

Tentamen för DD1370

Databasteknik och informationssystem

10 April 2015

Hjälpmedel:

Inga hjälpmedel utom papper och penna

Tänk på:

Skriv *högst* en uppgift på varje blad.

Använd *endast framsidan* på varje blad.

Skriv *namn och personnr* på varje blad.

Uppgifterna kommer inte i svårighetsordning.

Skriv tydligt, motivera svaren – endast begriplig och läsbar lösning ger poäng.

Maximal poäng finns angiven inom parentes vid varje uppgift.

Totalt ger tentamen en poäng (max 65), som sedan läggs ihop med era bonuspoäng.

En summa (tenta+bonus) på 40 ger säkert godkänt.

(lösningsförslag kommer på kurswebben)

Lycka till,

Petter

1. (Totalt: 14p)

- a) (3p) Ange 3 datatyper som kan användas för att ange när något har hänt. För varje datatyp, ange ett exempel på användning där inte de övriga två passar lika bra.

Lösning: Date (datum), Time (klockslag), Timestamp (datum och klockslag). Exempel: Födelsedag (Date), Solförmörkelse (Timestamp), Tid då Konsum stänger på vardagar (Time).

- b) (2p) Ange 2 datatyper som kan användas för att ange ett tal. För varje datatyp, ange ett exempel på användning där inte de övriga två passar lika bra.

Lösning: Integer (heltal), Float (decimaltal) Exempel: Antal kunder (Integer), Resultat i längdhopp (Float).

- c) (3p) Antag att tabellerna A och B är givna enligt nedan.

RegNr	Tillverkare	PNr	Namn
ABC123	BMW	991111-1111	Anna
AAA111	Volvo	881111-1111	Audi
BBB222	Audi		

Rita tabellen man får som resultat av:

```
SELECT * FROM A, B WHERE Tillverkare='Audi'.
```

Lösning: Resultatet blir följande

RegNr	Tillverkare	PNr	Namn
BBB222	Audi	991111-1111	Anna
BBB222	Audi	881111-1111	Audi

- d) (2p) Antag att tabellerna A och B är givna enligt ovan. Rita tabellen man får som resultat av

```
SELECT * FROM A JOIN B ON Tillverkare=Namn.
```

Lösning: Resultatet blir följande

RegNr	Tillverkare	PNr	Namn
BBB222	Audi	881111-1111	Audi

- e) (2p) Antag att tabellerna A och B är givna enligt ovan. Rita tabellen man får som resultat av

```
SELECT * FROM A JOIN B ON Tillverkare='Audi'.
```

Lösning: Resultatet blir följande

RegNr	Tillverkare	PNr	Namn
BBB222	Audi	991111-1111	Anna
BBB222	Audi	881111-1111	Audi

(alltså samma som ovan).

- f) (2p) Antag att tabell C har 1000 rader och 1000 kolumner, samt att tabell D också har 1000 rader och 1000 kolumner. Hur stor blir tabellen man får som resultat av

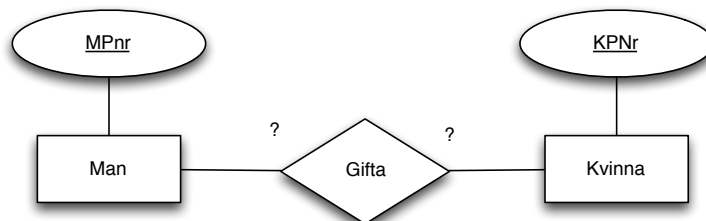
```
SELECT * FROM C, D.
```

Lösning: Resultatet blir 1000000 rader och 2000 kolumner.

2. (Totalt: 16p)

Betrakta ER-modellen i Figurerna 1 till 6. I just denna uppgift behöver ni inte motivera och räkna upp vilka regler ni använder, Det räcker med svaret.

a) (2p) Betrakta figur 1. Vad borde det stå vid frågetecknen? Överför ER-modellen i figuren till en Databasstruktur.



Figur 1: En enkel ER-modell.

Lösning: Vid frågetecknen borde det stå två 1:or, eftersom man bara kan vara gift med en person. Det finns två möjliga databasstrukturer, eftersom 1-1 samband kan bilda ny kolumn på valfri sida av sambandet:

Man(MPnr,KPNr)

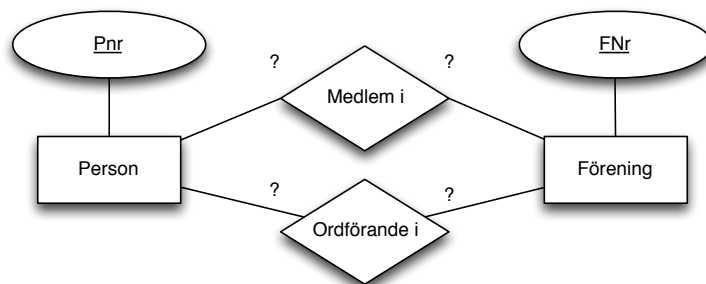
Kvinna(KPNr)

eller

Man(MPnr)

Kvinna(KPNr, MPnr)

b) (2p) Betrakta figur 2. Vad borde det stå vid frågetecknen? Överför ER-modellen i figuren till en Databasstruktur.

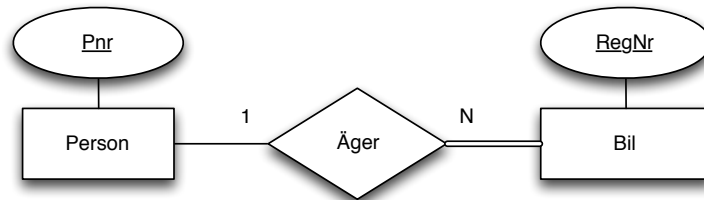


Figur 2: En enkel ER-modell.

Lösning: Vid frågetecknen överst borde det stå N och M, eftersom man kan vara medlem i många föreningar, och en förening kan ha många medlemmar. Vid frågetecknen underst borde det stå 1 och N, eftersom man kan vara ordförande i många föreningar, men varje förening bara har en ordförande. Databasstrukturen blir

Person(Pnr)
Förening(Fnr, Pnr)
Medlem(Pnr,Fnr)

- c) (2p) Betrakta figur 3. Överför ER-modellen i figuren till en Databasstruktur. Vad betyder dubbelsträcket, och vad innebär det för databasen vi skapar utifrån ER-modellen?



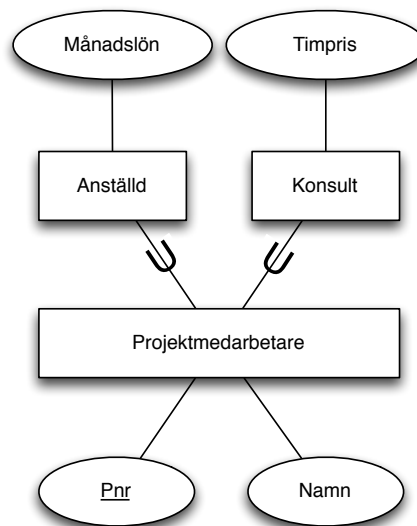
Figur 3: En enkel ER-modell.

Lösning: Databasstrukturen blir

Person(Pnr)
Bil(RegNr, Pnr)

Dubbelsträcket betyder fullständigt deltagande. I databasen måste alltså alla värden i kolumnen Pnr i tabellen Bil vara ifyllda.

- d) (3p) Betrakta figur 4. Överför ER-modellen i figuren till en Databasstruktur.



Figur 4: En enkel ER-modell.

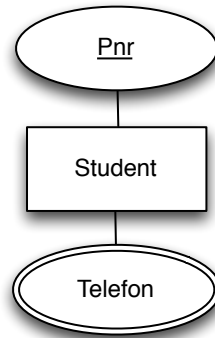
Lösning: Databasstrukturen blir

Projektmedarbetare(Pnr, Namn)

Anställd(Pnr, Månadslön)

Konsult(Pnr, Timpris)

- e) (2p) Betrakta figur 5. Överför ER-modellen i figur till en Databasstruktur.



Figur 5: En enkel ER-modell.

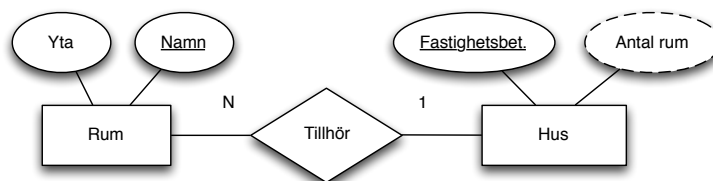
Lösning: Databasstrukturen blir

Student(Pnr)

Telefon(Pnr, TelefonNr)

- f) (5p) Betrakta figur 6.

- Överför ER-modellen i figur till en Databasstruktur.
- Rita den ER-modell som uppstår då man gör om en av entiteterna till en svag entitetstyp.
- Överför den nya ER-modellen till en Databasstruktur. Vad är skillnaden?



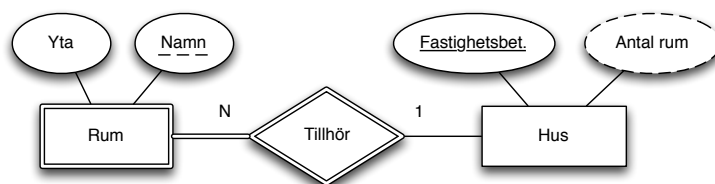
Figur 6: En enkel ER-modell.

Lösning: Den första databasstrukturen blir

Hus(Fastighetsbet.)

Rum(Namn, Yta, Fastighetsbet.)

Antal rum är ett härlett attribut som inte dyker upp databasstrukturen. Om vi gör om Rum till en svag entitetstyp (dubbelstreckad) så blir Namn en partiell nyckel (steckad understykning), Tillhör blir



Figur 7: En enkel ER-modell.

identifierande samband (dubbelstreckat) och deltagandet till vänster fullständigt (dubbelstreckat), se Figur 7.

Databasstrukturen för figur 7 blir

Hus(Fastighetsbet.)

Rum(Fastighetsbet., Namn, Yta)

Antal rum är fortfarande ett härlett attribut som inte dyker upp i databasstrukturen. Skillnaden i databasstruktur är att primärnyckeln för Rum nu är kombinationen av Fastighetsbet. och Namn, eftersom Namn nu bara behöver vara unikt inom respektive Hus.

3. (Totalt: 10p) Ett universitet använder följande databasstruktur för att hålla reda på studenter, kurser (inklusive examinatore/lärare) och program (inklusive programansvariga).

Person(Pnr, Förnamn, Efternamn, Adress)

Student(Pnr, ProgramKod)

Lärare(Pnr)

Kurs(KursId, AntalHP, Godkänthgräns, Pnr)

Program(ProgramKod, AntalHP, Pnr)

Tentar(Pnr, KursId, Resultat)

Ingår(KursId, ProgramKod)

- a) (8p) Rita upp en ER-modell som skulle resultera i ovanstående Databasstruktur. Modellen skall innehålla **minst ett arv** och **ett sammansatt attribut**. Motivera varje steg genom att mkt kort beskriva den regel i "kokboken" som använts. Om du ser flera möjliga alternativa ER-modeller, så välj det alternativ som gör att man kan skriva vettiga SQL-frågor i uppgift 4 nedan.

Lösning:

Se figur 8. Steg 1 i kokboken (Varje vanlig entitetstyp blir en tabell, attribut blir kolumner) verkar ha tillämpats på *Person*, *Student*, *Lärare*, *Kurs* och *Program*. Gör vi dem till egna entiteter med attribut så får vi ungefär rätt databasstruktur för dessa. Dock väntar vi lite med Tentar och Ingår vilka kanske är resultat av någon annan regel.

Steg 2 i kokboken (Varje 1:N-samband blir referensattribut i “många”-sidans tabell) verkar ha tillämpats på ett samband mellan Lärare och Kurs resp. Program, vi kallar sambanden Examinator och Programansvarig. Samma sak med Student och Program, vi kallar det sambandet Tillhör.

Steg 3 handlar om 1:1-samband, något sådant verkar vi inte ha.

Steg 4 (Varje N:M-samband bildar egen tabell) verkar ha gett upphov till tabellen *Ingår* och *Tentar*.

Steg 5 handlar om flervägssamband, något sådant verkar inte finnas.

Steg 6 handlar om attribut på samband, Resultat i Tentar verkar vara ett attribut.

Steg 7 handlar om svaga entiteter, några sådana verkar inte finnas.

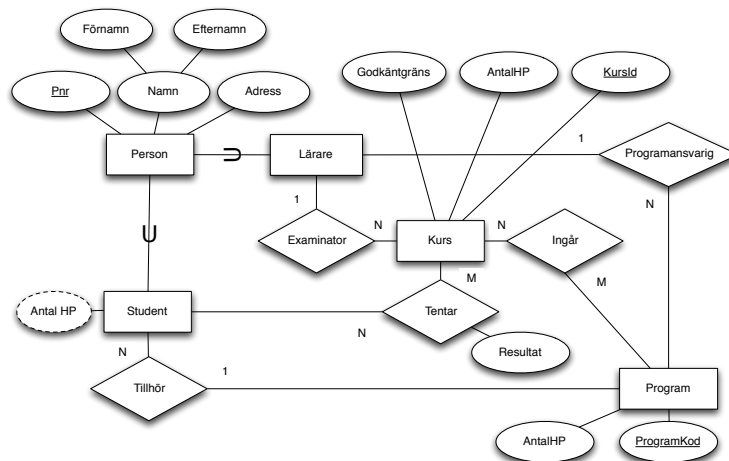
Steg 8 (sammansatta attribut blir som delarna). Det var givet att vår modell skall innehålla minst ett sammansatt attribut, vi låter Namn vara sammansatt av Förnamn och Efternamn.

Steg 9 (Varje flervärd attribut blir en egen tabell. Primärnyckeln består av entitetstypens primärnyckel, kombinerad med det flervärda attributet.) Något sådant verkar inte finnas.

Steg 10 handlar om härledda attribut. Dessa syns inte i databasstrukturen.

Steg 11 (Varje subclass blir en egen tabell) Det var givet att vår modell skall innehålla minst ett arv. Det verkar rimligt att tro att Student och Lärare är subclasser av Person.

Tillämpar vi reglerna i kokboken på ER-modellen i figur 8 kommer vi således få den givna databasstrukturen.



Figur 8: En ER-modell som skulle ge upphov till den givna databasstrukturen. Notera att *Antal HP* tillkommer i deluppgift 3b

b) (2p) Hur skulle ER-modell och databasstruktur ändras om man lade till ett härlett attribut ‘Antal HP’ till Studenterna?

Lösning: I figuren skulle vi få en ny sträckad oval som det står "Antal HP" i, med enkelstreck till Student-entiteten, se figur 8.
Databasstrukturen ändras inte eftersom de härledda attributen inte innehåller någon egen data. Istället kan man lägga till en vy för att beräkna det härledda attributet.

4. (Totalt: 25p) Givet samma databasstruktur som i frågan ovan, dvs

Person(Pnr, Förnamn, Efternamn, Adress)

Student(Pnr, ProgramKod)

Lärare(Pnr)

Kurs(KursId, AntalHP, Godkäntgräns, Pnr)

Program(ProgramKod, AntalHP, Pnr)

Tentar(Pnr, KursId, Resultat)

Ingår(KursId, ProgramKod)

Skriv SQL-frågor som löser följande uppgifter.

a) (2p) Lista alla Personer vars efternamn börjar på B.

Lösning:

```
SELECT * FROM "Person" WHERE "Efternamn" LIKE 'B%'
```

b) (2p) Lista alla Studenter vars efternamn börjar på B.

Lösning:

```
SELECT * FROM "Person" WHERE  
"Efternamn" LIKE 'B%'  
AND "Pnr" IN  
( SELECT "Pnr" FROM "Student" )
```

c) (2p) Lista alla Kurser på mer än 5 HP.

Lösning:

```
SELECT * FROM "Kurs" WHERE "AntalHP" > 5
```

d) (2p) Lista alla studenter (Pnr) som har klarat kursen DD101.

Lösning:

```
SELECT "Student"."Pnr" FROM "Kurs", "Tentar", "Student"  
WHERE "Kurs"."KursId" = "Tentar"."KursId"  
AND "Tentar"."Pnr" = "Student"."Pnr"  
AND "Kurs"."KursId" = 'DD101'  
AND "Kurs"."Godkäntgräns" <= "Tentar"."Resultat"
```

e) (4p) Lista alla studenter (Pnr) och hur många HP varje student har tagit. Kalla de två kolumnerna för Pnr och TotalHP.

Lösning:

```

SELECT "Student"."Pnr", SUM( "AntalHP" )AS TotalHP
FROM "Kurs", "Tentar", "Student"
WHERE "Kurs"."KursId" = "Tentar"."KursId"
AND "Tentar"."Pnr" = "Student"."Pnr"
AND "Kurs"."Godkäntgräns" <= "Tentar"."Resultat"
GROUP BY "Student"."Pnr"

```

- f) (4p) Antag att du redan löst uppgiften ovan, och lagrat resultatet i en Vy som heter *KlaradeHP*. Lista alla studenter (Förnamn och Efternamn) som borde ta examen, dvs har tillräckligt många poäng för sina program.

Lösning:

```

SELECT "Person"."Förnamn", "Person"."Efternamn"
FROM "Person" WHERE "Person"."Pnr" IN
( SELECT "Pnr" FROM "KlaradeHP", "Student", "Program"
WHERE "KlaradeHP"."Pnr" = "Student"."Pnr"
AND "Student"."ProgramKod" = "Program"."ProgramKod"
AND "TotalHP" >= "Program"."AntalHP" )

```

- g) (2p) Lista snittpoängen per student på alla tentor hen har skrivit.

Lösning:

```

SELECT "Pnr", AVG( "Resultat" )
FROM "Tentar" GROUP BY "Pnr"

```

- h) (3p) Lista alla Programansvariga som inte finns med i tabellen med Lärare. **Lösning:**

```

SELECT "Program"."Pnr" FROM "Program"
WHERE "Program"."Pnr"
NOT IN ( SELECT "Pnr" FROM "Lärare" )

```

- i) (4p) Lista Efternamn på alla lärare som undervisar mer än 2 kurser.

Lösning:

```

SELECT "Efternamn" FROM "Person"
WHERE "Person"."Pnr" IN
( SELECT "Pnr" FROM "Kurs"
GROUP BY "Pnr"
HAVING COUNT( * ) > 2 )

```