

Tentamen
Databasmetodik –
DB:DSK/FK/DVK/ATD/SP/EIT mfl. äldre
kurstillfällen
8 augusti 2013 kl. 9 - 13

Inga hjälpmedel tillåtna (syntaxsammanställning behövs inte på denna tentamen)

Skriv bara på en sida av pappret

Skriv läsligt, annars kan tentamen inte rättas

OM du gör en eller flera uppgifter som FX-uppgift så vänligen ange detta!

Lycka till!

Uppgift 1 (svarar mot lärandemål 1 avseende databasteknik och informationsadministration) Rätt lösningsrad kommer efter sista frågan!

Uppgift 1 består av 10 flervalsfrågor där rätt alternativ ska väljas (bara *ett* är rätt). Välj ett alternativ för var och en av flervalsfrågorna nedan.

1) Vilket av följande påståenden om vyer är sant?

- a) En vy kan uppdateras om koden som definierar vyn baserar sig på högst två bastabeller.
- b) En vy kan uppdateras koden som definierar vyn inte innehåller aggregatfunktioner.
- c) En vy kan uppdateras om koden som definierar vyn inte innehåller aggregatfunktioner och bara baserar sig på en bastabell.

2) Vilket av följande påståenden är sant om entitetsintegritet?

- a) En sammansatt primärnyckels ingående kolumner måste alla var för sig vara dubblettfria.
- b) Om primärnyckeln består av flera kolumner får högst en av de kolumnerna vara NULL.
- c) Alla kandidatnycklar i en tabell kan väljas till att bli primärnyckeln i den tabellen.

3) Vilken operation är unär?

- a. Join
- b. Aggregering
- c. Differens
- d. Kartesisk produkt

4) ON DELETE RESTRICT är ett exempel på en främmande nyckel-regel, vad innebär den?

- a) Man får inte ta bort en rad så länge primärnyckel-kolumnen i denna rad motsvaras av främmande nyckel-värden i andra tabeller.
- b) Man får inte ta bort en rad så länge värdet på främmande nyckel-kolumnen i denna rad svarar mot en primärnyckel-kolumn i en annan tabell.
- c) Man får inte ta bort vare sig rader som innehåller en primärnyckel-kolumn om denna kolumn motsvaras av främmande nyckel-kolumner i en annan tabell eller rader som innehåller en främmande nyckel-kolumn mot en primärnyckel i en annan tabell.

5) Vilket av följande är falskt?

- a) Efter en DELETE-sats kan tabellen ha färre rader än förr.
- b) Efter en DELETE-sats kan tabellen vara oförändrad.
- c) Efter en UPDATE-sats kan tabellen vara oförändrad.
- d) Efter en UPDATE-sats kan tabellen ha färre rader än förr.
- e) Efter en DELETE-sats kan tabellen vara tom.
- f) Efter en UPDATE-sats kan samtliga rader i en tabell vara förändrade.

6) En relations/tabells kardinalitet är:

- a) antalet attribut/kolumner i relationen/tabellen.
- b) högsta värdet som finns i primärnyckel-kolumnen.
- c) antalet rader i relationen/tabellen

7) Vilket av följande är inte sant vad gäller nycklar?

- a) En kandidatnyckel som inte väljs till primärnyckel blir alternativnyckel.
- b) Främmande nyckel-kolumner kan ingå i en primärnyckel.
- c) En supernyckel kan alltid bli primärnyckel.

8) Vad av följande gäller för 2PL (Two-Phase Locking)?

- a) En transaktion kan endast begära ett nytt lås om den inte redan har släppt ett annat lås.
- b) En transaktion kan få ett nytt lås trots att den redan släppt ett lås, om den körs mot endast en tabell.
- c) En transaktion kan begära ett nytt lås om den först släpper ett lås den fått tidigare.

9) Vilket av följande måste alltid finnas i en relationstabell?

- a) En surrogatnyckel.
- b) En främmande nyckel.
- c) En primärnyckel
- d) en alternativnyckel

10) Varför använder databashanteringssystem sig av lås med avseende på resurser (tabeller, rader, kolumner etc.) ?

- a) För att minimera att data, som accessas (skrives, läses) av olika användare samtidigt, förändras så att de samtidiga användarna ser olika information om samma resurs.
- b) För att få skrivningar och läsningar av informationen i databasen att ske så effektivt som möjligt.
- c) För att omöjliggöra att data, som accessas (skrives, läses) av olika användare samtidigt, förändras på ett sådant sätt att de samtidiga användarna ser olika information om samma resurs.

Rätt lösningsrad: c c b a D c c a c c

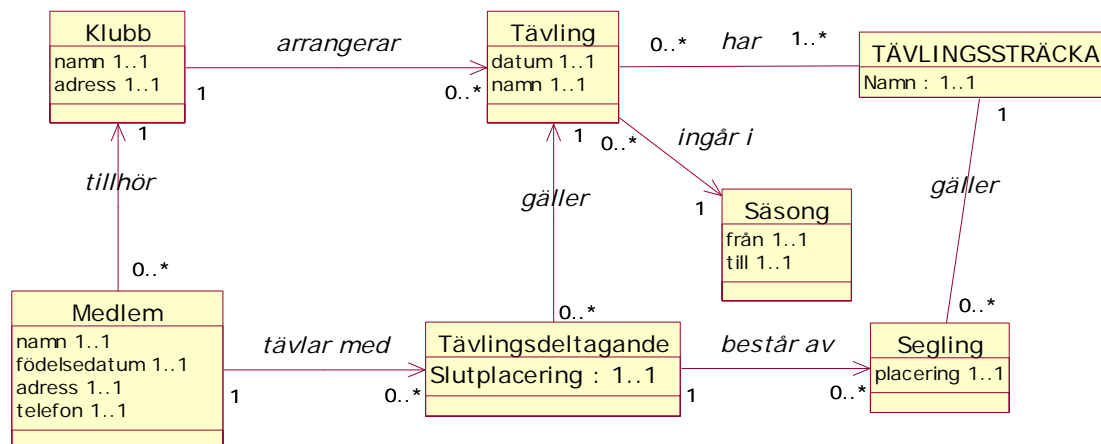
Uppgift 2 (svarar mot lärandemål 2, konceptuell modellering och databasmodellering)

Konstruera ett UML klass-schema som ger möjlighet att representera samtliga utsagor nedan. Ange avbildningsregler för samtliga attribut och för samtliga roller i associationer, partiella attribut (attribut med min-värde = 0) ska undvikas:

Nu skall ni få försöka modellera en tävlingsverksamhet, nämligen JOLLESEGLING. Jag kommer medvetet att förenkla en del, men i det stora hela stämmer det rätt bra. Tänk er att modellen skall användas för att representera information om alla tävlingar inom ett distrikt. Alla tävlande kommer från klubbar i detta distrikt. De tävlande är antingen seniorer eller juniorer. En tävling består av flera seglingar som varje tävlande deltar i och varje segling gäller en viss tävlingssträcka. T ex består tävlingen 'Gotland Runt' av tre sträckor: Visby-Sudret, Sudret-Fårö och den fruktade sträckan Fårö-runt. Som vinnare in en tävling koras den som har lägst summa när man adderat alla placeringar på de olika seglingarna för de olika sträckorna och räknat bort den sämsta placeringen. Under en säsong kan en seglare delta i flera tävlingar. För klubbarna och individerna behövs sedvanliga adress-/personuppgifter. För tävlingarna behövs information om datum och arrangerande klubb.

(Förutsätt att det är en enmansjolle typ E-jolle och att segelnr och båtnummer är utan betydelse)

Lösningförslag:



Uppgift 3(svarar mot lärandemål 4, analytisk db-design)

- a) Konstruera en tabell som *inte* är i 1NF, motivera ditt svar! Visa även hur tabellen kan överföras till 1NF!

Vilken tabell som helst som har icke-atomära värden i sina kolumner. Tex

OST

Ostnamn

Osttyp

Pris

Svecia	Hårdost	60.78
Herrgård	{Hårdost, Mjukost}	{50.00, 90.00}
Brie	Vitmögelost	140.00

Här har alltså den andra raden en mängd värden för Osttyp och Pris (givet att datatypen för Ostnamn och Osttyp är en sträng och datatypen för Pris är flyttal).

Ett sätt att överföra tabellen till 1NF är att distribuera ut alla icke-atomära värden till egna rader. Fler sätt finns.

OST

Ostnamn	Osttyp	Pris
Svecia	Hårdost	60.78
Herrgård	Hårdost	50.00
Herrgård	Mjukost	90.00
Brie	Vitmögelost	140.00

b) Betrakta följande tabell:

DIAMANT(**AffärsId**, **Diamanttyp**, Viktklass, Pris), där kolumnerna AffärsId och Diamanttyp tillsammans utgör primärnyckel..

Följande funktionella beroenden råder :

AffärsId, Diamanttyp → Viktklass, Pris
Diamanttyp → Pris

Vilken normalform är tabellen DIAMANT i? Motivera ditt svar!

Överför tabellen till 3NF (motivera varför den inte är i 3NF i så fall och hur den kan bli det) eller motivera varför den redan är i 3NF.

Lösningsförslag:

DIAMANT är i 1NF och inte mer pga att den inte är i 2NF pga att det finns ett icke nyckel attribut som är partiellt beroende av nyckeln.

För att överföra tabellen till 3NF överförs den först till 2NF:

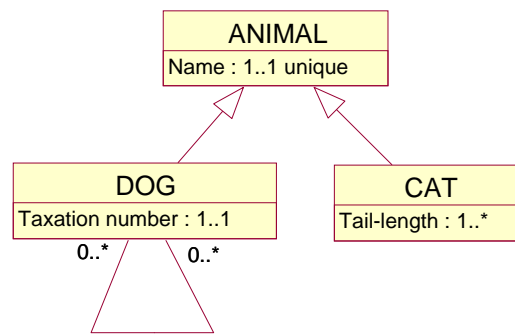
DIAMANT(Affärsid, Diamanttyp, Viktklass)

DIAMANTPRIS(Diamanttyp, Pris), DIAMANT.Diamanttyp FN mot DIAMANTPRIS.Diamanttyp

Båda dessa tabeller är nu i 2NF eftersom inga partiella eller transitiva beroenden existerar mellan icke nyckelattribut och primärnyckeln.

Uppgift 4(svarar mot lärandemål 3, syntetisk db-design)

Betrakta följande konceptuella schema:



ANIMAL identifieras av sitt Name, övriga identifierare framgår av schemat. Surrogatnycklar får inte införas.

Översätt det konceptuella schemat till ett databas-schema. Du kan använda följande struktur för definition av databas (dvs CREATE TABLE-satser ska *inte* skrivas):

TABELLNAMN1(Primärnyckelkolumn1, Primärnyckelkolumn2, annankolumn1, annankolumn2)
 TABELLNAMN2(Primärnyckelkolumn1, annankolumn1, annankolumn2), där
 TABELLNAMN2.(annankolumn1,annankolumn2) är FN mot TABELL1.(Primärnyckelkolumn1,Primärnyckelkolumn2).

Där sista raden betyder att tabellen TABELLNAMN2 har en främmande nyckel mot tabellen TABELLNAMN1.

Lösningförslag:

ANIMAL(Name)

DOG(Name, Taxation number), DOG.Name FN mot ANIMAL.Name

CAT(Name), CAT.Name FN mot ANIMAL.Name

CAT-TAILS(Name, Length), CAT_TAILS.Name FN mot CAT.Name

DOGDOG(DN1, DN2), DOGDOG.DN1 FN mot DOG.Name, DOGDOG.DN2 FN mot DOG.Name

Uppgift 5 (svarar mot lärandemål 5, frågespråk mot relationsmodellen)

Databasschemat nedan modellerar det faktum att hästar har färger och mankhöjder, personer har adresser och personer rider på olika hästar. Följande främmande nyckel-regler gäller:

RIDNING.HästID << HÄST.HästID

RIDNING.Person << PERSON.Namn

HÄST

HästId	HästNamn	Mankhöjd	Färg
1	Man O' War	1.60	Röd
2	Ego Boy	1.62	Grå
3	Darley Arabian	1.67	Röd
4	Byerley Turk	1.51	Brun

RIDNING

Person	HästId	Datum
Lisa	1	120321
Lisa	1	120101
Lily	4	121231
John	4	121231
Lily	1	121231
Lisa	2	120321
John	1	120401

PERSON

Namn	Adress
Lisa	Byvägen 3
Lily	Byvägen 11 B
Olle	Strandvägen 8
John	Solgatan 4

(Primärnycklarna i tabellerna ovan är understrukna).

- a) **Översätt följande till SQL och relationsalgebra: Vilka hästar har ridits av personer på Byvägen 3? Ange deras namn och mankhöjd? OBS att lösningen måste vara generell och inte basera sig på hur innehållet i databasen ser ut just nu!**

Lösningsförslag SQL:

```
SELECT HästNamn, Mankhöjd
FROM PERSON, RIDNING, HÄST
WHERE Namn=Person
AND HÄST.HästId = RIDNING.HäsId
AND Adress = 'Byvägen 3'
```

Lösningsförslag relationsalgebra:

$$TEMP \leftarrow (RIDNING \bowtie_{Person=Person} (\sigma_{Adress='Byvägen 3'} PERSON))$$

$$\Pi_{Hästnamn, Mankhöjd} (HÄST \bowtie TEMP)$$

/* här kan man använda natural-join eftersom kolumnen 'HästId' finns i både TEMP och HÄST */

- b) **Översätt följande till SQL och relationsalgebra: Vilka hästar har ridits av alla personer, det räcker att ange hästens HästId? OBS att lösningen måste vara generell och inte basera sig på hur innehållet i databasen ser ut just nu!**

Lösningsförslag SQL:

```

SELECT HästId
FROM HÄST H
WHERE NOT EXISTS
  (SELECT *
   FROM PERSON
   WHERE Namn NOT IN
     (SELECT Person
      FROM RIDNING R
      WHERE R.HästId=H.HästId))

```

Lösningförslag relationsalgebra:

Nämnare (Person) $\leftarrow \pi_{\text{Namn}}(\text{PERSON})$

Täljare $\leftarrow \pi_{\text{HästId, Person}}(\text{RIDNING})$

Svar \leftarrow Täljare DIVISION Nämnare

- c) Vad blir resultatet (rita tabell med rader) om följande SQL-sats exekveras ?
(HÄR ska svaret basera sig på hur databasen ser ut just nu, dvs på exempel-relationerna ovan)

```

SELECT Person
FROM PERSON, RIDNING
WHERE Person=Namn
AND Adress = 'Solgatan 4'

```

Lösningförslag SQL:

```

Person
John
John

```

- d) Vad blir resultatet (rita tabell med rader) om följande relationsalgebraiska uttryck exekveras (baserat på exempel-relationerna ovan)? (HÄR ska svaret basera sig på hur databasen ser ut just nu, dvs på exempel-relationerna ovan)

$\pi_{\text{Person}}(\sigma_{\text{Adress}='Solgatan 4'}(\text{RIDNING} \bowtie_{\text{Person=Namn}} \text{PERSON}))$

Lösningförslag relationsalgebra:

```

Person
John

```