

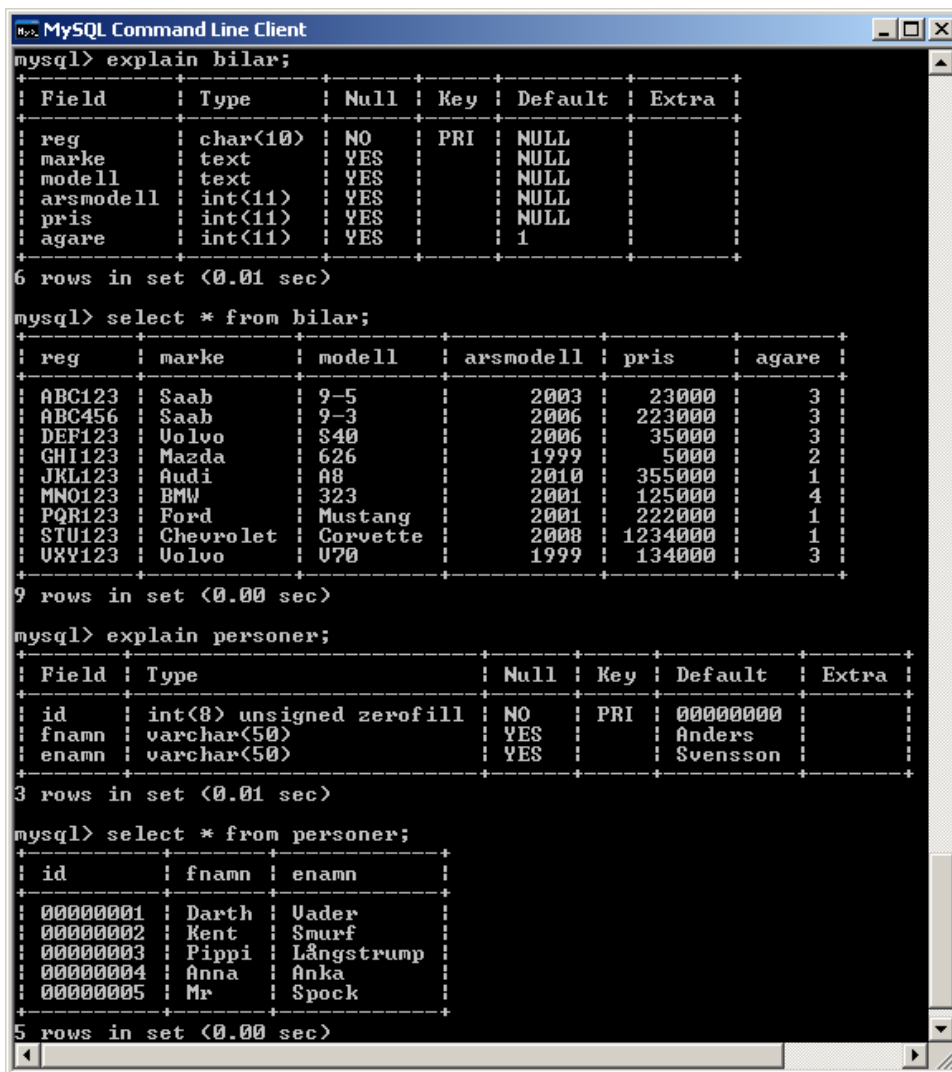
Övning 9 – SQL grunderna IV - SVAR

Övning 9 är det fjärde steget i vår utforskning av SQL med hjälp av MySQL! Som vanligt är det viktigt att du har gjort och förstått de tidigare stegen för att gå vidare. Svaren på alla frågor får du från denna eller tidigare övningar, så ha de tidigare övningarna tillgängliga så du kan kika där vid behov!

I förra övningen lärde du dig att arbeta med två olika tabeller. Vi fortsätter bland annat med detta i denna övning!

Innan du är klar med övningen skall du för mig ha visat upp dina svar på de sista frågorna i övningen!

- ✓ Starta MySQL och ta fram databasen "fordon" från förra övningen! Så här såg de båda tabellerna ut:



```
mysql> explain bilar;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| reg   | char(10) | NO | PRI | NULL |  |
| marke | text    | YES |  | NULL |  |
| modell | text    | YES |  | NULL |  |
| arsmode | int(11) | YES |  | NULL |  |
| pris  | int(11) | YES |  | NULL |  |
| agare | int(11) | YES |  | 1 |  |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.01 sec)

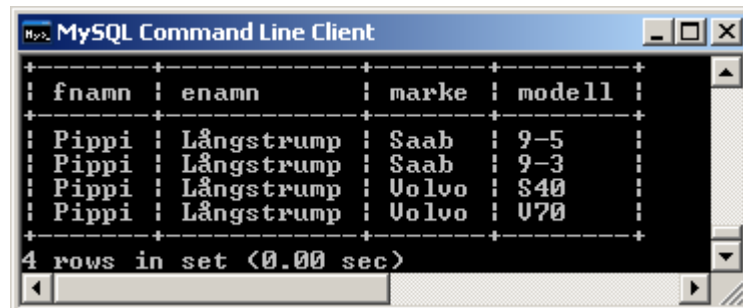
mysql> select * from bilar;
+-----+-----+-----+-----+-----+-----+
| reg   | marke | modell | arsmode | pris  | agare |
+-----+-----+-----+-----+-----+-----+
| ABC123 | Saab  | 9-5    | 2003    | 23000 | 3     |
| ABC456 | Saab  | 9-3    | 2006    | 223000 | 3     |
| DEF123 | Volvo | S40    | 2006    | 35000 | 3     |
| GHI123 | Mazda | 626    | 1999    | 5000  | 2     |
| JKL123 | Audi  | A8      | 2010    | 355000 | 1     |
| MNO123 | BMW   | 323    | 2001    | 125000 | 4     |
| PQR123 | Ford  | Mustang | 2001    | 222000 | 1     |
| STU123 | Chevrolet | Corvette | 2008    | 1234000 | 1     |
| VXY123 | Volvo | U70    | 1999    | 134000 | 3     |
+-----+-----+-----+-----+-----+-----+
9 rows in set (0.00 sec)

mysql> explain personer;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id     | int(8) unsigned zerofill | NO | PRI | 00000000 |  |
| fnamn  | varchar(50) | YES |  | Anders |  |
| enamn  | varchar(50) | YES |  | Svensson |  |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.01 sec)

mysql> select * from personer;
+-----+-----+-----+
| id | fnamn | enamn |
+-----+-----+-----+
| 00000001 | Darth | Vader |
| 00000002 | Kent  | Smurf |
| 00000003 | Pippi | Långstrump |
| 00000004 | Anna  | Anka |
| 00000005 | Mr    | Spock |
+-----+-----+-----+
5 rows in set (0.00 sec)
```

Först lite repetition från tidigare övningar!

1. Lista Pippi Långstrumps alla bilar!

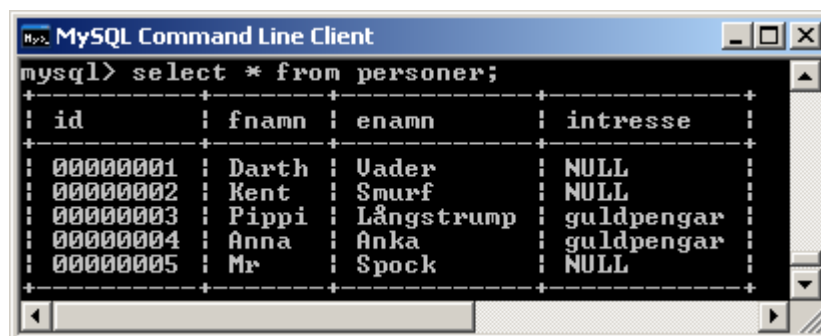


A screenshot of the MySQL Command Line Client window. It displays the result of a query, showing four rows of data. The columns are labeled fnamn, enamn, marke, and modell. The data rows are: Pippi Långstrump Saab 9-5, Pippi Långstrump Saab 9-3, Pippi Långstrump Volvo S40, and Pippi Långstrump Volvo V70. At the bottom, it says '4 rows in set (0.00 sec)'.

fnamn	enamn	marke	modell
Pippi	Långstrump	Saab	9-5
Pippi	Långstrump	Saab	9-3
Pippi	Långstrump	Volvo	S40
Pippi	Långstrump	Volvo	V70

Svar: select personer.fnamn, personer.enamn, bilar.marke, bilar.modell from personer inner join bilar on personer.id=bilar.agare and personer.id=3;

2. Lägg till ett nytt fält "intresse" i tabellen "personer" och låt Pippi Långstrump och Anna Anka få intresset 'guldpengar'!



A screenshot of the MySQL Command Line Client window. It displays the result of a query 'select * from personer;'. The table has four columns: id, fnamn, enamn, and intresse. The data rows are: 00000001 Darth Vader NULL, 00000002 Kent Smurf NULL, 00000003 Pippi Långstrump guldpengar, 00000004 Anna Anka guldpengar, and 00000005 Mr Spock NULL.

id	fnamn	enamn	intresse
00000001	Darth	Vader	NULL
00000002	Kent	Smurf	NULL
00000003	Pippi	Långstrump	guldpengar
00000004	Anna	Anka	guldpengar
00000005	Mr	Spock	NULL

Svar:

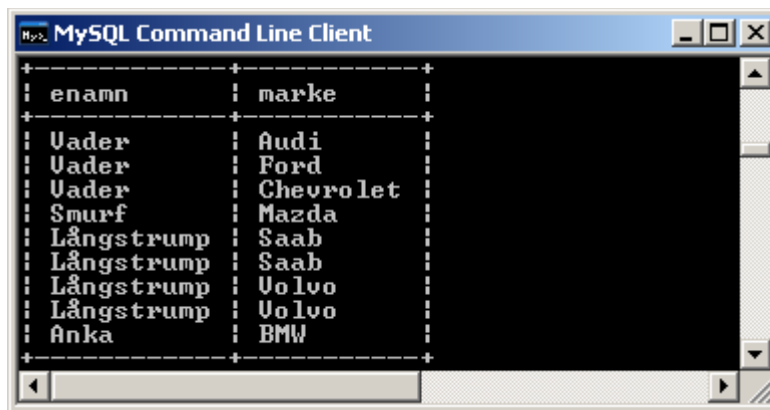
alter table personer add intresse text;

update personer set intresse='guldpengar' where id=3 or id=4;

Låt oss nu fortsätta undersöka frågor som rör flera tabeller!

Tidigare sa vi att man kan använda "inner join" när man vill ange vilka poster som skulle skrivas ut:

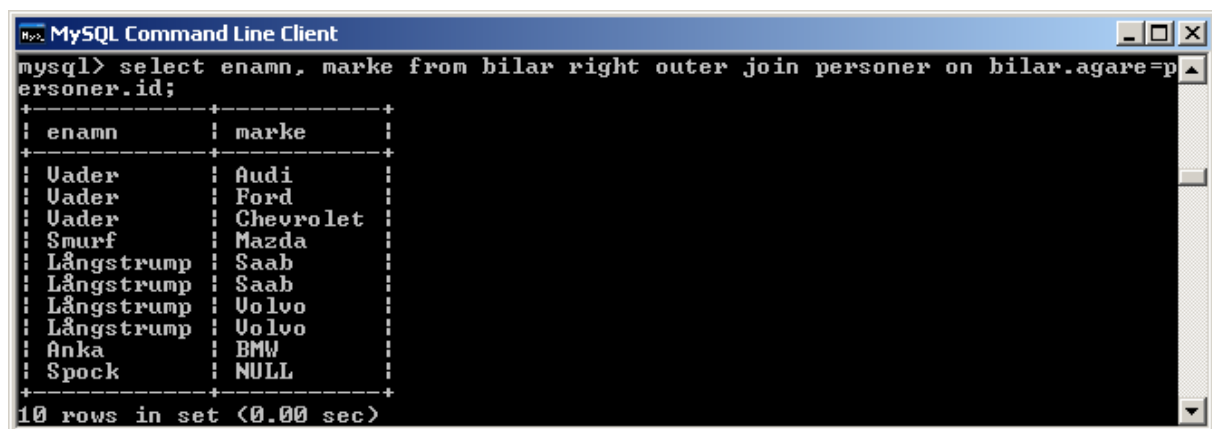
- ✓ Skriv "select personer.enamn, bilar.marke from bilar inner join personer on bilar.agare=personer.id;"



enamn	marke
Vader	Audi
Vader	Ford
Vader	Chevrolet
Smurf	Mazda
Långstrump	Saab
Långstrump	Saab
Långstrump	Volvo
Långstrump	Volvo
Anka	BMW

Eftersom Mr Spock inte har någon bil så kommer den stackarn inte med i resultatet – bara de som har bil kommer med! Men om man nu vill visa även personer som INTE har någon bil, vad gör man då? Jo, då finns det något som heter "outer join", vilken finns i två varianter: "left outer join" och "right outer join":

- ✓ Skriv "select personer.enamn, bilar.marke from bilar right outer join personer on bilar.agare=personer.id;"



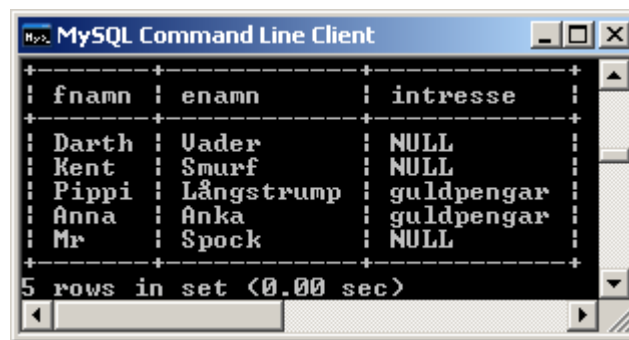
```
mysql> select enamn, marke from bilar right outer join personer on bilar.agare=personer.id;
```

enamn	marke
Vader	Audi
Vader	Ford
Vader	Chevrolet
Smurf	Mazda
Långstrump	Saab
Långstrump	Saab
Långstrump	Volvo
Långstrump	Volvo
Anka	BMW
Spock	NULL

10 rows in set (0.00 sec)

Aha, Mr Spock kom med denna gång! Med "bilar right outer join personer" får man alltså med alla personer i tabellen "personer" (dvs den tabell som är till HÖGER = right om 'right outer join!'), även om villkoret ("bilar.agare=personer.id") inte är uppfyllt! Eftersom Mr Spock inte har någon bil så finns inte Mr Spocks id-nummer någonstans i fältet bilar.agare, och likheten "bilar.agare=personer.id" är då aldrig sann!

3. Lista namnet på alla personers intresse, även om personen inte har något!

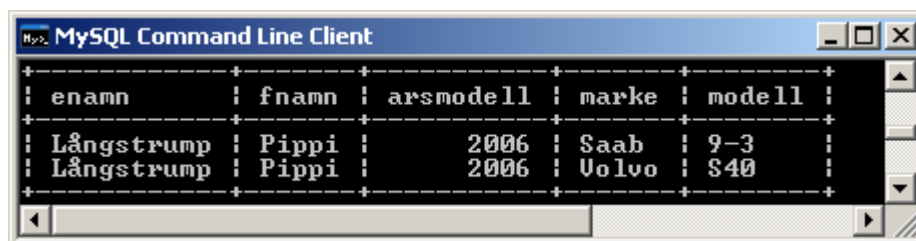


fnamn	enamn	intresse
Darth	Vader	NULL
Kent	Smurf	NULL
Pippi	Långstrump	guldpengar
Anna	Anka	guldpengar
Mr	Spock	NULL

5 rows in set (0.00 sec)

SVAR: select fnamn, enamn, intresse from personer;
(ah, en liten trick question – inte svårare än så!!)

4. När en bil är 3 år gammal börjar bilfirmor skicka reklam till ägarna. Lista namnen, årsmodell, märke och modell på alla ägarna till bilar av årsmodell 2006!

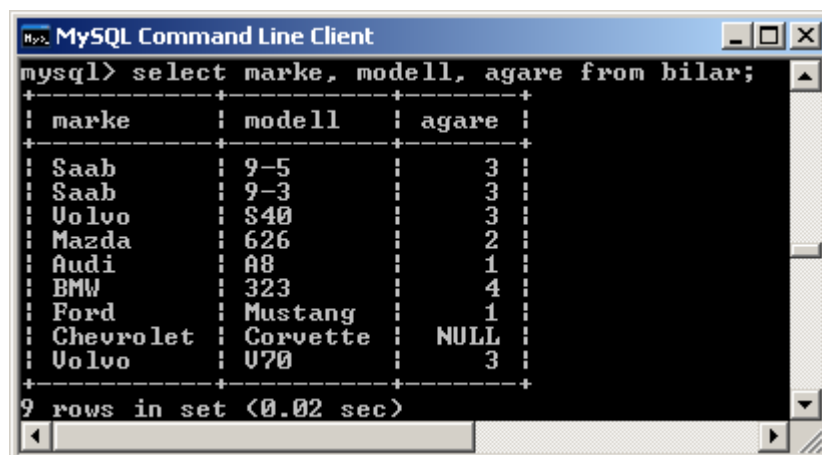


enamn	fnamn	arsmodell	marke	modell
Långstrump	Pippi	2006	Saab	9-3
Långstrump	Pippi	2006	Volvo	S40

Svar:

select personer.enamn, personer.fnamn, bilar.arsmodell, bilar.marke, bilar.modell from bilar inner join personer on bilar.agare=personer.id and bilar.arsmodell=2006;

5. Antag nu att Darth Vader säljer sin Corvette till en bilfirma. Då har bilen ingen ägare (dvs för Corvetten skall ägare anges med null)! Gör detta i din databas!



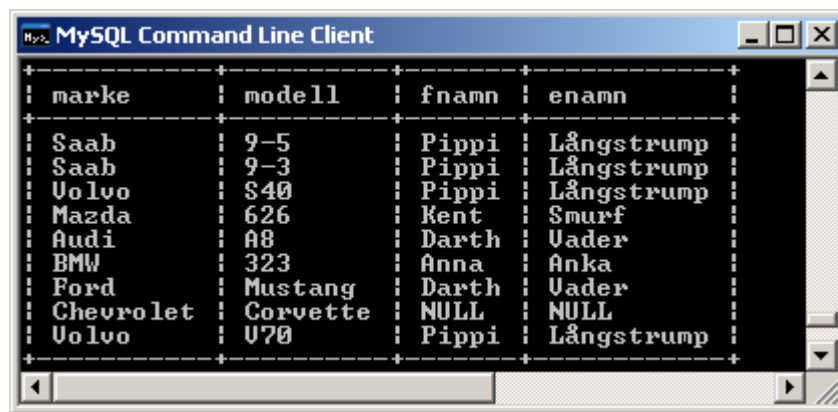
```
mysql> select marke, modell, agare from bilar;
```

marke	modell	agare
Saab	9-5	3
Saab	9-3	3
Volvo	S40	3
Mazda	626	2
Audi	A8	1
BMW	323	4
Ford	Mustang	1
Chevrolet	Corvette	NULL
Volvo	U70	3

9 rows in set (0.02 sec)

SVAR: update bilar set agare=null where modell='corvette';

6. Lista namnet på alla bilar ägare, även för de bilar som inte har någon ägare!

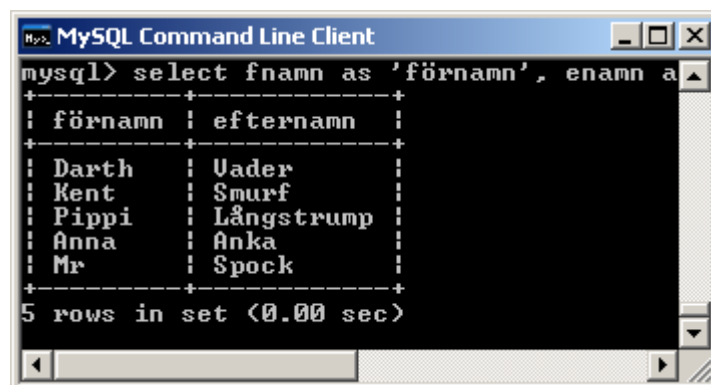


marke	modell	fnamn	enamn
Saab	9-5	Pippi	Långstrump
Saab	9-3	Pippi	Långstrump
Volvo	S40	Pippi	Långstrump
Mazda	626	Kent	Smurf
Audi	A8	Darth	Vader
BMW	323	Anna	Anka
Ford	Mustang	Darth	Vader
Chevrolet	Corvette	NULL	NULL
Volvo	V70	Pippi	Långstrump

SVAR: `select bilar.marke, bilar.modell, personer.enamn, personer.fnamn from bilar left outer join personer on bilar.agare=personer.id;`

Nu ser det kanske lite konstigt ut att ha en rubrik i tabellen ovan som heter "fnamn" (t.ex.). Det kan man ändra på med hjälp av "select ... as ... from ...":

- ✓ Skriv "select fnamn as 'förnamn', enamn as efternamn from personer;".
Fnuittarna runt "förnamn" behövs pga bokstaven ö...



```
mysql> select fnamn as 'förnamn', enamn as 'efternamn' from personer;
```

förnamn	efternamn
Darth	Vader
Kent	Smurf
Pippi	Långstrump
Anna	Anka
Mr	Spock

5 rows in set (0.00 sec)

Då skall vi fortsätta titta lite närmare på funktioner (tidigare har vi tittat på funktioner såsom max(), avg(),...)!

Hittills har vi bara kunna se själva datat för varje "select" vi har använt. Men om vi nu vill använda data t.ex. i en mening (som "Kalle Anka äger en bil med registreringsnummer 313), hur gör man då? Jo, använder funktionen "concat()" så klart! Concat är en förkortning av concatenate som betyder "slå ihop, sätta ihop".

- ✓ Skriv "select concat(marke, "är bäst!") from bilar where marke='volvo';"

```

MySQL Command Line Client
mysql> select concat(marke, " är bäst!") from bilar where marke='volvo';
+-----+
| concat(marke, " är bäst!") |
+-----+
| Volvo är bäst!             |
| Volvo är bäst!             |
+-----+
2 rows in set (0.03 sec)

```

7. Det blir en tråkig fältrubrik, välj en annan!

```

MySQL Command Line Cl...
+-----+
| Bästa märke                 |
+-----+
| Volvo är bäst!             |
| Volvo är bäst!             |
+-----+
2 rows in set (0.00 sec)

```

SVAR: select concat(marke, "är bäst!") as "Bästa märke" from bilar where marke='volvo';

8. Testa concat() på lite olika sätt!

SVAR: experimentera!

Andra roliga funktioner är olika datumfunktioner!

9. Inför ett nytt fält "fodelsedatum" i tabellen "personer". Datatypen för det nya fältet skall vara "date"!

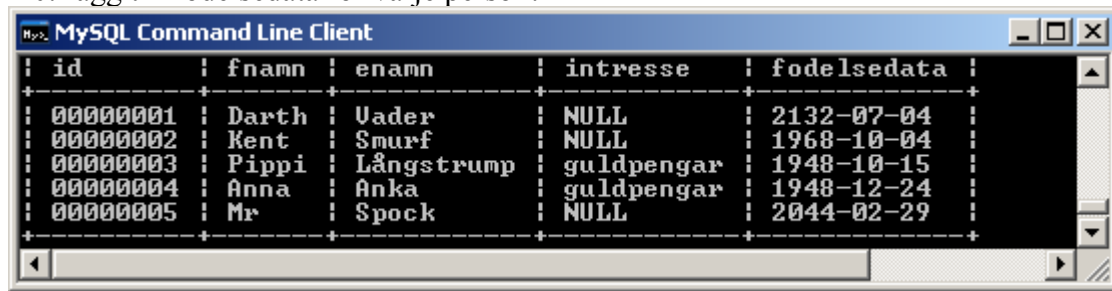
```

MySQL Command Line Client
mysql> explain personer;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | int(8) unsigned zerofill | NO | PRI | 00000000 |
| fnamn | varchar(50) | YES | | Anders |
| enamn | varchar(50) | YES | | Svensson |
| intresse | text | YES | | NULL |
| fodelsedata | date | YES | | NULL |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.02 sec)

```

SVAR: alter table personer add fodelsedata date;

10. Lägga till födelsedata för varje person!

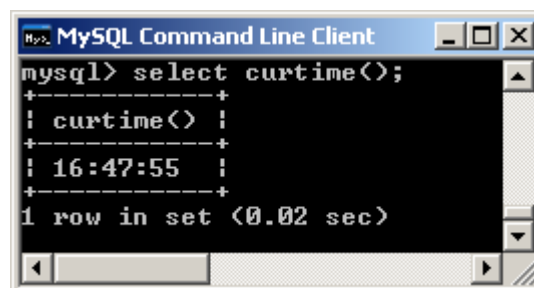


id	fnamn	enamn	intresse	fodelsedata
00000001	Darth	Uader	NULL	2132-07-04
00000002	Kent	Smurf	NULL	1968-10-04
00000003	Pippi	Långstrump	guldpenngar	1948-10-15
00000004	Anna	Anka	guldpenngar	1948-12-24
00000005	Mr	Spock	NULL	2044-02-29

SVAR: update personer set fodelsedata="2132-07-04" where id=1;
OSV.

Man kan nu t.ex. räkna ut hur gammal en viss person är (fast vissa personer i tabellen har inte fötts än...). Men då behöver man även veta dagens datum!

- ✓ Skriv "select curdate()" för att få reda på dagens datum! (curdate står för "current date", dvs "aktuell tid")
- ✓ Skriv "select curtime()" för att få reda på dagens tid!

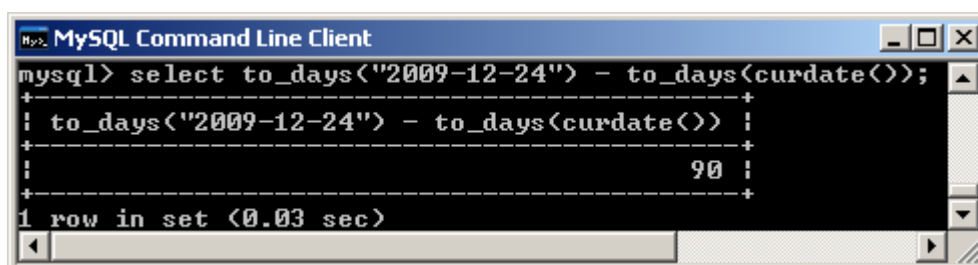


curtime()
16:47:55

1 row in set (0.02 sec)

Vill man veta hur långt det är mellan två datum så kan man lämpligtvis först göra om datumen till dagar genom att använda funktionen "to_days()" och lite matematik:

- ✓ Skriv "select to_days("2009-12-24") - to_days(curdate());" för att ta reda på hur långt det är till julaften!



to_days("2009-12-24") - to_days(curdate())
90

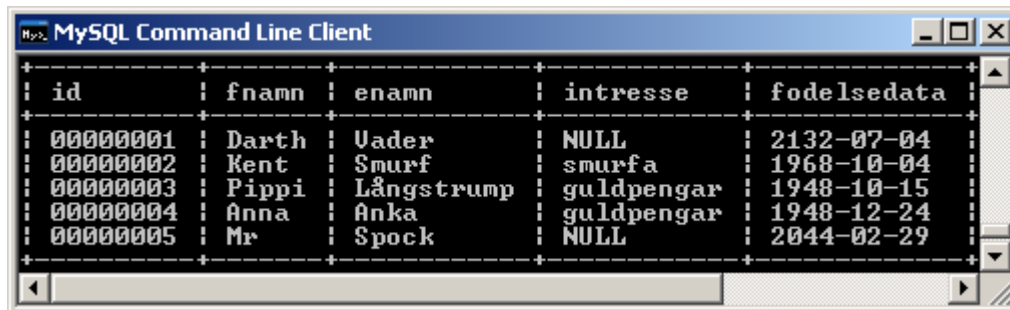
1 row in set (0.03 sec)

Man kan också bara titta på år, månader eller dagar var för sig via funktionerna "year()", "month()" och "dayofmonth()":

- ✓ Skriv "select dayofmonth("2009-12-24");" för att ta ut att det är den 24 i månaden!
- ✓ Testa även de andra funktionerna "month()" och "year()" på samma sätt!

GÖR FÖLJANDE UPPGIFTER OCH VISA UPP FÖR MIG:

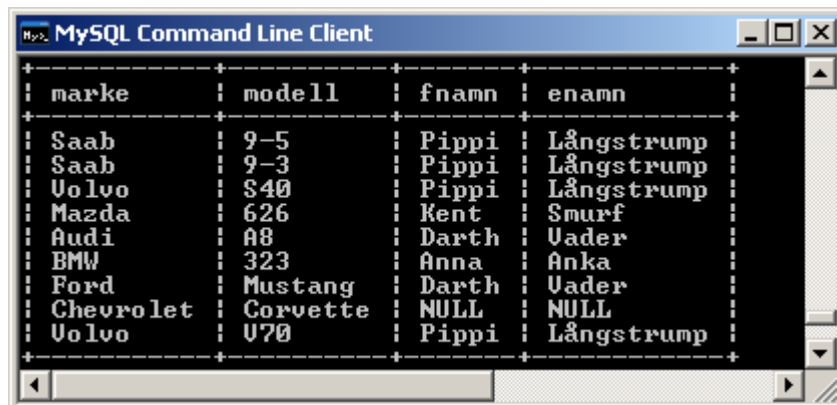
Uppgift 1: Sätt in att Kent Smurf har intresset "smurfa" i din databas!



id	fnamn	enamn	intresse	fodelsedata
00000001	Darth	Vader	NULL	2132-07-04
00000002	Kent	Smurf	smurfa	1968-10-04
00000003	Pippi	Långstrump	guldpengar	1948-10-15
00000004	Anna	Anka	guldpengar	1948-12-24
00000005	Mr	Spock	NULL	2044-02-29

SVAR: update personer set intresse='smurfa' where id=2;

Uppgift 2: Lista namnet på alla bilar ägare, även för de bilar som inte har någon ägare!



marke	modell	fnamn	enamn
Saab	9-5	Pippi	Långstrump
Saab	9-3	Pippi	Långstrump
Volvo	S40	Pippi	Långstrump
Mazda	626	Kent	Smurf
Audi	A8	Darth	Vader
BMW	323	Anna	Anka
Ford	Mustang	Darth	Vader
Chevrolet	Corvette	NULL	NULL
Volvo	V70	Pippi	Långstrump

SVAR: select bilar.marke, bilar.modell, personer.enamn, personer.fnamn from bilar left outer join personer on bilar.agare=personer.id; (fråga 6!)

Uppgift 3: Räkna ut hur många dagar gammal du är!

SVAR: select to_days("2009-09-25") - to_days("1964-06-04") ;

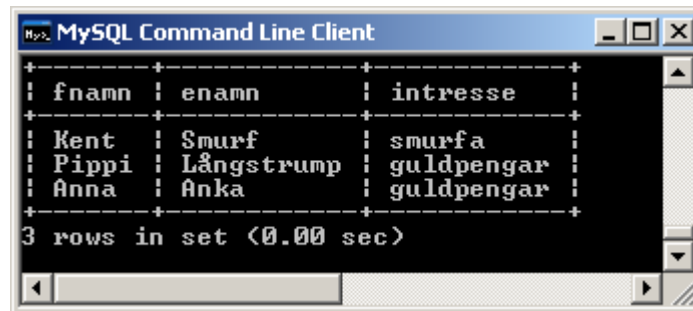
Uppgift 3: Lista namnen på alla personer som äger bilar men inte har något intresse!

Uppgift 4: Skapa nu en fråga

Uppgift 5: Skapa nu en fråga som listar vilka bilar Mr Spock har!

Extrauppgifter (lite svårare):

Extrauppgift 1: Lista namnen på alla personer som har ett intresse!



fnamn	enamn	intresse
Kent	Smurf	smurfa
Pippi	Långstrump	guldpengar
Anna	Anka	guldpengar

3 rows in set (0.00 sec)

SVAR: select fnamn,enamn, intresse from personer where intresse!='null';
eller
select fnamn,enamn, intresse from personer where intresse<>'null';

Extrauppgift 2: Lista namnen på alla personer som äger bilar och inte har något intresse!
Observera rubrik!

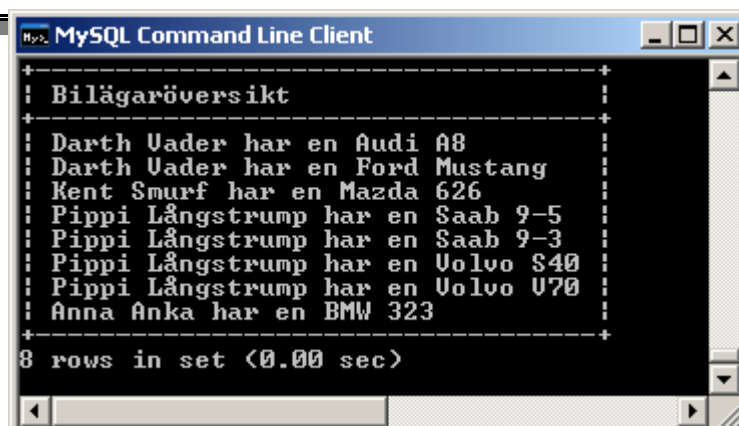


fnamn	enamn	intresse	har bl.a. bilmärket
Anna	Anka	guldpengar	BMW
Pippi	Långstrump	guldpengar	Saab
Kent	Smurf	smurfa	Mazda

3 rows in set (0.00 sec)

SVAR: select personer.fnamn,personer.enamn, personer.intresse, bilar.marke as "har bl.a. bilmärket" from personer, bilar where personer.id=bilar.agare and personer.intresse!='*' group by enamn;

Extrauppgift 3: Vilken SQL-fråga skriver ut följande tabell?

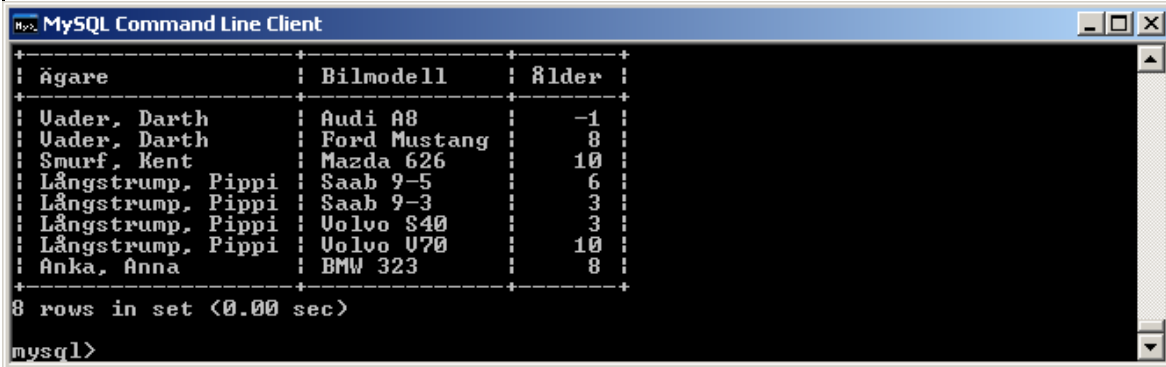


Bilägaröversikt
Darth Vader har en Audi A8
Darth Vader har en Ford Mustang
Kent Smurf har en Mazda 626
Pippi Långstrump har en Saab 9-5
Pippi Långstrump har en Saab 9-3
Pippi Långstrump har en Volvo S40
Pippi Långstrump har en Volvo V70
Anna Anka har en BMW 323

8 rows in set (0.00 sec)

SVAR: select concat(personer.fnamn," ",personer.enamn," har en ",bilar.marke,"
",bilar.modell) as "Bilägaröversikt" from personer inner join bilar on
personer.id=bilar.agare;

Extrauppgift 4: Vilken SQL-fråga skapar följande tabell:



MySQL Command Line Client

Ägare	Bilmodell	Ålder
Uader, Darth	Audi A8	-1
Uader, Darth	Ford Mustang	8
Smurf, Kent	Mazda 626	10
Långstrump, Pippi	Saab 9-5	6
Långstrump, Pippi	Saab 9-3	3
Långstrump, Pippi	Volvo S40	3
Långstrump, Pippi	Volvo V70	10
Anka, Anna	BMW 323	8

8 rows in set (0.00 sec)

mysql>

SVAR: select concat(personer.enamn, " ", personer.fnamn) as "Ägare",
concat(bilar.marke," ", bilar.modell) as "Bilmodell", year(curdate())-
year(concat(bilar.arsmodell,"-01-01")) as "Ålder" from personer inner join bilar on
personer.id=bilar.agare;

Extrauppgift 5: Hitta på egna (svårare!) frågor och försök hitta rätt SQL-fråga!