

```
#Først laver jeg schemaet til databasen
CREATE SCHEMA WorkshopDDL;

#Så indsætter jeg værdierne fra WorkshopDDL.txt
create table kunde (
  kunde_id int not null,
  navn char(50) not null,
  adresse char(50) not null,
  mail char(25) not null,
  tlf char(8) not null,
  primary key (kunde_id)
);

create table bil (
  bil_id int not null,
  registreringsnr char(7) not null,
  model char(20) not null,
  aargang int not null,
  kunde_id int not null,
  primary key(bil_id),
  foreign key (kunde_id) references kunde(kunde_id)
);

create table mekaniker(
  mekaniker_id int not null,
  navn char(40) not null,
  primary key (mekaniker_id)
);

create table reparation(
  reparation_id int not null,
  dato date not null,
  bil_id int not null,
  mekaniker_id int not null,
  primary key (reparation_id),
  foreign key (bil_id) references bil(bil_id),
  foreign key (mekaniker_id) references mekaniker(mekaniker_id)
);

create table reservedel(
  reservedel_id int not null,
  navn char(40) not null,
  pris int not null,
  primary key (reservedel_id)
);

create table resrep(
  resrep_id int not null,
```

```
reservedel_id int not null,  
reparation_id int not null,  
primary key (resrep_id),  
foreign key (reservedel_id) references reservedel(reservedel_id),  
foreign key (reparation_id) references reparation(reparation_id)  
);
```

insert into kunde values

```
(1,'Søren Frederiksen','Vindingevej 19 5230 Odense M','sf@hotmail.com','66109131'),  
(2,'Henrik Hemmingsen','Karetmagerstien 5 5210 Odense NV','hs@gmail.com','68110916'),  
(3,'Asger Johansen','Andedammen 19 5000 Odense C','aj@hotmail.com','75120948'),  
(4,'Rasmus Iversen','Kirstestien 5 5250 Odense SV','ri@gmail.com','86198248');
```

insert into bil values

```
(1,'MA39604','Renault Clio',2006,1),  
(2,'KX78097','Opel Kadett',1998,2),  
(3,'MA45647','Renault 4CV',1961,3),  
(4,'XP37918','VW Up!',2012,4),  
(5,'MH40136','Opel Kadett',2001,3),  
(6,'JS14151','Aston Martin DBS',1956,4);
```

insert into mekaniker values

```
(1,'Palle'),  
(2,'Poul'),  
(3,'Per');
```

insert into reservedel values

```
(1,'Udstoedningsroer',1300),  
(2,'Baathorn',600),  
(3,'Daek',300),  
(4,'Bremser',950),  
(5,'Tandrem',6000),  
(6,'Olie',300);
```

insert into reparation values

```
(1,'2018-10-18',2,1),  
(2,'2019-11-23',3,2),  
(3,'2019-12-06',5,3),  
(4,'2019-12-11',4,2),  
(5,'2020-01-02',3,1),  
(6,'2017-07-10',6,2);
```

insert into resrep values

```
(1,2,2),  
(2,3,5),  
(3,2,5),  
(4,1,3),  
(5,6,4),
```

```
(6,6,6);

#Q1 - List navnene på alle de kunder, som mekaniker Palle har betjent.
SELECT kunde.navn as 'Kundens navn'
FROM kunde, bil, reparation, mekaniker
WHERE kunde.kunde_id = bil.kunde_id AND bil.bil_id = reparation.bil_id AND reparation.mekaniker_id = mekaniker.mekaniker_id AND mekaniker.navn = 'Palle';

#Q2 - List navnene på alle de reservedele, som ikke indgår i mindst en reparation.
SELECT DISTINCT(reservedel.navn) 'Reservedel'
FROM reservedel, resrep
WHERE reservedel.reservedel_id NOT IN (SELECT resrep.reservedel_id FROM resrep);

#Q3 - List navnene på de kunder, som har mere end en bil.
SELECT kunde.navn as 'Kunde navn'
FROM kunde
WHERE (SELECT COUNT(*) FROM bil WHERE kunde.kunde_id = bil.kunde_id) > 1;

#Q4 - Lav en liste med registreringsnummer og model over de biler, som blev repareret i 2019.
SELECT bil.registreringsnr as 'Bilens reg.nr.', bil.model as 'Bilens model'
FROM bil, reparation
WHERE bil.bil_id = reparation.reparation_id AND YEAR(reparation.dato) = 2019;

#Q5 - Skriv et SQL-udtryk som kan forhøje prisen på samtlige reservedele med 7 %,
SET SQL_SAFE_UPDATES = 0;
UPDATE reservedel
SET pris = pris * 1.07;

#Q6 - Skriv en forespørgsel, som viser, at en model (i dette tilfælde "Opel Kadett") forekommer mere end én gang i attributten model i bil-tabellen.
SELECT bil.model as 'Bilmodel'
FROM bil
GROUP BY bil.model
HAVING COUNT(*) > 1;

#Q7 - Én af mekanikerne er veteranbilsekspert. Kan du regne ud hvilken? HINT: Ham der har arbejdet på flest biler der er ældre end 25 år
SELECT mekaniker.navn as 'Veteranbilsekspert'
FROM mekaniker, reparation, bil
WHERE mekaniker.mekaniker_id = reparation.mekaniker_id
AND reparation.bil_id = bil.bil_id
AND YEAR(CURDATE()) - bil.aargang > 25
HAVING COUNT(*) > 1;

#Q8 - List alle biler, baseret på gennemsnitlig pris pr. reparation.
SELECT DISTINCT bil.model as 'Bilmodel', bil.registreringsnr 'Bilens reg.nr.', SUM(reservedel.pris) / COUNT(reparation.bil_id) as 'Gennemsnitspris pr. reparation'
FROM reparation, resrep, bil, reservedel
```

```
WHERE reservedel.reservedel_id = resrep.reservedel_id
  AND reparation.reparation_id = resrep.resrep_id
  AND bil.bil_id = reparation.bil_id
GROUP BY bil.bil_id
ORDER BY reservedel.pris;
```

#Q9 - Chefen på værkstedet vil gerne give en bonus til den mekaniker, der har tjent værkstedet flest penge. Hvem skal have bonussen?

```
SELECT mekaniker.navn as 'Mekaniker navn', SUM(reservedel.pris) as 'Tjente penge'
FROM mekaniker, reparation, resrep, reservedel
WHERE reparation.mekaniker_id = mekaniker.mekaniker_id
  AND reservedel.reservedel_id = resrep.reservedel_id
  AND reparation.reparation_id = resrep.resrep_id
GROUP BY mekaniker.navn
ORDER BY pris DESC
LIMIT 1;
```

#Q10 - Poul ønsker en liste over hvor meget arbejde de andre mekanikere historisk set

sammenlagt har haft pr. kvartal, men er kun interesseret i det første og sidste kvartal.

```
SELECT mekaniker.navn as 'Mekaniker navn', COUNT(reparation.dato) as "Reparationer i første og sidste kvartal"
FROM mekaniker, reparation
WHERE (MONTH(reparation.dato) < 4 OR MONTH(reparation.dato) > 9)
  AND mekaniker.navn != 'Poul'
  AND reparation.mekaniker_id = mekaniker.mekaniker_id
GROUP BY mekaniker.navn;
```