

#1A

```
select prosjektnr, prosjektnavn
from PROSJEKT
inner join FIRMA on PROSJEKT.firmanr = FIRMA.firmanr
where firmanavn like '%Sen & sur%' and statuskode like 'A';
```

#1B

```
select AKTIVITET.prosjektnr, prosjektnavn, aktivitetnr, aktivitetnavn, startdato
from AKTIVITET
inner join PROSJEKT on AKTIVITET.prosjektnr = PROSJEKT.prosjektnr
where statuskode like 'F' or statuskode like 'T' and avtaltbeløp > 900000
order by AKTIVITET.prosjektnr, aktivitetnr;
```

#1C ALTERNATIVE 1

```
select *
from PROSJEKT
where not exists (select prosjektnr
                  from AKTIVITET
                  where PROSJEKT.prosjektnr = AKTIVITET.prosjektnr);
```

#1C ALTERNATIVE 2

```
select PROSJEKT.*
from PROSJEKT
left join AKTIVITET on PROSJEKT.prosjektnr = AKTIVITET.prosjektnr
where AKTIVITET.prosjektnr is null;
```

#1C ALTERNATIVE 3

```
select *
from PROSJEKT
where PROSJEKT.prosjektnr not in (select prosjektnr
                                   from AKTIVITET);
```

#1C ALTERNATIVE 4

```
select *
from PROSJEKT
where (select count(*)
       from AKTIVITET
       where AKTIVITET.prosjektnr = PROSJEKT.prosjektnr) < 1;
```

#1D

```
select AKTIVITET.prosjektnr, prosjektnavn, sum(timer) as totaltAntallTimer
from AKTIVITET
inner join PROSJEKT on AKTIVITET.prosjektnr = PROSJEKT.prosjektnr
group by PROSJEKT.prosjektnr
having totaltAntallTimer > 1000;
```

#1E

```
select AKTIVITET.prosjektnr, prosjektnavn, min(startdato)
FROM AKTIVITET
inner join PROSJEKT on AKTIVITET.prosjektnr = PROSJEKT.prosjektnr
group by PROSJEKT.prosjektnr;
```

#1F.2

```
select PROSJEKT.prosjektnr, prosjektnavn, sum(avtaltbeløp)/sum(AKTIVITET.timer) as timeslønn
FROM PROSJEKT
inner join AKTIVITET on PROSJEKT.prosjektnr = AKTIVITET.prosjektnr
group by PROSJEKT.prosjektnr;
```

#1H insert into PROSJEKT values (992121, 'Skifte elektrisk anlegg', null, 911, 'T', 2);

#2A 2nf siden den ikke kan være 3nf pgs firma trenger egen tabell.

#2B.1: 1nf siden den bare er avhengig av en del av pk som er prosjektnr.

#2B.2: bcnf siden alle kolonnene blir bestemt av hele pk (composite key) og ingen av kolonnene kan determinere en del av nøkkelen.

#3A.

```
create table KODESTATUS (  
    statuskode char(1) not null,  
    beskrivelse varchar(45),  
    primary key (statuskode)  
);
```

#3A

```
create table FIRMA (  
    firmanr bigint not null,  
    firmanavn varchar(45),  
    primary key (firmanr)  
);
```

#3A

```
create table ANSATTE (  
    id int not null,  
    navn varchar(45),  
    primary key (id)  
);
```

#3A

```
create table PROSJEKT (  
    prosjektnr int not null,  
    prosjektnavn varchar(45),  
    avtaltebeløp bigint,  
    firmanr bigint,  
    statuskode char(1),  
    prosjektlederid int,  
    primary key (prosjektnr),  
    foreign key (firmanr) references FIRMA(firmanr) on delete restrict on update cascade,  
    foreign key (statuskode) references KODESTATUS(statuskode) on delete restrict on update cascade,  
    foreign key (prosjektlederid) references ANSATTE(id) on delete restrict on update cascade  
);
```

#1G

```
create table AKTIVITET (  
    prosjektnr int not null,  
    aktivitetnr int not null,  
    aktivitetnavn varchar(45),  
    startdato date,  
    sluttdato date,  
    virkeligsluttdato date,  
    timer float,  
    primary key (prosjektnr, aktivitetnr),  
    foreign key (prosjektnr) references PROSJEKT(prosjektnr) on delete restrict on update cascade  
);
```

#3B

```
create table ROLLEKODER (  
    rollekode char(1) not null,  
    beskrivelse varchar(45),  
    primary key (rollekode)  
);
```

#3B

```
create table ANSATTROLLER (  
    prosjektnr int not null,  
    aktivitetnr int not null,  
    ansattid int not null,  
    rollekode char(1),  
    primary key (prosjektnr, aktivitetnr, ansattid),  
    foreign key (prosjektnr, aktivitetnr) references AKTIVITET(prosjektnr, aktivitetnr) on delete restrict on  
update cascade,  
    foreign key (ansattid) references ANSATTE(id) on delete restrict on update cascade,  
    foreign key (rollekode) references ROLLEKODER(rollekode) on delete restrict on update cascade  
);
```

#3B

```
create table ARBEIDSTIMER (  
    prosjektnr int not null,  
    aktivitetnr int not null,  
    ansattid int not null,  
    dato date not null,  
    timer float,  
    primary key (prosjektnr, aktivitetnr, ansattid, dato),  
    foreign key (prosjektnr, aktivitetnr, ansattid) references ANSATTROLLER(prosjektnr, aktivitetnr,  
ansattid) on delete restrict on update cascade  
);
```

#3B

```
create table MILEPÆLOVERSIKT (  
    aktivitetnavn varchar(45) not null,  
    milepælkode varchar(2) not null,  
    milepæl varchar(45),  
    primary key (aktivitetnavn, milepælkode),  
    foreign key (aktivitetnavn) references AKTIVITET(aktivitetnavn) on delete restrict on update cascade  
);
```

#3B

```
create table MILPÆLER (  
    prosjektnr int not null,  
    aktivitetnr int not null,  
    aktivitetnavn varchar(45) not null,  
    milepælkode varchar(45) not null,  
    primary key (prosjektnr, aktivitetnr, aktivitetnavn, milepælkode),  
    foreign key (prosjektnr, aktivitetnr, aktivitetnavn) references AKTIVITET(prosjektnr, aktivitetnr,  
aktivitetnavn),  
    foreign key (milepælkode) references MILEPÆLOVERSIKT(milepælkode) on delete restrict on update  
cascade  
);
```

#4

Visninger/utsnitt (views på engels) er når man lager en virtuell tabell som inneholder data fra en annen tabell ved hjelp av en spørring, der resultatet av spørringen blir satt i den virtuelle tabellen.

Man velger selv hvilken del av tabellen som skal i view'et slik at man kan filtrere ut alt som er irrelevant. Brukes blant annet for å koble og forenkle spørringer mot flere tabeller til en. Kan til og med inneholde resultat av aggregatfunksjoner (max, min, avg osv.).

Brukes for å forenkle kompliserte spørringer der man har gjort mye av arbeidet på forhånd når view'et opprettes. Brukes også for å begrense tilgang til data i tabellene på grunn av for eksempel sikkerhetsmessige årsaker. Trenger ikke noe særlig ekstra lagringsplass siden dataene er lagret i tabellene de er koblet mot, men ytelsen blir verre siden det blir en ekstra prosess (original spørring i den virkelige tabellen + kjøring av spørring i den virtuelle tabellen). Noen view'er er read only slik at man ikke kan insert/delete/update data i tabellen.