

Answers

Union, Intersection, Set Difference, Outer Joins

Handout: union intersection set difference outer joins 2.13.20.pdf

(markdown converted at <https://md2pdf.netlify.com/>)

Professor Database

Prof (**pid**, lname, fname, rank, research)

Dept (**did**, name, address, *chair*)

Appt (*pid*, *did*, percentage)

Proj (**prid**, sponsor, budget, *pid*)

CoPI (*prid*, *pid*)

Grad (**gid**, lname, fname, *major*, advisor)

RA (*gid*, *prid*, *pid*)

Table names are **bold**. **Primary keys** are **bold**. *Foreign keys* are shown in *italics* and match the same field in another table unless specified.

Notes:

1. *chair* in **Dept** refers to **pid** in **Prof**
2. *pid* in **Proj** means PI on Proj
3. *major* in **Grad** refers to **did** in **Dept**

4. *advisor* in **Grad** refers to **gid** in **Grad**

Questions

1. Union. List professors who are PI or coPI on an NSF grant (prid, pid, lname, label 'PI' or 'CoPI').

```
select prid, pid, lname, "PI"
from Professor join Project using(pid)
where sponsor="NSF"
union
select prid, p.pid, lname, "coPI"
from CoPI join Professor p using(pid)
join Project using(prid)
where sponsor="NSF"
order by lname desc
```

2. Set Difference. List professors who are not PI on a grant (pid, lname, fname).

```
select p.pid, lname, fname
from Professor p left join Project j on p.pid=j.pid
where j.pid is null
```

3. Set Difference. List professors who are not PI on a grant from the Wellcome Trust (pid, lname, fname).

```
#using a subquery in the where clause
select pid, lname, fname
from professor
where pid not in (select pid
                  from project)
```

```
where sponsor = "Wellcome Trust")
```

#using a left join and a derived table

```
select p.pid, lname, fname
from professor p left join (select *
                           from Project
                           where sponsor = "Wellcome Trust") as proj2
on p.pid=proj2.pid
where proj2.pid is null
```

#doesn't work because looks row by row

#includes Kirill in the output because he's got some rows

#with and without Wellcome Trust

```
select p.pid, lname, fname
from professor p left join project j on p.pid=j.pid
and sponsor <> "Wellcome Trust"
```

4. Intersection. List professors who are PI on both 'NIH' and 'Kidney Foundation' grants (pid, lname, fname).

#Reva's solution

#uses two copies of project table

```
select pid, lname, fname
from professor join project j using (pid) join project k using (pid)
where j.sponsor="NIH" and k.sponsor="Kidney Foundation"
```

#uses a subquery in the where clause

```
select pid, lname, fname
from professor join project using(pid)
where sponsor = "NIH" and pid in (select pid
```

```
from project
where sponsor = "Kidney Foundation")
```

```
#Conor's solution
#similar to the previous solution, but more complicated
select nihpi.pid, nihpi.lname, nihpi.fname
from
(select pid, lname, fname
from professor join project using(pid)
where sponsor = "NIH") as nihpi
join
(select pid, lname, fname
from professor join project using(pid)
where sponsor = "Kidney Foundation") as kfpi
using(pid)
```

5. List students who are RA for a project with either Gary Benson or Daniel Segre (gid, lname).

```
#Gary's solution
#grad table only has fname
select gid, grad.fname, lname, prid
from RA join grad using(gid) join professor using (pid)
where lname in ("Benson", "Segre")

#similar to previous solution
select gid, grad.fname, lname
from RA r join professor a using(pid) join grad using(gid)
where a.lname = "Benson" or a.lname = "Segre"

#another way
select gid, fname
```

```
from grad join RA using(gid)
where pid in (select pid from professor where lname = "Benson" or lname = "Segre")
```

#works, but overly complicated

```
select student.gid, student.fname
from RA ra1 join RA ra2 using (gid) join grad student using(gid)
where
    ra1.pid in (select pid from professor where lname like "Benson")
or
    ra2.pid in (select pid from professor where lname like "Segre")
```

6. List students who are both RAs and Advisors (gid, lname).

```
#uses subquery in where clause
select gid, fname
from grad join RA using(gid)
where grad.gid in (select advisor from grad)
```

#incorrect because returns student who has an advisor and the advisor is an RA

```
select grad.gid, RA.gid, fname
from grad join RA on advisor=RA.gid
```

#incorrect because returns student who is an RA and has an advisor

```
select gid, fname
from grad
where advisor is not null
and gid in (select gid from RA)
```

7. List all projects that each professor is involved with, whether as PI or coPI in order by professor name (pid, lname, prid, role (1=PI, 2=coPI)). Order by lname ascending and then role descending.

8. For each project, list all the students who are involved with that project as RAs, and their faculty advisor on that project (prid, gid, student lname, pid, professor lname). Order by project id ascending, then student lname ascending. Include projects without RAs.

9. List all projects that are sponsored by the NSF or NIH, but don't have CoPIs (prid, sponsor).