

Q1. PG accommodation

Assumption : one flat is owned by one owner, one flat can be used by many customers

Flats(flatno,bldgname,rooms,ownerid)

Customer(cno,cname,address,flatno)

Owner(ownerid,aname,mobile)

1. List all customers along with flatno and building name

Old syntax

```
Select cno,cname,address,c.flatno,f.flatnobldgname
From customer c, flats f
Where c.flatno=f.flatno
```

New syntax

```
Select cno,cname,address,flatno,bldgname
From customer c inner join flats f
On c.flatno=f.flatno
```

2. List all flats which are vacant

Select *

From flats f

Where not exists (select * from customer c where c.flatno=f.flatno)

Not suggestable to use outer join better to use nested query.

Select *

From flat f left join customer c on c.flatno=f.flatno

Where c.cname is null

3. List the flat details along with owner names

Old syntax

Select *

From owner o , flats f

where o.ownerid = f.ownerid

New syntax

Select *

From owner o inner join flats f

On o.ownerid = f.ownerid

4. Display flat details, owner details and customer details for all customers who has taken pg accommodation

Old syntax

```
Select c.cno,c.cname,c.address,c.flatno,f.bldgname,o.ownerid,o.aname
```

```
From customer c, flats f,owner o
```

```
Where c.flatno=f.flatno and o.ownerid=f.ownerid
```

New syntax

```
Select c.cno,c.cname,c.address,c.flatno,f.bldgname,o.ownerid,o.ename  
From customer c inner join flats f on c.flatno=f.flatno  
Inner join owner o on o.ownerid=f.ownerid
```

5. List all customers who has not yet booked pg accommodation
Select *
From customer c
Where flatno is null;

Q2. Faculty course example

Assumption : one faculty can conduct many courses,

one course can be assigned to many faculties

Faculty (fid, fname,address)

Course(cid, cname,duration days)

Course-faculty(cid,fid,date_of_assignment)

1. Find all faculties who have courses java assigned in may
Select *
From faculty
Where fid in (Select fid
From course_faculty
Where cid=(Select cid
From course
Where cname='JAVA') and month(date_of_assignment)=5)
2. List all faculties for whom no course is assigned
Select *
From faculty f
Where not exists (select * from course_faculty cf where cf.fid=f.fid)
3. List all courses for which no faculty is assigned
Select *
From course c
Where not exists (select *
From course_faculty cf

Where c.cid=cf.cid)

4. List all courses and faculty details who stays in either pune or Mumbai

Old syntax

```
Select c.cname,c.cid,f.fid,f.fname
From course c,faculty f,course_faculty cf
Where c.cid=cf.cid and f.fid=cf.fid and f.address in ('PUNE','MUMBAI')
```

New syntax

```
Select c.cname,c.cid,f.fid,f.fname
From course c inner join course_faculty cf on c.cid=cf.cid
Inner join faculty f on f.fid=cf.fid
where f.address in ('PUNE','MUMBAI')
```

5. List all course details, faculty details for courses with duration > 30 days

Old syntax

```
Select c.cname,c.cid,f.fid,f.fname,c.duration
From course c,faculty f,course_faculty cf
Where c.cid=cf.cid and f.fid=cf.fid and c.duration >30
```

New syntax

```
Select c.cname,c.cid,f.fid,f.fname, c.duration
From course c inner join course_faculty cf on c.cid=cf.cid
Inner join faculty f on f.fid=cf.fid
where c.duration >30
```

6. List all faculties for whom no course is assigned also display faculty with courses assigned

```
select f.fid,f.name,cf.cid,cf.date_assigned
from faculty f left join course_faculty cf on f.fid=cf.fid
```

7. List all faculties for whom no course is assigned also display faculty with courses assigned also display courses which has faculty assigned and also display course for which no faculty is assigned

```
select f.fid,f.name,cf.cid,cf.date_assigned,null courseid,null cname
from faculty f left join course_facuty cf on f.fid=cf.fid
union
select null,null,cf.cid,cf.date_assigned, c.cid,c.cname
from course c left join course_facuty cf on c.cid=cf.cid
```