

Tentamen

Databaser, modellering och implementering 7,5 hp

Distanskurs från Östersund (DSV)

Onsdag 5 juni, 2019, kl. 09.00 – 14.00 (5 timmar)

Antal uppgifter: 3 (samt deluppgifter)

Maxpoäng: 40

Betygskala¹:

F	Fx	E	D	C	B	A
0-39	40-49	50-59	60-69	70-79	80-89	90-100

¹ A-E är Godkänt, Fx och F är underkänt. Enhet som används är %. Betyget beräknas genom att uppgifterna viktas och därefter räknas din poäng på uppgifterna om till en procentsats. Betyget baseras på denna procentsats enligt skalan. Dock underkänns (Fx) hela tentan om en av uppgifterna är underkänd (mindre än halva poängen på uppgiften).

Personnummer: 940219-8130

Ditt namn: Marcus Roos

Anvisningar:

Förutom de regler som specificerats på sidan Tentaregler i Moodle för denna kurs gäller även följande:

- Ta en titt lite nu och då i forumet i Moodle eftersom jag där kommer att skriva eventuella tips/förtydligande som behövs för uppgifterna.
- Tentamen får endast skrivas på den plats du angav i din anmälan.
- Tentamen ska skrivas personligen. Det är inte tillåtet att kopiera svar från någon annan, från webbsidor eller liknanden. Det är inte heller tillåtet att ta hjälp från någon annan.
- Innan skrivningstiden är slut måste du skicka in ditt svar. Detta görs i Moodle på samma sätt och på samma ställe där du tidigare har lämnat in dina inlämningsuppgifter.
- Det som ska lämnas in är detta dokument (eller motsvarande) i vilket du svarat på uppgifterna. Har du använt andra program måste du klippa-och-klistra detta till ditt dokumentet samt bifoga originalfilerna från övriga program du använt. Konvertera sen ditt dokument till pdf.
- Döp din pdf-fil enligt följande format:
ååmmdd-nnnn, Efternamn Förnamn.pdf
- Om det blir några problem med din nätanslutning under tentamenstiden, eller när du ska lämna in ditt svar, måste du omedelbart kontakta mig, Robert Jonsson på telefon ~~010-14-28-138~~ eller 070-698 77 43. Jag kan även nås via forumet i Moodle, samt på robert.jonsson@miun.se.

Uppgift 1: Datamodellering (20p)

Bussbolag

Ett nystartat privat bussbolag planerar att ta upp konkurrensen på tätortstrafiken i Östersund med det kommunalägda bussbolaget (påhittade uppgifter). I samband med detta vill de införa en databas för att hålla koll på alla sina anställda, bussar och de linjer som ska köras. De har nu anlitat dig för att ta fram en modell för databasen enligt de förutsättningar som ges nedan.



Bolaget har ett antal chaufförer som identifieras med ett unikt anställningsnummer. En chaufför har ett namn (består av för- och efternamn), en lön och ett eller flera telefonnummer av olika typ (hem, mobil, arbete etc.).

De bussar bolaget äger identifieras med sitt registreringsnummer. För varje buss ska det även lagras uppgifter om antal sittplatser, årsmodell och ett datum när bussen köptes.

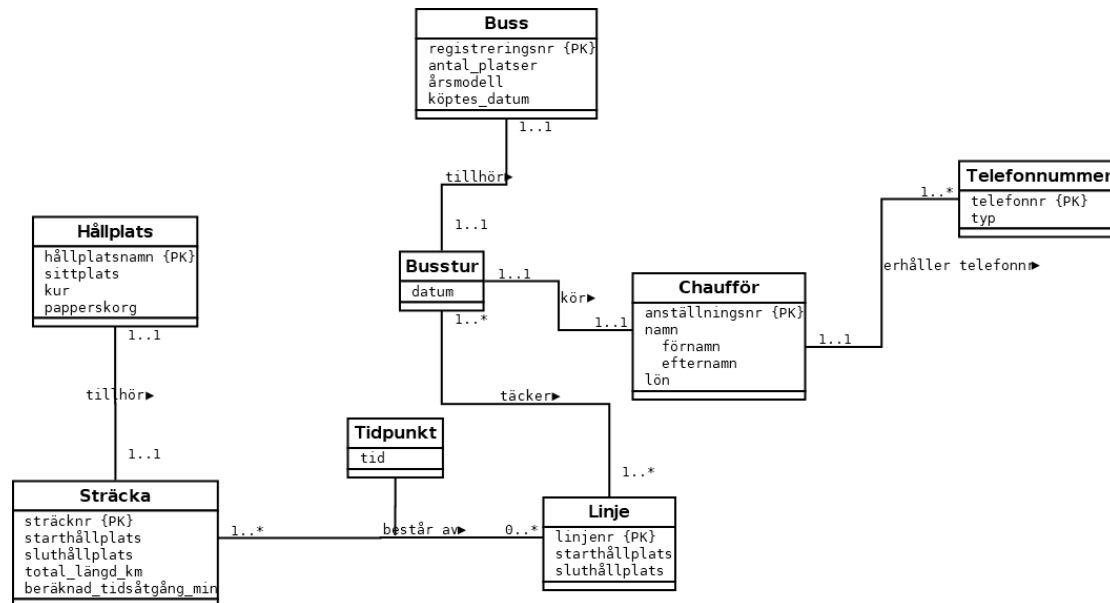
Alla hållplatser ska lagras i databasen. Varje hållplats har ett unikt namn och utöver namnet ska uppgifter om hållplatsen erbjuder sittplatser för väntande passagerare, om hållplatsen har en kur av något slag (för skydd mot regn) och om det finns papperskorg vid hållplatsen.

Bussbolaget planerar att köra ett flertal olika linjer vilka i sin tur består av flera sträckor (minst en per linje). Varje sträcka identifieras av ett unikt nummer och består av en starthållplats och en sluthållplats. För varje sträcka ska det också lagras hur lång sträckan är (km) och hur lång tid det beräknas ta att köra sträckan (minuter).

Varje linje identifieras också med ett unikt nummer och har en start- och sluthållplats. Alla sträckor behöver inte ingå i någon linje, men en och samma sträcka kan ingå i många olika linjer. För varje sträcka, som ingår i en linje, lagras den tidpunkt som sträckan beräknas köras i denna linje (för att kunna avgöra i vilken ordning sträckorna ingår i en linje).

Varje busstur identifieras av samma nummer som den linje turen tillhör. Varje busstur körs av en viss chaufför, ett visst datum, med en viss buss och på en viss linje.

- a) Gör en konceptuell datamodell (ER-modell) utifrån beskrivningen ovan. Redogör kortfattat för de förutsättningar och antaganden du finner nödvändiga att göra.



Jag valde att lägga till en ny tabell Tidpunkt som registrerar en tid och består av primärnycklarna från sträcka och linje. Med busstur gjorde jag likadant och låter dess primärnyckel vara från linje, främmandenycklar är chaufför och buss. På så sätt kan varje busstur kopplas ihop till en viss buss, vilken chaufför som kör den, samt vilken linje bussturen kommer gå mellan (hållplatser). För telefonnummer valde jag att skapa en ny tabell då en chaufför kan ha många olika telefoner och av olika typer, telefonnummer använder anställningsnr som främmande nyckel men även namn hade fungerat lika bra då telefonnr är primärnyckeln.

- b) Översätt den konceptuella datamodellen (ER-modellen), som du kom fram till i uppgift a till relationsmodellen, oavsett vilken notation du använde för att göra modellen i a. Markera alla primär- och främmandenycklar.

Chaufför(anställningsnr, förnamn, efternamn, lön)
Telefonnummer(telefonnr, typ, anställningsnr*)
Buss(registreringsnr, antal_platser, årsmodell, köptes_datum)
Hållplats(hållplatsnamn, sittplats, kur, papperskorg, sträcknr*)
Sträcka(sträcknr, starthållplats, sluthållplats, total_längd_km, beräknad_tidsåtgång_min, linjenr*)
Tidpunkt(linjenr*, sträcknr*, tid)
Linje(linjenr, starthållplats, sluthållplats)
Busstur(linjenr*, datum, anställningsnr*, buss*)

Uppgift 2: SQL (15 p)

Nedanstående relationsmodell är ett förslag till en databas för att lagra uppgifter som rör Eurovision Song Contest. Utgå ifrån denna modell (med primärnycklar understrukna) när du svarar på uppgifterna nedan:

Land (landskod, land)
År (år, datum, stad, landskod)
Roll (roll, beskrivning)
Person (personid, förnamn, efternamn, landskod)
Programledare (personid, år)
Sång (år, startnr, titel, poäng, placering, landskod)
Medverkande (år, startnr, personid, roll)

Förklaring: I År lagras bl.a. uppgifter om vilken stad som stod som värd för ett Schlager-EM. I Roll lagras olika roller som hör till en sång (kompositör, sångare, textförfattare etc). I Sång lagras alla sånger som varit med, hur det gick och vilket land som framförde sången. Startnr anger i vilken startordning sångerna framförs (börjar alltid med 1, men totala antalet sånger kan variera från år till år). I Medverkande lagras data om alla som medverkat med någon roll i en sång.

Uppgifter:

- a) **2 poäng.** I tabellerna ovan är endast primärnycklarna markerade. Din uppgift är att identifiera alla främmandenycklar. Svara enligt:

```
FOREIGN KEY (År.landskod) REFERENCES Land(landskod)
FOREIGN KEY (Person.landskod) REFERENCES Land(landskod)
FOREIGN KEY (Programledare.personid) REFERENCES Person(personid)
FOREIGN KEY (Programledare.år) REFERENCES År(år)
FOREIGN KEY (Sång.år) REFERENCES År(år)
FOREIGN KEY (Sång.landskod) REFERENCES Land(landskod)
FOREIGN KEY (Medverkande.år) REFERENCES År(år)
FOREIGN KEY (Medverkande.startnr) REFERENCES Sång(startnr)
FOREIGN KEY (Medverkande.personid) REFERENCES Person(personid)
FOREIGN KEY (Medverkande.roll) REFERENCES Roll(roll)
```

- b) **2 poäng.** Skriv en SQL-fråga (CREATE TABLE) som skapar tabellen Medverkande. Glöm inte att definiera primär- och främmandenycklar.

```
CREATE TABLE Medverkande (  
    år INT,  
    startnr INT,  
    personid INT,  
    roll VARCHAR(30),  
    CONSTRAINT Medverkande_pk PRIMARY KEY(år, startnr, personid, roll),  
    CONSTRAINT Medverkande_fk1 FOREIGN KEY(år) REFERENCES År(år),  
    CONSTRAINT Medverkande_fk2 FOREIGN KEY(startnr) REFERENCES Sång(startnr),  
    CONSTRAINT Medverkande_fk3 FOREIGN KEY(personid) REFERENCES Person(personid),  
    CONSTRAINT Medverkande_fk4 FOREIGN KEY(roll) REFERENCES Roll(roll));
```

- c) **2 poäng.** Skriv en SQL-fråga som listar år, titel och poäng för alla sånger som vunnit när de startat som nummer 1. Listan ska sorteras efter år.

```
SELECT år, titel, poäng FROM Sång  
WHERE placering LIKE '1'  
AND startnr LIKE '1'  
ORDER BY år
```

- d) **3 poäng.** Skriv en SQL-fråga som listar för- och efternamn på programledarna av den Eurovision Song Contest som hölls i staden Stockholm år 2016. Sortera efter efternamn.

```
SELECT a.förnamn, a.efternamn  
FROM Person a, År b, Programledare c  
WHERE b.år=2016 AND b.stad LIKE 'Stockholm' AND c.personid=a.personid  
ORDER BY a.efternamn ASC;
```

- e) **3 poäng.** Skriv en SQL-fråga som visar alla länder som någon gång har deltagit i tävlingen. Namnet på landet tillsammans med deras bästa, deras sämsta och deras genomsnittliga placering, samt antal gånger de deltagit ska visas. Sortera resultatet så att länder med bäst genomsnittliga placering kommer först.

```
SELECT a.land, ROUND(avg(b.placering),2) AS Genomsnitt,  
MIN(b.placering) AS Sämsta,  
MAX(b.placering) AS Bästa, COUNT(b.landskod) AS Antal  
FROM Land a, Sång b  
WHERE a.landskod = b.landskod  
GROUP BY a.land  
ORDER BY Genomsnitt DESC;
```

- f) **3 poäng.** Skriv en SQL-fråga som listar namnet på de länder som någon gång vunnit två år i rad. (Tips: "nuvarande år" – "föregående år" = 1) Sortera länderna i bokstavsordning.

Uppgift 3: Normalisering (5 p)

Givet är följande tabell Biläggande:

Regnr	Personnr	Bilägare	Skadefria_år	Försäkringsklass	Inköpsdatum	Försälj_datum	Märke
ABC123	651102-0233	Kalle Blom	10	B	1979-01-23	1988-12-12	Volvo
	770425-1332	Stina Grann	5	B	1988-12-12	1991-06-21	
	540801-7743	Göran Bravur	5	B	1991-06-21		
QWE456	770425-1332	Stina Grann	5	B	1991-07-01		Saab
DDE721	710510-3981	Sara Linde	2	A	2000-01-01	2002-03-29	BMW
	620914-6429	Jenny Forss	15	C	1990-11-20	2000-01-01	
UTE122	800712-7656	Roger Lind	2	A	1998-07-12	1998-08-12	Volvo
	591212-8811	Göran Bravur	20	D	1998-08-12		

Följande fakta är knuten till tabellen:

- Primärnyckel i tabellen är registreringsnummer.
- Tabellen visar endast ett fåtal av alla rader som finns inlagd.
- Varje rad i tabellen separeras med en heldragen linje.
- En bil kan endast ha en ägare åt gången.
- Fler antal skadefria år ger en högre (A, B, C, ..., Ö) försäkringsklass.

Uppgifter:

- a) Utifrån de rader med data som visas i tabellen ovan samt de förutsättningar som ges i texten ska du redogöra för de fullständigt funktionella beroenden som existerar. Svaret måste motiveras/ utvecklas.

ffd1: Regnr --> Personnr, Bilägare, Skadefria_år, försäkringsklass, märke, Inköpsdatum, Försälj_datum

ffd2: Personnr --> Bilägare, Skadefria_år, Försäkringsklass

ffd3: Skadefria_år --> Försäkringsklass

ffd4: Personnr, Bilägare --> Skadefria_år, Försäkringsklass

ffd5: Inköpsdatum --> Märke *

ffd6: försälj_datum --> Märke *

(De markerade är jag osäker om jag vill räkna med då endast ett fåtal värden syns, samtliga är unika så ett ffb uppfylls här, men det är enligt mig inte en regel utan det kan mycket väl hända att olika märken köps in vid samma inköpsdatum. Jag låter däremot det här beroendet stå kvar då tabellen vi ser uppfyller beroendet, som sagt, att bilar säljs och köps in av olika märken vid samma datum är inte alls omöjligt.)

b) Vilken är den högsta normalform som tabellen ovan uppfyller. Varför?

Tabellen uppfyller onormaliserad form, vi ser att vid varje registreringsnummer (primärnyckeln) så finns det flera värden per kolumn, det är en lista av värden. Även om varje bil endast kan ha en ägare åt gången så är enligt tabellen flera ägare registrerade till en och samma bil, vi ser i inköpsdatum och försälj_datum när bilen bytte ägare. Den här tabellen behöver delas upp i flera tabeller för att komma bort från onormaliserad form.

c) Om du anser att tabellen inte uppfyller Boyce-Codd-normalform, visa hur tabellen normaliseras till BCNF. Svaret måste motiveras/utvecklas.

Först måste multipla värdena delas upp i separata tabeller.

Regnr	Personnr	Bilägare	Skadefria_år	Försäkringsklass	Inköpsdatum	Försälj_datum	Märke
ABC123	651102-0233	Kalle Blom	10	B	1979-01-23	1988-12-12	Volvo
	770425-1332	Stina Grann	5	B	1988-12-12	1991-06-21	
	540801-7743	Göran Bravur	5	B	1991-06-21		
QWE456	770425-1332	Stina Grann	5	B	1991-07-01		Saab
DDE721	710510-3981	Sara Linde	2	A	2000-01-01	2002-03-29	BMW
	620914-6429	Jenny Forss	15	C	1990-11-20	2000-01-01	
UTE122	800712-7656	Roger Lind	2	A	1998-07-12	1998-08-12	Volvo
	591212-8811	Göran Bravur	20	D	1998-08-12		

Vi ser här att Regnr är primärnyckeln, och de upprepande grupperna är personnr, bilägare, skadefria_år, försäkringsklass, inköpsdatum, försälj_datum samt märke.

Regnr	Personnr	Bilägare	Skadefria_år	Försäkringsklass	Inköpsdatum	Försälj_datum	Märke
ABC123	651102-0233	Kalle Blom	10	B	1979-01-23	1988-12-12	Volvo
ABC123	770425-1332	Stina Grann	5	B	1988-12-12	1991-06-21	Volvo
ABC123	540801-7743	Göran Bravur	5	B	1991-06-21		Volvo
QWE456	770425-1332	Stina Grann	5	B	1991-07-01		Saab
DDE721	710510-3981	Sara Linde	2	A	2000-01-01	2002-03-29	BMW
DDE721	620914-6429	Jenny Forss	15	C	1990-11-20	2000-01-01	BMW
UTE122	800712-7656	Roger Lind	2	A	1998-07-12	1998-08-12	Volvo
UTE122	591212-8811	Göran Bravur	20	D	1998-08-12		Volvo

Tabellen är nu i 1NF.

För att få den till 2NF behöver vi få bort redundansen, det finns flertalet regnr i tabellen. Märke är t.ex. beroende av regnr, likväl som personnummer är beroende av bilägare, eller på samma sätt som bilägare och skadefriaår, försäkringsklass håller ihop, dessa går att separera.

Regnr	Märke
ABC123	Volvo
QWE456	Saab
DDE721	BMW
UTE122	Volvo

Regnr	Personnr	Inköpsdatum	Försälj_datu m
ABC123	651102-0233	1979-01-23	1988-12-12
ABC123	770425-1332	1988-12-12	1991-06-21
ABC123	540801-7743	1991-06-21	
QWE456	770425-1332	1991-07-01	
DDE721	710510-3981	2000-01-01	2002-03-29
DDE721	620914-6429	1990-11-20	2000-01-01
UTE122	800712-7656	1998-07-12	1998-08-12
UTE122	591212-8811	1998-08-12	

Personnr	Bilägare	Skadefria_å r	Försäkrings klass
651102-0233	Kalle Blom	10	B
770425-1332	Stina Grann	5	B
540801-7743	Göran Bravur	5	B
710510-3981	Sara Linde	2	A
620914-6429	Jenny Forss	15	C
800712-7656	Roger Lind	2	A
591212-8811	Göran Bravur	20	D

Här uppfylls 2NF då tabellerna är uppdelade för att minsta redundansen, och samtidigt kopplar samman regnumret med ett märke, med en person och mellan vilka datum de ägde bilen. Personerna är uppdelade i en separat tabell på så sätt kan man få information om de via föregående tabeller, däremot uppfylls in 3NF då redundans fortfarande kan uppstå då man måste uppdatera fler tabeller åt gången. Jag är tyvärr inte helt säker på hur jag ska gå vidare med det nu dock, och jag ser att tiden börjar ta slut för mig, kommer klura på det här efteråt tills det sitter.

Jag uppdaterade lite nedanför, skadefria år går att lägga i en egen tabell, samt dela upp ägarna och åren genom att ta bort försäkringsklass.

Skadefria_å r	Försäkrings klass
10	B
5	B
2	A
15	C
20	D

Personnr	Bilägare	Skadefria_år
651102-0233	Kalle Blom	10
770425-1332	Stina Grann	5
540801-7743	Göran Bravur	5
710510-3981	Sara Linde	2
620914-6429	Jenny Forss	15
800712-7656	Roger Lind	2
591212-8811	Göran Bravur	20