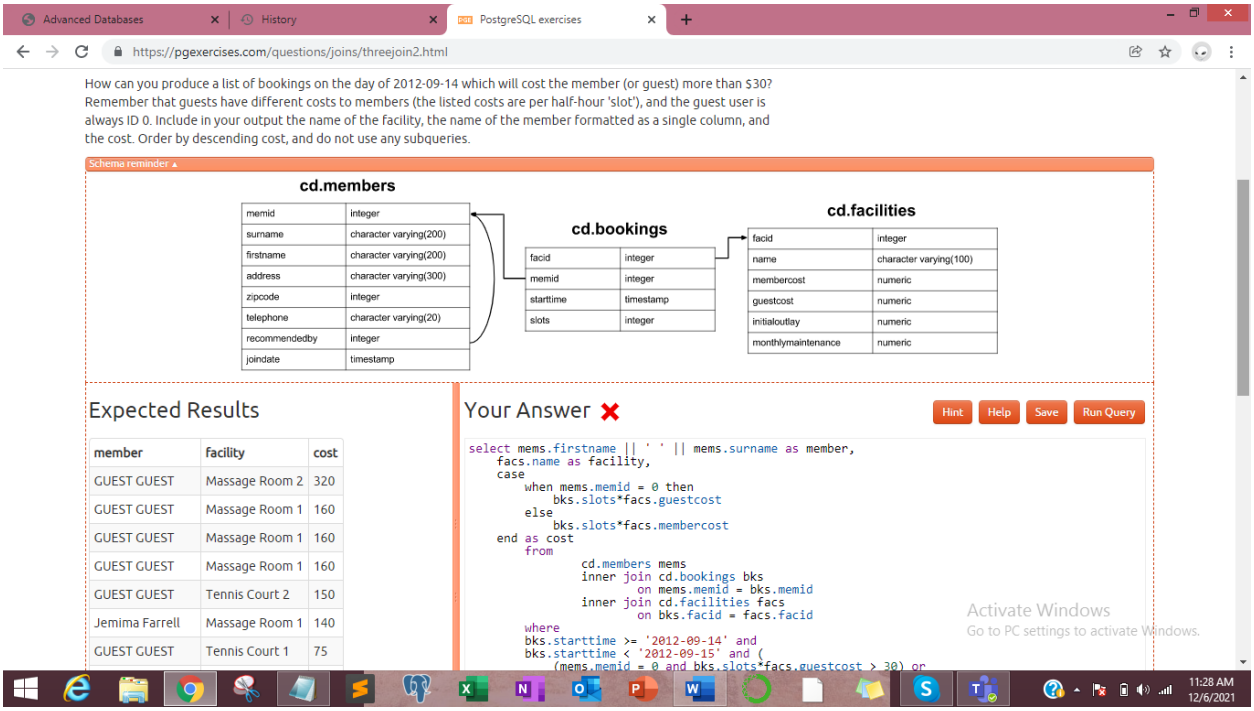
Your Answer

Hint | Help | Save | Run Query

```
select distinct mems.firstname || ' ' || mems.surname as member, facs.name as facility
    from
        cd.members mems
        inner join cd.bookings bks
            on mems.memid = bks.memid
        inner join cd.facilities facs
            on bks.facid = facs.facid
    where
        facs.name in ('Tennis Court 2','Tennis Court 1')
order by member, facility
```

member	facility

11:27 AM
12/6/2021




```
select mems.firstname || ' ' || mems.surname as member,
    facs.name as facility,
    case
        when mems.memid = 0 then
            bks.slots*facs.guestcost
        else
            bks.slots*facs.membercost
    end as cost
        from
                cd.members mems
                inner join cd.bookings bks
                        on mems.memid = bks.memid
                inner join cd.facilities facs
                        on bks.facid = facs.facid
        where
        bks.starttime >= '2012-09-14' and
        bks.starttime < '2012-09-15' and (
            (mems.memid = 0 and bks.slots*facs.guestcost > 30) or
            (mems.memid != 0 and bks.slots*facs.membercost > 30)
        )
order by cost desc;
```

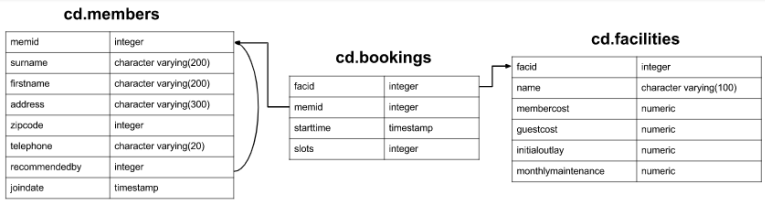
Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/joins/sub.html

Question

How can you output a list of all members, including the individual who recommended them (if any), without using any joins? Ensure that there are no duplicates in the list, and that each firstname + surname pairing is formatted as a column and ordered.

Schema reminder



Expected Results

member	recommender
Anna Mackenzie	Darren Smith
Anne Baker	Ponder Stibbons
Burton Tracy	
Charles Owen	Darren Smith
Darren Smith	
David Farrell	
David Jones	Janice Joplette

Your Answer

Hint Help Save Run Query

```
select distinct mems.firstname || ' ' ||  mems.surname as member,
    (select recs.firstname || ' ' || recs.surname as recommender
        from cd.members recs
        where recs.memid = mems.recommendedby
    )
    from
        cd.members mems
order by member;
```

member	recommender
Anna Mackenzie	Darren Smith
Anne Baker	Ponder Stibbons

11:30 AM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/joins/tjsub.html

Question

The Produce a list of costly bookings exercise contained some messy logic: we had to calculate the booking cost in both the `WHERE` clause and the `CASE` statement. Try to simplify this calculation using subqueries. For reference, the question was:

How can you produce a list of bookings on the day of 2012-09-14 which will cost the member (or guest) more than $30? Remember that guests have different costs to members (the listed costs are per half-hour 'slot'), and the guest user is always ID 0. Include in your output the name of the facility, the name of the member formatted as a single column, and the cost. Order by descending cost.

Schema reminder

Expected Results

member	facility	cost
GUEST GUEST	Massage Room 2	320
GUEST GUEST	Massage Room 1	160
GUEST GUEST	Massage Room 1	160

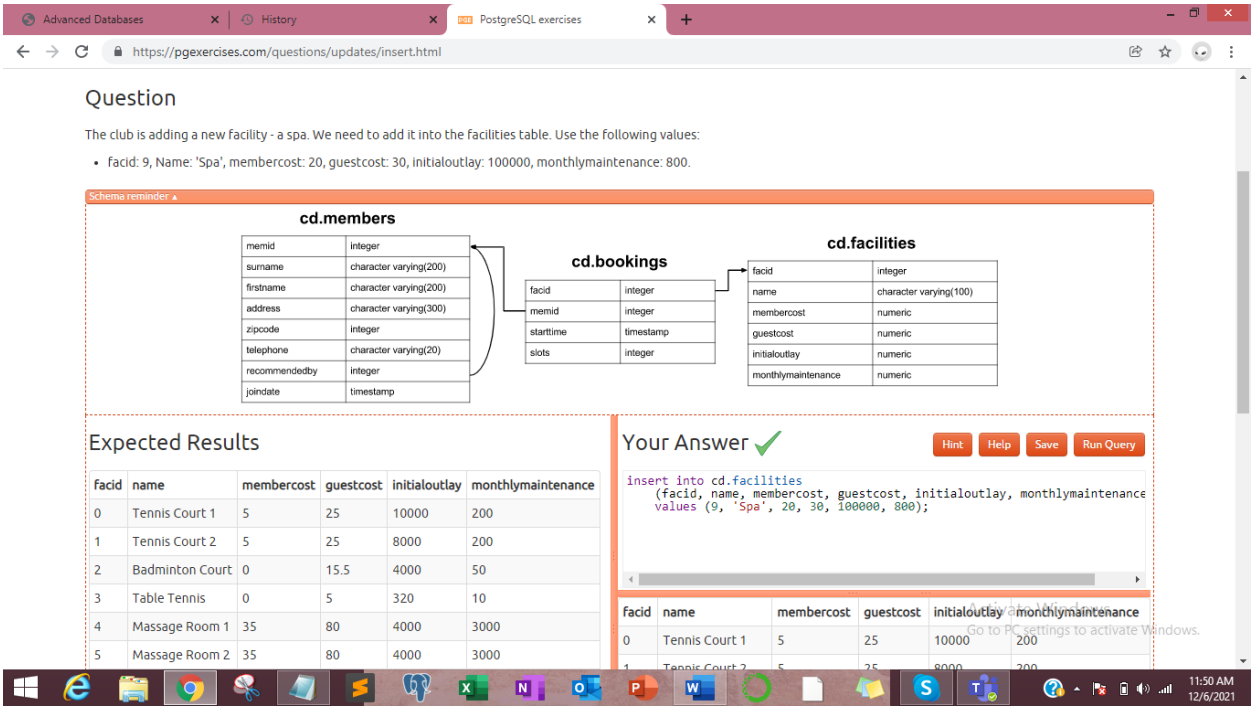
Your Answer

Hint Help Save Run Query

```
        from
            cd.members mems
            inner join cd.bookings bks
                on mems.memid = bks.memid
            inner join cd.facilities facs
                on bks.facid = facs.facid
        where
            bks.starttime >= '2012-09-14' and
            bks.starttime < '2012-09-15'
    ) as bookings
```


11:33 AM
12/6/2021

```sql
select member, facility, cost from (
        select
                mems.firstname || ' ' || mems.surname as member,
                facs.name as facility,
                case
                        when mems.memid = 0 then
                                bks.slots*facs.guestcost
                        else
                                bks.slots*facs.membercost
                end as cost
                from
                        cd.members mems
                        inner join cd.bookings bks
                                on mems.memid = bks.memid
                        inner join cd.facilities facs
                                on bks.facid = facs.facid
                where
                        bks.starttime >= '2012-09-14' and
                        bks.starttime < '2012-09-15'
        ) as bookings
        where cost > 30
order by cost desc;
```



Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/updates/insert2.html

In the previous exercise, you learned how to add a facility. Now you're going to add multiple facilities in one command.
Use the following values:

- facid: 9, Name: 'Spa', membercost: 20, guestcost: 30, initialoutlay: 100000, monthlymaintenance: 800.
- facid: 10, Name: 'Squash Court 2', membercost: 3.5, guestcost: 17.5, initialoutlay: 5000, monthlymaintenance: 80.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

facid	name	membercost	guestcost	initialoutlay	monthlymaintenance
0	Tennis Court 1	5	25	10000	200
1	Tennis Court 2	5	25	8000	200
2	Badminton Court	0	15.5	4000	50
3	Table Tennis	0	5	320	10
4	Massage Room 1	35	80	4000	3000
5	Massage Room 2	35	80	4000	3000

Your Answer

Hint Help Save Run Query

```
insert into cd.facilities
    (facid, name, membercost, guestcost, initialoutlay, monthlymaintenance)
    values
        (9, 'Spa', 20, 30, 100000, 800),
        (10, 'Squash Court 2', 3.5, 17.5, 5000, 80);
```

facid	name	membercost	guestcost	initialoutlay	monthlymaintenance
0	Tennis Court 1	5	25	10000	200
1	Tennis Court 2	5	25	8000	200

11:51 AM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/updates/insert3.html

Question

Let's try adding the spa to the facilities table again. This time, though, we want to automatically generate the value for the next facid, rather than specifying it as a constant. Use the following values for everything else:

- Name: 'Spa', membercost: 20, guestcost: 30, initialoutlay: 100000, monthlymaintenance: 800.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

facid	name	membercost	guestcost	initialoutlay	monthlym
0	Tennis Court 1	5	25	10000	200
1	Tennis Court 2	5	25	8000	200
2	Badminton Court	0	15.5	4000	50
3	Table Tennis	0	5	320	10
4	Massage Room 1	35	80	4000	3000
5	Massage Room 2	35	80	4000	3000

Your Answer

Hint Help Save Run Query

```
insert into cd.facilities
    (facid, name, membercost, guestcost, initialoutlay, monthlymaintenance)
    select (select max(facid) from cd.facilities)+1, 'Spa', 20, 30, 100000, 800;
```

facid	name	membercost	guestcost	initialoutlay	monthlymaintenance
0	Tennis Court 1	5	25	10000	200

11:53 AM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/updates/updatemultiple.html

Question

We want to increase the price of the tennis courts for both members and guests. Update the costs to be 6 for members, and 30 for guests.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

facid	name	membercost	guestcost	initialoutlay	monthlymaintenance
0	Tennis Court 1	6	30	10000	200
1	Tennis Court 2	6	30	8000	200
2	Badminton Court	0	15.5	4000	50
3	Table Tennis	0	5	320	10
4	Massage Room 1	35	80	4000	3000
5	Massage Room 2	35	80	4000	3000
6	Squash Court	3.5	17.5	5000	80

Your Answer

Hint | Help | Save | Run Query

```
update cd.facilities
    set
        membercost = 6,
        guestcost = 30
    where facid in (0,1);
```

facid	name	membercost	guestcost	initialoutlay	monthlymaintenance
0	Tennis Court 1	6	30	10000	200
1	Tennis Court 2	6	30	8000	200

11:57 AM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/updates/updatecalculated.html

Question

We want to alter the price of the second tennis court so that it costs 10% more than the first one. Try to do this without using constant values for the prices, so that we can reuse the statement if we want to.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

facid	name	membercost	guestcost	initialoutlay	monthlymair
0	Tennis Court 1	5	25	10000	200
1	Tennis Court 2	5.5	27.5	8000	200
2	Badminton Court	0	15.5	4000	50
3	Table Tennis	0	5	320	10
4	Massage Room 1	35	80	4000	3000
5	Massage Room 2	35	80	4000	3000
6	Squash Court	3.5	17.5	5000	80

Your Answer

Hint | Help | Save | Run Query

```
update cd.facilities facs
    set
        membercost = (select membercost * 1.1 from cd.facilities where facid = 0),
        guestcost = (select guestcost * 1.1 from cd.facilities where facid = 0)
    where facs.facid = 1;
```


11:58 AM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/updates/delete.html

Question

As part of a clearout of our database, we want to delete all bookings from the cd.bookings table. How can we accomplish this?

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

bookid	facid	memid	starttime	slots

Your Answer

Hint Help Save Run Query

```
delete from cd.bookings;
```


11:58 AM 12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/updates/deletewh.html

Question

We want to remove member 37, who has never made a booking, from our database. How can we achieve that?

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

memid	surname	firstname	address	zipcode	telephone	recomme
0	GUEST	GUEST	GUEST	0	(000) 000-0000	
1	Smith	Darren	8 Bloomsbury Close, Boston	4321	555-555-5555	
2	Smith	Tracy	8 Bloomsbury Close, New York	4321	555-555-5555	
3	Rownam	Tim	23 Highway Way, Boston	23423	(844) 693-0723	
4	Joplette	Janice	20 Crossing Road, New York	234	(833) 942-4710	1
5	Butters	Gerald	1065 Huntingdon Avenue, Boston	56754	(844) 078-4130	1
6	Tracy	Burton	3 Tunisia Drive, Boston	45678	(822) 354-9973	

Your Answer

Hint Help Save Run Query

```
delete from cd.members where memid = 37;
```

memid	surname	firstname	address	zipcode
0	GUEST	GUEST	GUEST	0
1	Smith	Darren	8	4321

11:59 AM 12/6/2021

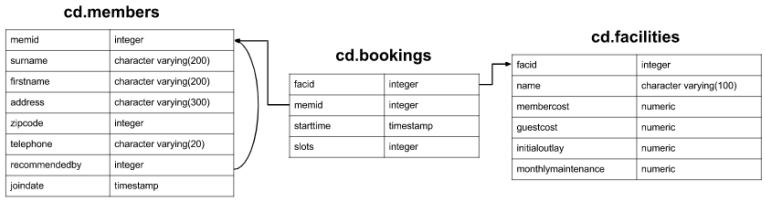
Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/updates/deletewh2.html

Question

In our previous exercises, we deleted a specific member who had never made a booking. How can we make that more general, to delete all members who have never made a booking?

Schema reminder



Expected Results

memid	surname	firstname	address
0	GUEST	GUEST	GUEST
1	Smith	Darren	8 Bloomsbury Close, Boston
2	Smith	Tracy	8 Bloomsbury Close, New York
3	Rownam	Tim	23 Highway Way, Boston
4	Joplette	Janice	20 Crossing Road, New York
5	Butters	Gerald	1065 Huntingdon Avenue, Boston
6	Tracy	Burton	3 Tunisia Drive, Boston

Your Answer

Hint Help Save Run Query

```
delete from cd.members where memid not in (select memid from cd.bookings);
```


12:00 PM
12/6/2021

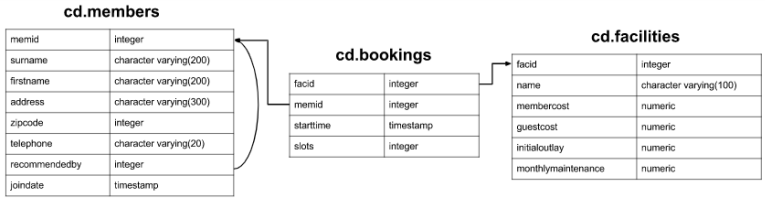
Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/count.html

Question

For our first foray into aggregates, we're going to stick to something simple. We want to know how many facilities exist - simply produce a total count.

Schema reminder



Expected Results

count
9

Your Answer

Hint Help Save Run Query

```
select count(*) from cd.facilities;
```


12:01 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/count2.html

Question

Produce a count of the number of facilities that have a cost to guests of 10 or more.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

count
6

Your Answer

Hint Help Save Run Query

```
select count(*) from cd.facilities where guestcost >= 10;
```


12:01 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/count3.html

Question

Produce a count of the number of recommendations each member has made. Order by member ID.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

recommendedby	count
1	5
2	3
3	1
4	2
5	1
6	1
9	2

Your Answer

Hint Help Save Run Query

```
select recommendedby, count(*)
    from cd.members
    where recommendedby is not null
    group by recommendedby
order by recommendedby;
```

recommendedby	count
1	5
2	3
3	1

12:07 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/fachours.html

Question

Produce a list of the total number of slots booked per facility. For now, just produce an output table consisting of facility id and slots, sorted by facility id.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

facid	Total Slots
0	1320
1	1278
2	1209
3	830
4	1404
5	228
6	1104

Your Answer

Hint Help Save Run Query

```
select facid, sum(slots) as "Total Slots"
    from cd.bookings
    group by facid
order by facid;
```

facid	Total Slots
0	1320
1	1278
2	1209

12:09 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/fachoursbymonth.html

Question

Produce a list of the total number of slots booked per facility in the month of September 2012. Produce an output table consisting of facility id and slots, sorted by the number of slots.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

facid	Total Slots
5	122
3	422
7	426
8	471
6	540
2	570

Your Answer

Hint Help Save Run Query

```
select facid, sum(slots) as "Total Slots"
    from cd.bookings
    where
        starttime >= '2012-09-01'
        and starttime < '2012-10-01'
    group by facid
order by sum(slots);
```

facid	Total Slots
5	122
3	422

12:10 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/fachoursbymonth2.html

Question

Produce a list of the total number of slots booked per facility per month in the year of 2012. Produce an output table consisting of facility id and slots, sorted by the id and month.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

facid	month	Total Slots
0	7	270
0	8	459
0	9	591
1	7	207
1	8	483
1	9	588
2	7	180

Your Answer

Hint | Help | Save | Run Query

```
select facid, extract(month from starttime) as month, sum(slots) as "Total Slots"
    from cd.bookings
    where extract(year from starttime) = 2012
    group by facid, month
order by facid, month;
```

facid	month	Total Slots
0	7	270
0	8	459

12:10 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/members1.html

Question

Find the total number of members (including guests) who have made at least one booking.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

count
30

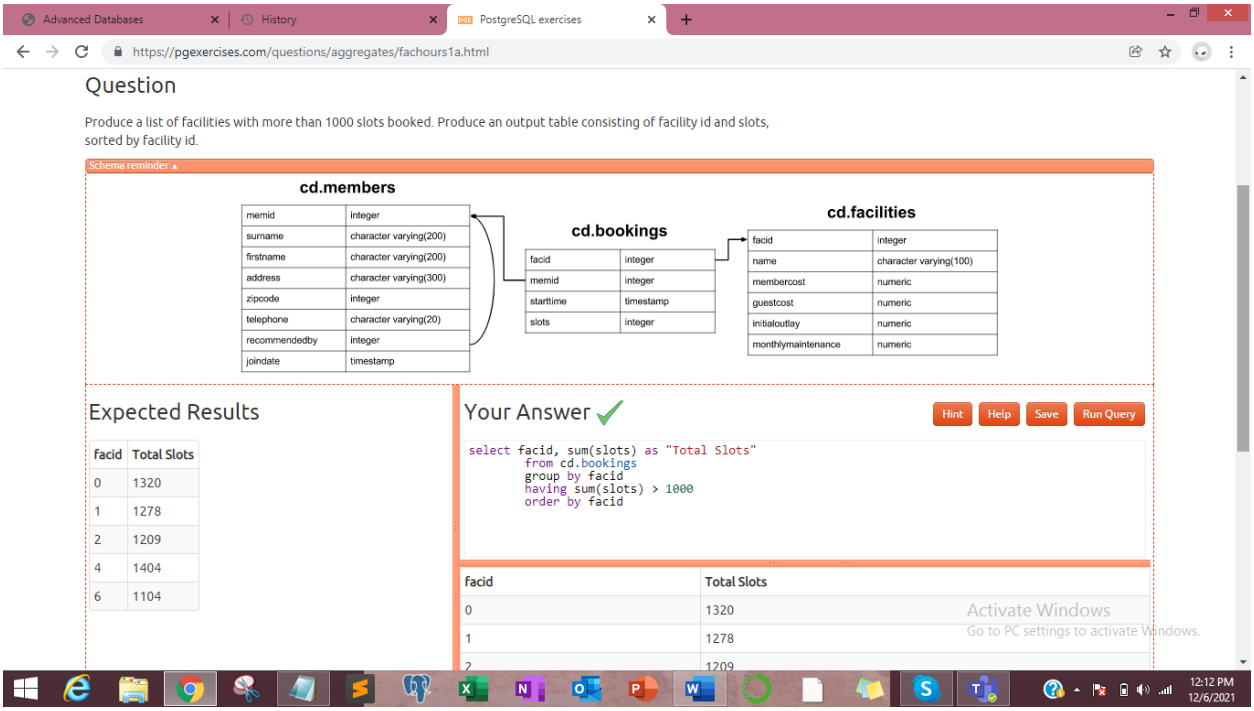
Your Answer

Hint | Help | Save | Run Query

```
select count(distinct memid) from cd.bookings
```

count
30

12:11 PM
12/6/2021


Advanced Databases
History
PostgreSQL exercises
https://pgexercises.com/questions/aggregates/fachours1a.html
Question
Produce a list of facilities with more than 1000 slots booked. Produce an output table consisting of facility id and slots, sorted by facility id.
Schema reminder
cd.members
memid integer
surname character varying(200)
firstname character varying(200)
address character varying(300)
zipcode integer
telephone character varying(20)
recommendedby integer
joindate timestamp
cd.bookings
facid integer
memid integer
starttime timestamp
slots integer
cd.facilities
facid integer
name character varying(100)
membercost numeric
guestcost numeric
initialoutlay numeric
monthlymaintenance numeric
Expected Results
facid Total Slots
0 1320
1 1278
2 1209
4 1404
6 1104
Your Answer
Hint
Help
Save
Run Query
select facid, sum(slots) as "Total Slots"
from cd.bookings
group by facid
having sum(slots) > 1000
order by facid
facid Total Slots
0 1320
1 1278
Activate Windows
Go to PC settings to activate Windows.
12:12 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/facrev.html

Question

Produce a list of facilities along with their total revenue. The output table should consist of facility name and revenue, sorted by revenue. Remember that there's a different cost for guests and members!

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

name	revenue
Table Tennis	180
Snooker Table	240
Pool Table	270
Badminton Court	1906.5
Squash Court	13468.0
Tennis Court 1	13860
Tennis Court 2	14310

Your Answer

Hint Help Save Run Query

```
select facs.name, sum(slots * case
            when memid = 0 then facs.guestcost
            else facs.membercost
        end) as revenue
    from cd.bookings bks
    inner join cd.facilities facs
        on bks.facid = facs.facid
    group by facs.name
order by revenue;
```


12:20 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/facrev2.html

Question

Produce a list of facilities with a total revenue less than 1000. Produce an output table consisting of facility name and revenue, sorted by revenue. Remember that there's a different cost for guests and members!

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

name	revenue
Table Tennis	180
Snooker Table	240
Pool Table	270

Your Answer

Hint Help Save Run Query

```
select name, revenue from (
    select facs.name, sum(case
                when memid = 0 then slots * facs.guestcost
                else slots * membercost
            end) as revenue
        from cd.bookings bks
        inner join cd.facilities facs
            on bks.facid = facs.facid
        group by facs.name
    ) as agg where revenue < 1000
order by revenue;
```


12:20 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/fachours2.html

Question

Output the facility id that has the highest number of slots booked. For bonus points, try a version without a `LIMIT` clause.
This version will probably look messy!

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

facid	Total Slots
4	1404

Your Answer

Hint Help Save Run Query

```
select facid, sum(slots) as "Total Slots"
    from cd.bookings
    group by facid
order by sum(slots) desc
LIMIT 1;
```

facid	Total Slots
4	1404

12:22 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/fachoursbymonth3.html

Question

Produce a list of the total number of slots booked per facility per month in the year of 2012. In this version, include output rows containing totals for all months per facility, and a total for all months for all facilities. The output table should consist of facility id, month and slots, sorted by the id and month. When calculating the aggregated values for all months and all facids, return null values in the month and facid columns.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

facid	month	slots
0	7	270
0	8	459
0	9	591
0		1320
1	7	207
1	8	483

Your Answer

Hint Help Save Run Query

```
select facid, extract(month from starttime) as month, sum(slots) as slots
    from cd.bookings
    where
        starttime >= '2012-01-01'
        and starttime < '2013-01-01'
    group by rollup(facid, month)
order by facid, month;
```

facid	month	slots
0	7	270

12:25 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/fachours3.html

Question

Produce a list of the total number of *hours* booked per facility, remembering that a slot lasts half an hour. The output table should consist of the facility id, name, and hours booked, sorted by facility id. Try formatting the hours to two decimal places.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

facid	name	Total Hours
0	Tennis Court 1	660.00
1	Tennis Court 2	639.00
2	Badminton Court	604.50
3	Table Tennis	415.00
4	Massage Room 1	702.00
5	Massage Room 2	114.00
6	Squash Court	552.00

Your Answer

Hint | Help | Save | Run Query

```
select facs.facid, facs.name,
    trim(to_char(sum(bks.slots)/2.0, '9999999999999999D99')) as "Total Hours"

    from cd.bookings bks
    inner join cd.facilities facs
        on facs.facid = bks.facid
    group by facs.facid, facs.name
order by facs.facid;
```

facid	name	Total Hours
0	Tennis Court 1	660.00
1	Tennis Court 2	639.00

12:27 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/nbooking.html

Question

Produce a list of each member name, id, and their first booking after September 1st 2012. Order by member ID.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

surname	firstname	memid	starttime
GUEST	GUEST	0	2012-09-01 08:00:00
Smith	Darren	1	2012-09-01 09:00:00
Smith	Tracy	2	2012-09-01 11:30:00
Rownam	Tim	3	2012-09-01 16:00:00
Joplette	Janice	4	2012-09-01 15:00:00
Butters	Gerald	5	2012-09-02 12:30:00
Tracy	Burton	6	2012-09-01 15:00:00
Dare	Nancy	7	2012-09-01 13:30:00

Your Answer

Hint | Help | Save | Run Query

```
select mems.surname, mems.firstname, mems.memid, min(bks.starttime) as starttime
    from cd.bookings bks
    inner join cd.members mems on
        mems.memid = bks.memid
    where starttime >= '2012-09-01'
    group by mems.surname, mems.firstname, mems.memid
order by mems.memid;
```

surname	firstname	memid	starttime
GUEST	GUEST	0	2012-09-01 08:00:00
Smith	Darren	1	2012-09-01 09:00:00
Smith	Tracy	2	2012-09-01 11:30:00

12:28 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/countmembers.html

Question

Produce a list of member names, with each row containing the total member count. Order by join date, and include guest members.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

count	firstname	surname
31	GUEST	GUEST
31	Darren	Smith
31	Tracy	Smith
31	Tim	Rownam
31	Janice	Joplette
31	Gerald	Butters

Your Answer

Hint Help Save Run Query

```
select count(*) over(), firstname, surname
    from cd.members
order by joindate
```

count	firstname	surname
31	GUEST	GUEST
31	Darren	Smith

12:29 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/nummembers.html

Question

Produce a monotonically increasing numbered list of members (including guests), ordered by their date of joining. Remember that member IDs are not guaranteed to be sequential.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

row_number	firstname	surname
1	GUEST	GUEST
2	Darren	Smith
3	Tracy	Smith
4	Tim	Rownam
5	Janice	Joplette
6	Gerald	Butters
7	Burton	Tracy
8	Nancy	Dare

Your Answer

Hint Help Save Run Query

```
select row_number() over(order by joindate), firstname, surname
    from cd.members
order by joindate
```

row_number	firstname	surname
1	GUEST	GUEST
2	Darren	Smith
3	Tracy	Smith

12:30 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/fachours4.html

Question

Output the facility id that has the highest number of slots booked. Ensure that in the event of a tie, all tieing results get output.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

facid	total
4	1404

Your Answer

Hint Help Save Run Query

```
select facid, total from (
    select facid, sum(slots) total, rank() over (order by sum(slots) desc) rank
            from cd.bookings
        group by facid
    ) as ranked
    where rank = 1
```

facid	total
4	1404

12:30 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/rankmembers.html

Question

Produce a list of members (including guests), along with the number of hours they've booked in facilities, rounded to the nearest ten hours. Rank them by this rounded figure, producing output of first name, surname, rounded hours, rank. Sort by rank, surname, and first name.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

firstname	surname	hours	rank
GUEST	GUEST	1200	1
Darren	Smith	340	2
Tim	Rownam	330	3
Tim	Boothe	220	4
Tracy	Smith	220	4
Gerald	Butters	210	6

Your Answer

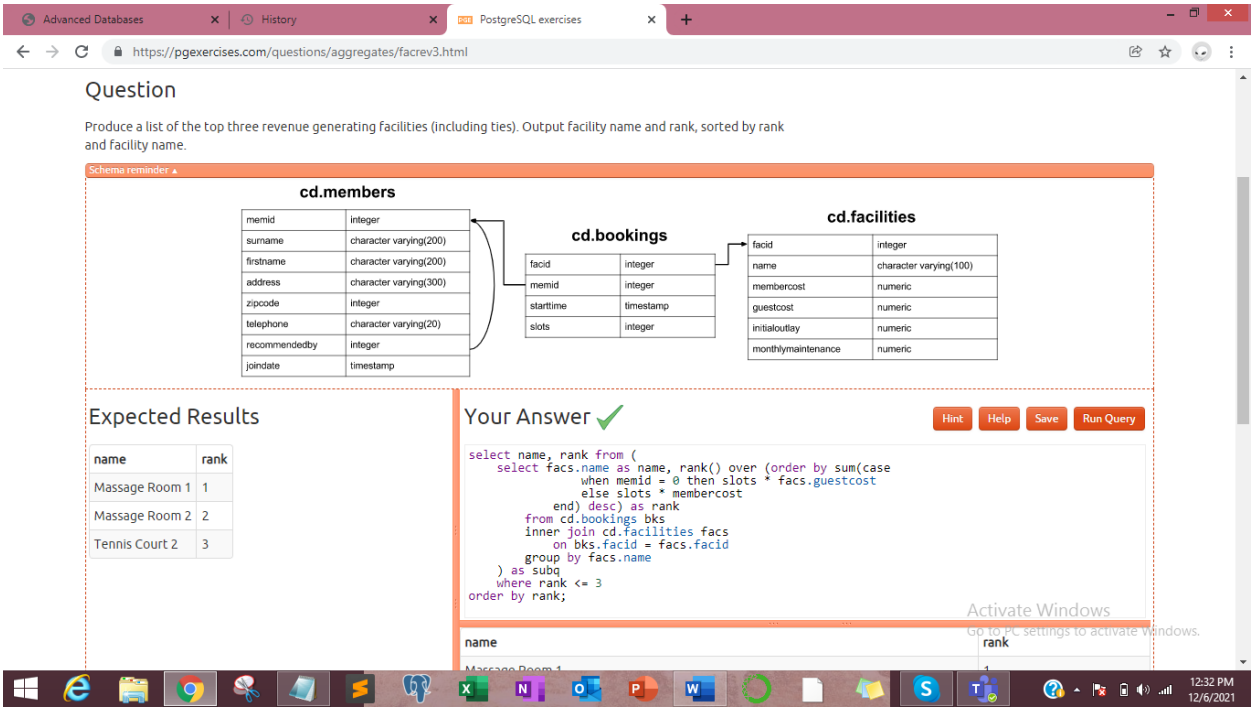
Hint Help Save Run Query

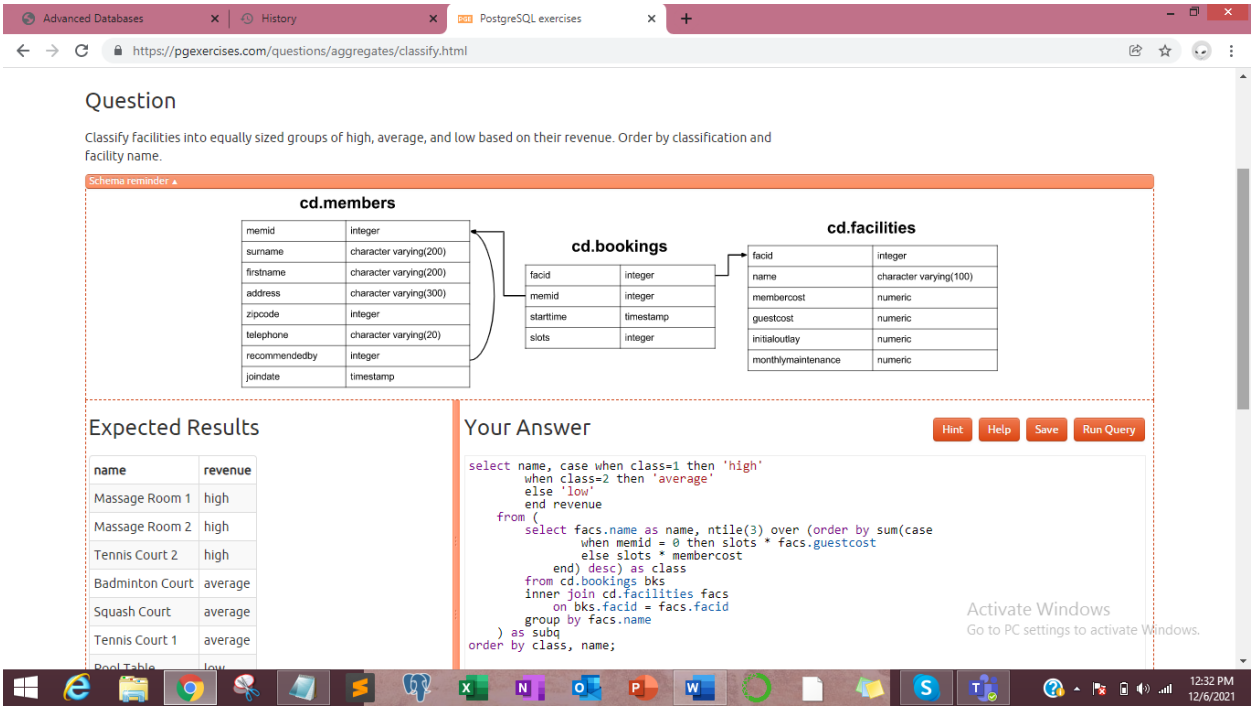
```
select firstname, surname,
    ((sum(bks.slots)+10)/20)*10 as hours,
    rank() over (order by ((sum(bks.slots)+10)/20)*10 desc) as rank

    from cd.bookings bks
    inner join cd.members mems
        on bks.memid = mems.memid
    group by mems.memid
order by rank, surname, firstname;
```

firstname	surname	hours	rank

12:31 PM
12/6/2021


Advanced Databases
History
PostgreSQL exercises
https://pgexercises.com/questions/aggregates/facrev3.html
Question
Produce a list of the top three revenue generating facilities (including ties). Output facility name and rank, sorted by rank and facility name.
Schema reminder
cd.members
memid integer
surname character varying(200)
firstname character varying(200)
address character varying(300)
zipcode integer
telephone character varying(20)
recommendedby integer
joindate timestamp
cd.bookings
facid integer
memid integer
starttime timestamp
slots integer
cd.facilities
facid integer
name character varying(100)
membercost numeric
guestcost numeric
initialoutlay numeric
monthlymaintenance numeric
Expected Results
name rank
Massage Room 1 1
Massage Room 2 2
Tennis Court 2 3
Your Answer
Hint Help Save Run Query
select name, rank from (
select facs.name as name, rank() over (order by sum(case
when memid = 0 then slots * facs.guestcost
else slots * membercost
end) desc) as rank
from cd.bookings bks
inner join cd.facilities facs
on bks.facid = facs.facid
group by facs.name
) as subq
where rank <= 3
order by rank;
Activate Windows
Go to PC settings to activate Windows.
name rank
12:32 PM
12/6/2021


Advanced Databases
History
PostgreSQL exercises
https://pgexercises.com/questions/aggregates/classify.html
Question
Classify facilities into equally sized groups of high, average, and low based on their revenue. Order by classification and facility name.
Schema reminder
cd.members
memid integer
surname character varying(200)
firstname character varying(200)
address character varying(300)
zipcode integer
telephone character varying(20)
recommendedby integer
joindate timestamp
cd.bookings
facid integer
memid integer
starttime timestamp
slots integer
cd.facilities
facid integer
name character varying(100)
membercost numeric
guestcost numeric
initialoutlay numeric
monthlymaintenance numeric
Expected Results
name revenue
Massage Room 1 high
Massage Room 2 high
Tennis Court 2 high
Badminton Court average
Squash Court average
Tennis Court 1 average
Pool Table low
Your Answer
Hint Help Save Run Query
select name, case when class=1 then 'high'
when class=2 then 'average'
else 'low'
end revenue
from (
select facs.name as name, ntile(3) over (order by sum(case
when memid = 0 then slots * facs.guestcost
else slots * membercost
end) desc) as class
from cd.bookings bks
inner join cd.facilities facs
on bks.facid = facs.facid
group by facs.name
) as subq
order by class, name;
Activate Windows
Go to PC settings to activate Windows.
12:32 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/payback.html

Question

Based on the 3 complete months of data so far, calculate the amount of time each facility will take to repay its cost of ownership. Remember to take into account ongoing monthly maintenance. Output facility name and payback time in months, order by facility name. Don't worry about differences in month lengths, we're only looking for a rough value here!

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

name	months
Badminton Court	6.8317677198975235
Massage Room 1	0.18885741265344664778
Massage Room 2	1.7621145374449339
Pool Table	5.3333333333333333
Snooker Table	6.9230769230769231

Your Answer

Hint Help Save Run Query

```
select  facs.name as name,
    facs.initialoutlay/((sum(case
            when memid = 0 then slots * facs.guestcost
            else slots * membercost
        end)/3) - facs.monthlymaintenance) as months
    from cd.bookings bks
    inner join cd.facilities facs
        on bks.facid = facs.facid
    group by facs.facid
order by name;
```


12:33 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/aggregates/rollingavg.html

Calculate a rolling average of total revenue

Question

For each day in August 2012, calculate a rolling average of total revenue over the previous 15 days. Output should contain date and revenue columns, sorted by the date. Remember to account for the possibility of a day having zero revenue. This one's a bit tough, so don't be afraid to check out the hint!

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

date	revenue
2012-08-01	1126.8333333333333333
2012-08-02	1153.0000000000000000

Your Answer

Hint Help Save Run Query

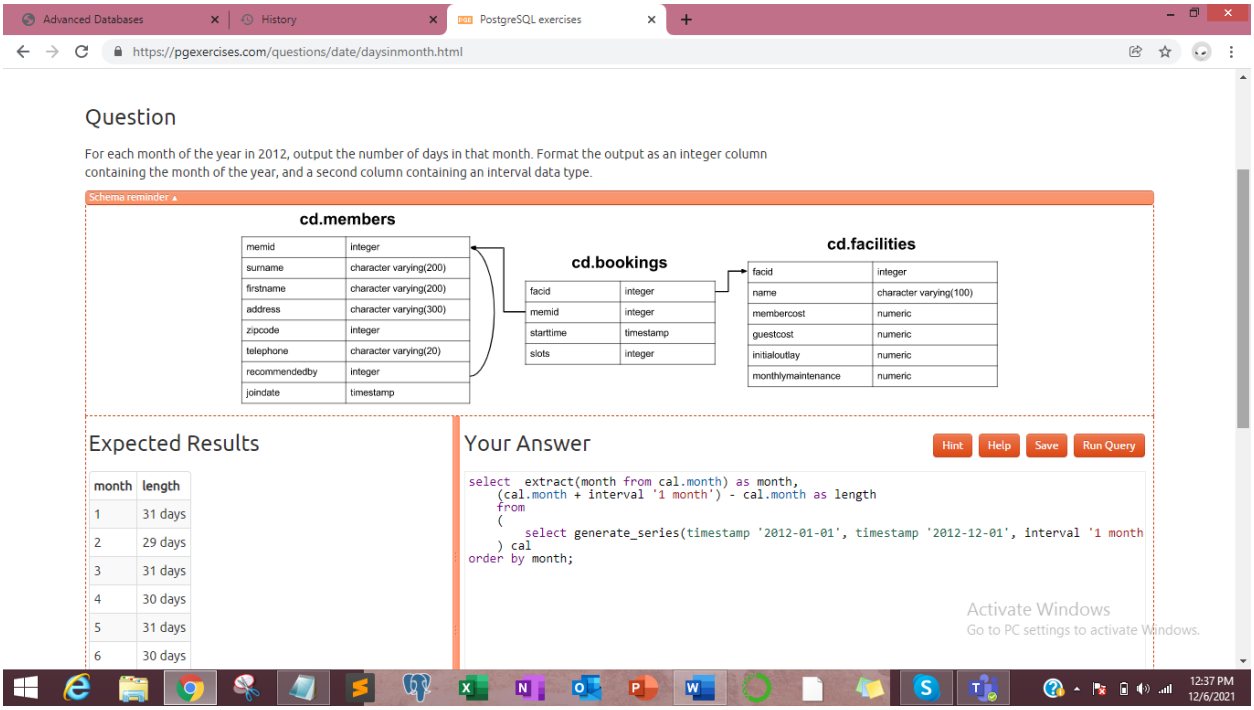
```
select  dategen.date,
    (
        -- correlated subquery that, for each day fed into it,
        -- finds the average revenue for the last 15 days
        select sum(case
            when memid = 0 then slots * facs.guestcost
```


12:34 PM
12/6/2021

```sql
select  dategen.date,
        (
                -- correlated subquery that, for each day fed into it,
                -- finds the average revenue for the last 15 days
                select sum(case
                        when memid = 0 then slots * facs.guestcost
                        else slots * membercost
                end) as rev

                from cd.bookings bks
                inner join cd.facilities facs
                        on bks.facid = facs.facid
                where bks.starttime > dategen.date - interval '14 days'
                        and bks.starttime < dategen.date + interval '1 day'
        )/15 as revenue
        from
        (
                -- generates a list of days in august
                select  cast(generate_series(timestamp '2012-08-01',
                        '2012-08-31','1 day') as date) as date
        )  as dategen
order by dategen.date;
```

TIMESTRAMP



Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/date/daysremaining.html

Question

For any given timestamp, work out the number of days remaining in the month. The current day should count as a whole day, regardless of the time. Use '2012-02-11 01:00:00' as an example timestamp for the purposes of making the answer. Format the output as a single interval value.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

remaining
19 days

Your Answer

Hint Help Save Run Query

```
select (date_trunc('month',ts.testts) + interval '1 month')
       - date_trunc('day', ts.testts) as remaining
    from (select timestamp '2012-02-11 01:00:00' as testts) ts
```

remaining
19 days

12:38 PM 12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/date/endtimes.html

Question

Return a list of the start and end time of the last 10 bookings (ordered by the time at which they end, followed by the time at which they start) in the system.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

starttime	endtime
2013-01-01 15:30:00	2013-01-01 16:00:00
2012-09-30 19:30:00	2012-09-30 20:30:00
2012-09-30 19:00:00	2012-09-30 20:30:00
2012-09-30 19:30:00	2012-09-30 20:00:00
2012-09-30 19:00:00	2012-09-30 20:00:00
2012-09-30 19:00:00	2012-09-30 20:00:00

Your Answer

Hint Help Save Run Query

```
select starttime, starttime + slots*(interval '30 minutes') endtime
    from cd.bookings
    order by endtime desc, starttime desc
    limit 10
```

starttime	endtime
2013-01-01 15:30:00	2013-01-01 16:00:00
2012-09-30 19:30:00	2012-09-30 20:30:00

12:38 PM 12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/date/bookingspermonth.html

Question

Return a count of bookings for each month, sorted by month

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

month	count
2012-07-01 00:00:00	658
2012-08-01 00:00:00	1472
2012-09-01 00:00:00	1913
2013-01-01 00:00:00	1

Your Answer

Hint Help Save Run Query

```
select date_trunc('month', starttime) as month, count(*)
    from cd.bookings
    group by month
    order by month
```

month	count
2012-07-01 00:00:00	658
2012-08-01 00:00:00	1472
2012-09-01 00:00:00	1913

12:39 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/date/utilisationpermonth.html

Question

Work out the utilisation percentage for each facility by month, sorted by name and month, rounded to 1 decimal place. Opening time is 8am, closing time is 8.30pm. You can treat every month as a full month, regardless of if there were some dates the club was not open.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

name	month	utilisation
Badminton Court	2012-07-01 00:00:00	23.2
Badminton Court	2012-08-01 00:00:00	59.2
Badminton Court	2012-09-01 00:00:00	76.0
Massage Room 1	2012-07-01 00:00:00	34.1
Massage Room 1	2012-08-01 00:00:00	63.5
Massage Room 1	2012-09-01 00:00:00	86.4

Your Answer

Hint Help Save Run Query

```
select name, month,
    round((100*slots)/
        cast(
            25*(cast((month + interval '1 month') as date)
            - cast (month as date)) as numeric),1) as utilisation
    from  (
        select facs.name as name, date_trunc('month', starttime) as month, sum(slots) as slots
            from cd.bookings bks
            inner join cd.facilities facs
                on bks.facid = facs.facid
            group by facs.facid, month
    ) as inn
order by name, month
```


12:39 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/string/concat.html

Question

Output the names of all members, formatted as 'Surname, Firstname'

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

name
GUEST, GUEST
Smith, Darren
Smith, Tracy
Rownam, Tim
Joplette, Janice
Butters, Gerald

Your Answer

Hint Help Save Run Query

```
select surname || ', ' || firstname as name from cd.members
```

name
GUEST, GUEST

12:41 PM
12/6/2021

Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/string/like.html

Question

Find all facilities whose name begins with 'Tennis'. Retrieve all columns.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

facid	name	membercost	guestcost	initialoutlay	monthlymaintenance
0	Tennis Court 1	5	25	10000	200
1	Tennis Court 2	5	25	8000	200

Your Answer

Hint Help Save Run Query

```
select * from cd.facilities where name like 'Tennis%';
```

facid	name	membercost	guestcost	initialoutlay	monthlymaintenance
0	Tennis Court 1	5	25	10000	200
1	Tennis Court 2	5	25	8000	200

12:41 PM
12/6/2021

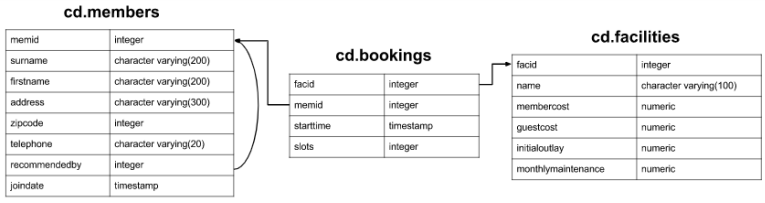
Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/string/case.html

Question

Perform a case-insensitive search to find all facilities whose name begins with 'tennis'. Retrieve all columns.

Schema reminder



Expected Results

facid	name	membercost	guestcost	initialoutlay	monthlymaintenance
0	Tennis Court 1	5	25	10000	200
1	Tennis Court 2	5	25	8000	200

Your Answer

Hint Help Save Run Query

```
select * from cd.facilities where upper(name) like 'TENNIS%';
```

facid	name	membercost	guestcost	initialoutlay	monthlymaintenance
0	Tennis Court 1	5	25	10000	200
1	Tennis Court 2	5	25	8000	200

12:42 PM
12/6/2021

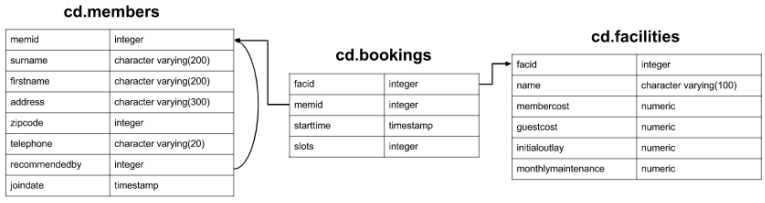
Advanced Databases | History | PostgreSQL exercises

https://pgexercises.com/questions/string/reg.html

Question

You've noticed that the club's member table has telephone numbers with very inconsistent formatting. You'd like to find all the telephone numbers that contain parentheses, returning the member ID and telephone number sorted by member ID.

Schema reminder



Expected Results

memid	telephone
0	(000) 000-0000
3	(844) 693-0723
4	(833) 942-4710
5	(844) 078-4130
6	(822) 354-9973
7	(833) 776-4001

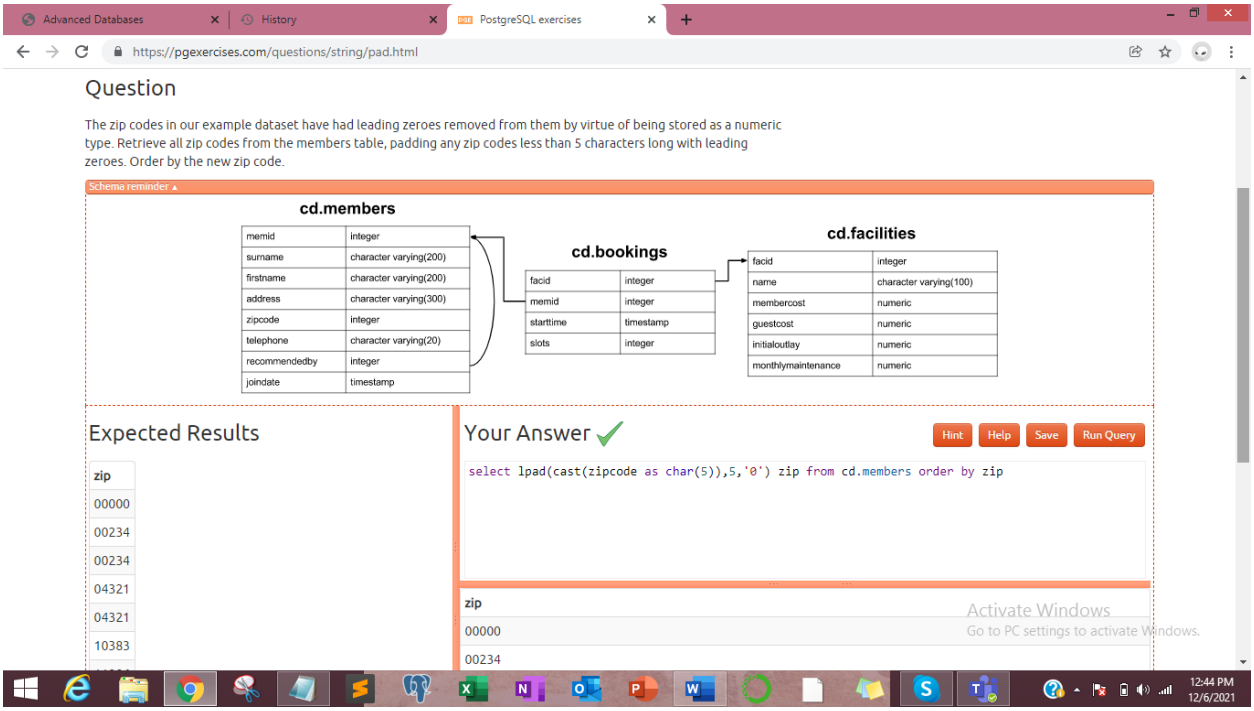
Your Answer

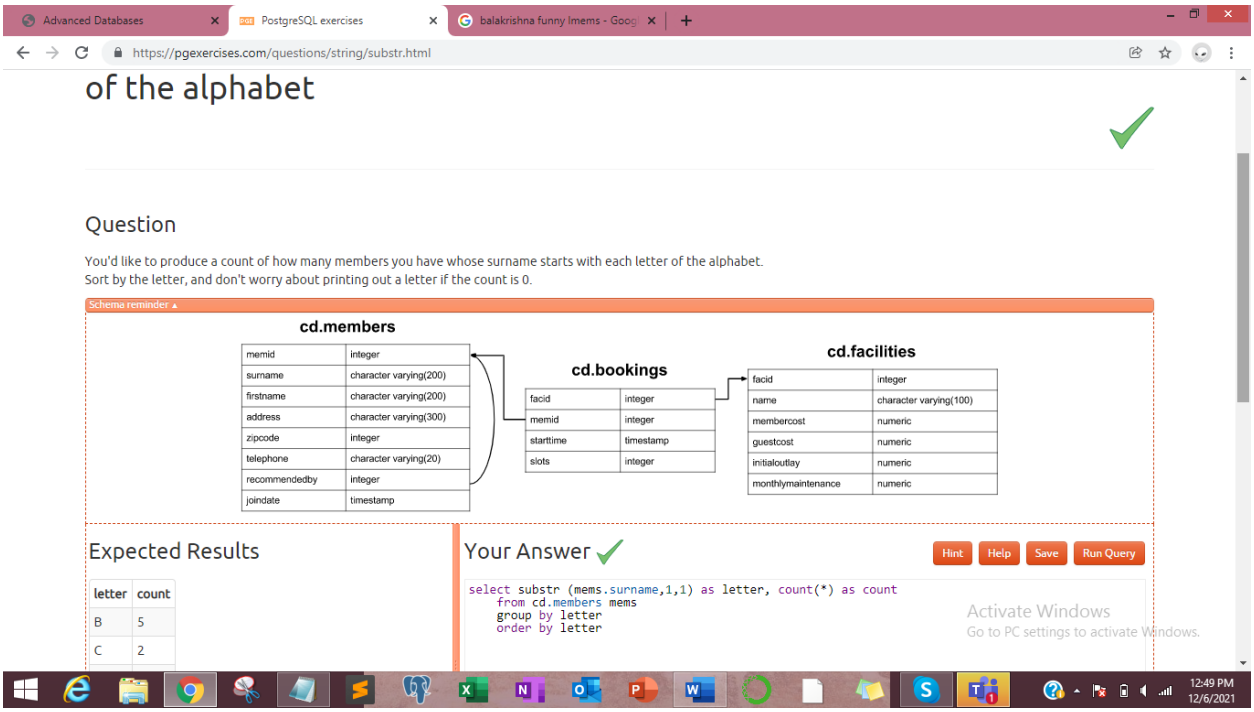
Hint Help Save Run Query

```
select memid, telephone from cd.members where telephone ~ '[()]';
```

memid	telephone
0	(000) 000-0000

12:42 PM
12/6/2021


Advanced Databases
History
PostgreSQL exercises
https://pgexercises.com/questions/string/pad.html
Question
The zip codes in our example dataset have had leading zeroes removed from them by virtue of being stored as a numeric type. Retrieve all zip codes from the members table, padding any zip codes less than 5 characters long with leading zeroes. Order by the new zip code.
Schema reminder
cd.members
memid integer
surname character varying(200)
firstname character varying(200)
address character varying(300)
zipcode integer
telephone character varying(20)
recommendedby integer
joindate timestamp
cd.bookings
facid integer
memid integer
starttime timestamp
slots integer
cd.facilities
facid integer
name character varying(100)
membercost numeric
guestcost numeric
initialoutlay numeric
monthlymaintenance numeric
Expected Results
zip
00000
00234
00234
04321
04321
10383
Your Answer
Hint
Help
Save
Run Query
select lpad(cast(zipcode as char(5)),5,'0') zip from cd.members order by zip
zip
00000
00234
Activate Windows
Go to PC settings to activate Windows.
12:44 PM
12/6/2021


Advanced Databases
PostgreSQL exercises
balakrishna funny lmems - Goog
https://pgexercises.com/questions/string/substr.html
of the alphabet
Question
You'd like to produce a count of how many members you have whose surname starts with each letter of the alphabet.
Sort by the letter, and don't worry about printing out a letter if the count is 0.
Schema reminder
cd.members
memid integer
surname character varying(200)
firstname character varying(200)
address character varying(300)
zipcode integer
telephone character varying(20)
recommendedby integer
joindate timestamp
cd.bookings
facid integer
memid integer
starttime timestamp
slots integer
cd.facilities
facid integer
name character varying(100)
membercost numeric
guestcost numeric
initialoutlay numeric
monthlymaintenance numeric
Expected Results
letter count
B 5
C 2
Your Answer
Hint
Help
Save
Run Query
select substr (mems.surname,1,1) as letter, count(*) as count
from cd.members mems
group by letter
order by letter
Activate Windows
Go to PC settings to activate Windows.
12:49 PM
12/6/2021

Advanced Databases | PostgreSQL exercises | balakrishna funny lmems - Goog

https://pgexercises.com/questions/recursive/getupward.html

Question

Find the upward recommendation chain for member ID 27: that is, the member who recommended them, and the member who recommended that member, and so on. Return member ID, first name, and surname. Order by descending member id.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

recommender	firstname	surname
20	Matthew	Genting
5	Gerald	Butters
1	Darren	Smith

Your Answer

Hint Help Save Run Query

```
with recursive recommenders(recommender) as (
    select recommendedby from cd.members where memid = 27
    union all
    select mems.recommendedby
        from recommenders recs
        inner join cd.members mems
            on mems.memid = recs.recommender
)
select recs.recommender, mems.firstname, mems.surname
    from recommenders recs
    inner join cd.members mems
        on recs.recommender = mems.memid
order by memid desc
```


12:51 PM
12/6/2021

Advanced Databases | PostgreSQL exercises | balakrishna funny lmems - Goog

https://pgexercises.com/questions/recursive/getdownward.html

Question

Find the downward recommendation chain for member ID 1: that is, the members they recommended, the members those members recommended, and so on. Return member ID and name, and order by ascending member id.

Schema reminder

cd.members

memid	integer
surname	character varying(200)
firstname	character varying(200)
address	character varying(300)
zipcode	integer
telephone	character varying(20)
recommendedby	integer
joindate	timestamp

cd.bookings

facid	integer
memid	integer
starttime	timestamp
slots	integer

cd.facilities

facid	integer
name	character varying(100)
membercost	numeric
guestcost	numeric
initialoutlay	numeric
monthlymaintenance	numeric

Expected Results

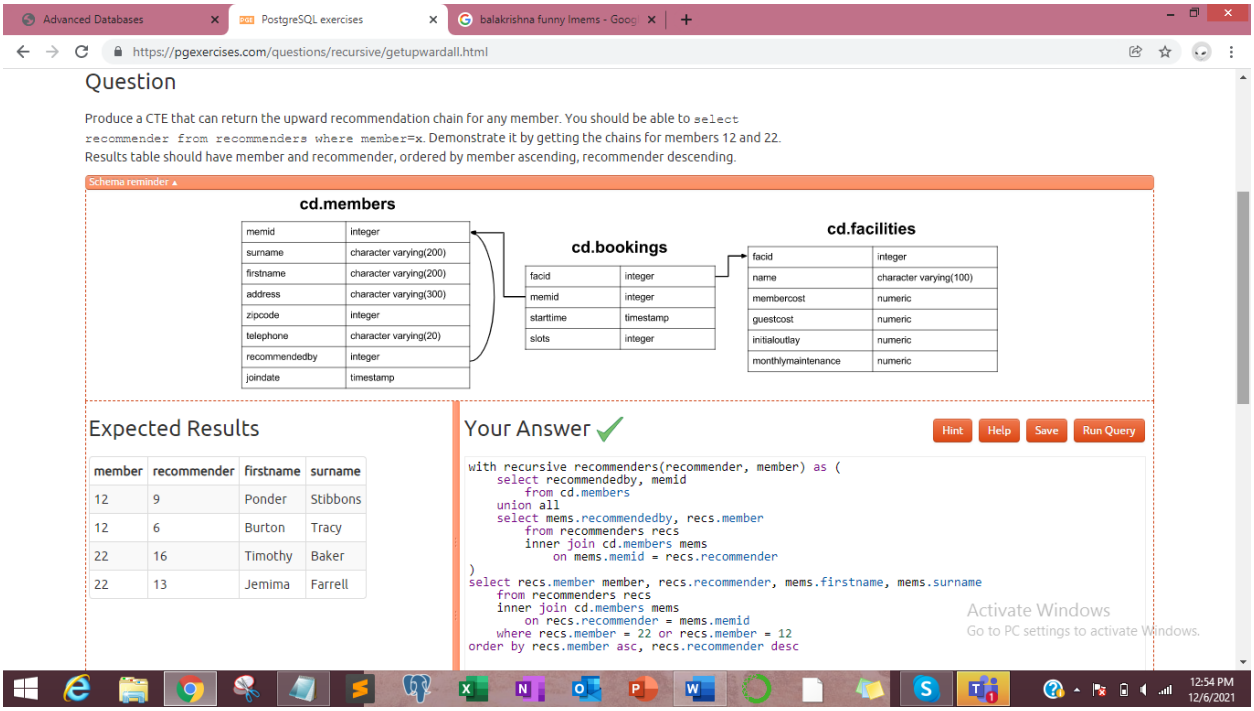
memid	firstname	surname
4	Janice	Joplette
5	Gerald	Butters
7	Nancy	Dare
10	Charles	Owen
11	David	Jones
14	Jack	Smith
20	Matthew	Genting

Your Answer ✓

Hint Help Save Run Query

```
with recursive recommendeds(memid) as (
    select memid from cd.members where recommendedby = 1
    union all
    select mems.memid
        from recommendeds recs
        inner join cd.members mems
            on mems.recommendedby = recs.memid
)
select recs.memid, mems.firstname, mems.surname
    from recommendeds recs
    inner join cd.members mems
        on recs.memid = mems.memid
order by memid
```


12:53 PM
12/6/2021


Advanced Databases
PostgreSQL exercises
balakrishna funny lmems - Goog
https://pgexercises.com/questions/recursive/getupwardall.html
Question
Produce a CTE that can return the upward recommendation chain for any member. You should be able to select recommender from recommenders where member=x. Demonstrate it by getting the chains for members 12 and 22.
Results table should have member and recommender, ordered by member ascending, recommender descending.
Schema reminder
cd.members
memid integer
surname character varying(200)
firstname character varying(200)
address character varying(300)
zipcode integer
telephone character varying(20)
recommendedby integer
joindate timestamp
cd.bookings
facid integer
memid integer
starttime timestamp
slots integer
cd.facilities
facid integer
name character varying(100)
membercost numeric
guestcost numeric
initialoutlay numeric
monthlymaintenance numeric
Expected Results
member recommender firstname surname
12 9 Ponder Stibbons
12 6 Burton Tracy
22 16 Timothy Baker
22 13 Jemima Farrell
Your Answer
Hint
Help
Save
Run Query
with recursive recommenders(recommender, member) as (
select recommendedby, memid
from cd.members
union all
select mems.recommendedby, recs.member
from recommenders recs
inner join cd.members mems
on mems.memid = recs.recommender
)
select recs.member member, recs.recommender, mems.firstname, mems.surname
from recommenders recs
inner join cd.members mems
on recs.recommender = mems.memid
where recs.member = 22 or recs.member = 12
order by recs.member asc, recs.recommender desc
Activate Windows
Go to PC settings to activate Windows.
12:54 PM
12/6/2021