

**Tentamen**  
**Databasmetodik –**  
**DB:DSK/FK/DVK/ATD/SP/EIT mfl. äldre**  
**kurstillfällen**  
**Lördag 8 juni kl. 10 - 14**

Inga hjälpmedel tillåtna (syntaxsammanställning behövs inte på denna tentamen)

Skriv bara på en sida av pappret

Skriv namn på varje papper

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)

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. **Rätt lösningsrad kommer efter sista frågan!**

### 1) Vilket av följande påståenden om vyer är sant?

- En vy kan aldrig uppdateras.
- En vy kan inte användas för att förenkla ett SQL-uttryck eftersom vyns resultat inte lagras i databasen.
- En vy kan användas som en mekanism för att ge olika användare tillgång till olika delar av databasen utan att känna till övriga delar.

### 2) Vilket av följande påståenden är sant om entitetsintegritet?

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

### 3) Unionskompatibilitet krävs för de relationsalgebraiska operationerna:

- snitt och division
- union och difference
- theta-join och natural join

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

- Man får inte ta bort en rad så länge primärnyckel-kolumnen i denna rad motsvaras av främmande nyckel-värden i andra tabeller.
- 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.
- 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) Tabellen RIDNING nedan bryter mot en typ av regel, vilken?

HÄST		RIDNING		
HästId	HästNamn	Person	HästId	Datum
1	Man O' War	Lisa	1	120321
2	Ego Boy	Lisa	1	120101
3	Darley Arabian	Lily	4	121231
4	Byerley Turk	John	5	121231
		Lily	1	121231

RIDNING.HästId << HÄST.HästId

- Referential integrity.
- Entity integrity
- 3NF.

### 6) En relations/tabells kardinalitet är:

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

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

- 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 är sant vad gäller referensintegritet?**

- a) En främmande nyckel får aldrig vara NULL.
- b) En främmande nyckel får aldrig innehålla dubletter.
- c) En främmande nyckel som inte tillåts vara NULL, måste matcha ett exakt värde i primärnyckeln i den tabell den främmande nyckeln refererar till.

**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 a c c a c c

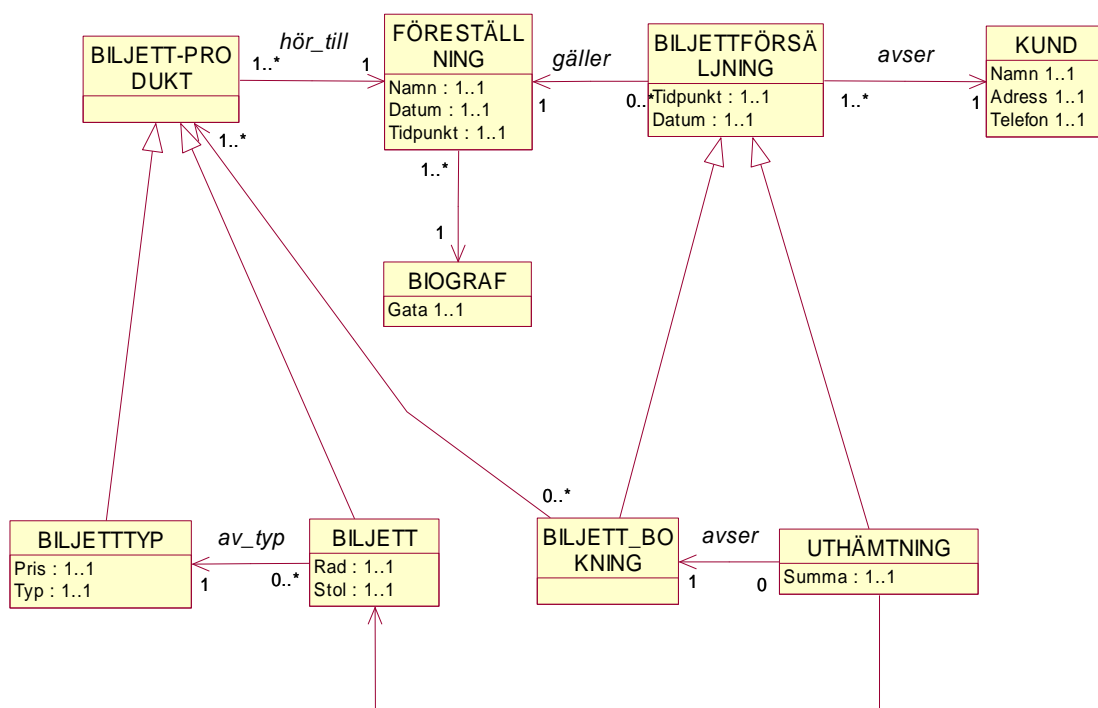
## Uppgift 2(svarar mot lärandemål 2, konceptuell modellering och databasmodellering)

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:

a) Konstruera ett konceptuellt schema, t.ex. i form av ett UML klassdiagram, för ett administrativt system för bokning och försäljning av biobiljetter, vars informationskrav framgår nedan. Avbildningsregler (cardinalities) skall anges för samtliga associationer.

- Anna Larsson bokade tre biljetter på rad 5, stol 11 och 12 och rad 10 stol 77 på Astoria på Kungsgatan 21 mars 1965, kl. 15. Föreställningen börjar 22 mars 1965 kl. 18.
- Anna Larsson bokade även en biljett på Rival, Biblioteksgatan, rad 10, stol 22, 21 juni 1999.
- Henrik Andersson bokade 5 ospecificerade biljetter (= han är garanterad 5 platser men vilka han får bestäms när han hämtar ut dem) på Astoria Kungsgatan 22 mars 1965, kl. 12. Föreställningen börjar 22 mars 1965 kl. 18.
- Anna hämtade aldrig ut de bokade biljetterna på Astoria och dessa biljetter såldes istället (kl. 17, 22 mars 1965) till Henrik Larsson som dessutom köpte två ytterligare biljetter på rad 8, stol 45 och 46 till den föreställningen (22 mars 1965 kl. 18).
- Astoria Kungsgatan tar alltid 75 kr för sina kvällsbiljetter (föreställningar som börjar kl. 18 och senare) och 50 kr för matiné-biljetter (föreställningar innan kl. 18).
- Rival på Biblioteksgatan tar 100 kr för sina kvällsbiljetter och 50 för matinébiljetterna.

### Lösningsförslag:



Här kan man tänka sig många olika lösningar. En svårighet var att man antingen kan köpa biljetter till en viss plats (stol och rad) eller också ospecificerade biljetter, en lösning är att ha en super-class BILJETTPRODUKT för de två typerna av biljetter. Då kan man boka bägge typerna av produkter (antingen bara biljettypen utan rad eller plats angiven eller också en specifik biljett med plats etc.) – visas genom en association mellan BILJETTBOKNING och BILJETTPRODUKT.

### 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!

Lösningsförslag:

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 och funktionella beroenden:

PERSON(Pnr, Namn, Vikt, Viktklass, Gata, Postnummer, Favoritmat)

Pnr → Namn, Vikt, Viktklass, Gata, Postnummer, Favoritmat

Vikt → Viktklass

Gata → Postnummer

Tabellen PERSON är i minst 1NF.

**Bestäm en primärnyckel för tabell PERSON. Motivera ditt val.**

**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:

Det finns ett funktionellt beroende som visar att kolumnen 'Pnr' bestämmer alla övriga kolumner funktionellt – därför kan 'Pnr' väljas till primärnyckel.

Tabellen var i minst 1NF (givet i texten), den är även i 2NF med avseende på den icke sammansatta primärnyckeln (med en icke sammansatt primärnyckel kan inga övriga icke-nyckelattribut vara partiellt funktionellt bestämda av primärnyckeln). Tabellen är dock inte i 3NF pga att det finns två transitiva beroenden: ett mellan Postnummer och primärnyckeln och ett mellan Viktklass och primärnyckeln.

Pnr → Gata (dekomponeringsregeln från Armstrongs axiom) och Gata → Postnummer (givet i texten) – således gäller (transitiva regeln från Armstrongs axiom) Pnr → Postnummer transitivt.

Pss Pnr → Vikt (dekomponeringsregeln från Armstrongs axiom) och Vikt → Viktklass (givet i texten) – således gäller (transitiva regeln från Armstrongs axiom) Pnr → Viktklass transitivt.

Vi bryter ut de kolumner som hade ett transitivt funktionellt beroende till primärnyckeln plus deras determinanter till två nya tabeller:

VIKT(Vikt, Viktklass) och

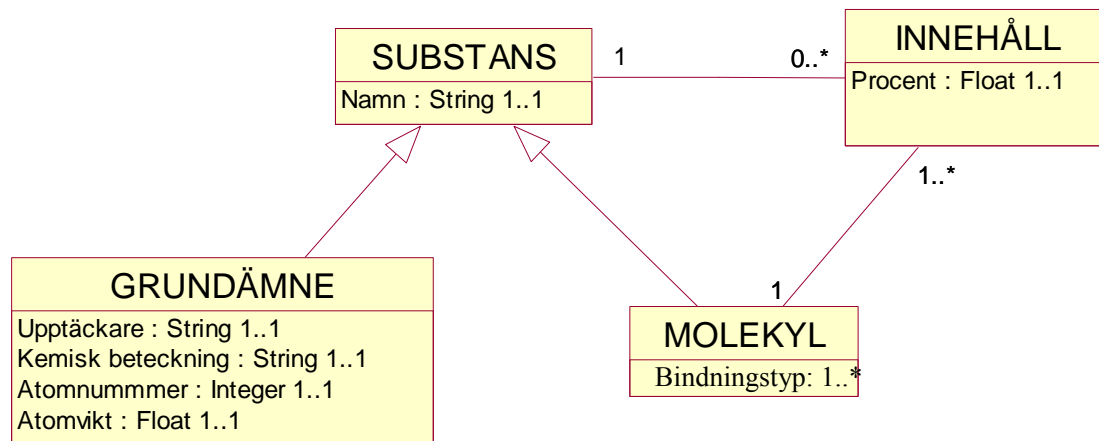
GATA(Gata, Postnummer)

Kvar i den gamla tabellen blir PERSON(Pnr, Vikt, Gata, Farvoritmat)

Två nya främmande nycklar har uppstått: PERSON.Vikt är FN mot VIKT.Vikt och PERSON.Gata är FN mot GATA.Gata.

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

Betrakta följande konceptuella schema:



SUBSTANS identifieras av sitt namn och INNEHÅLL av sina associationer mot SUBSTANS respektive MOLEKYL. Övriga identifierare framgår av schemat.

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:

**SUBSTANS(Namn)**

**MOLEKYL(Namn)** där MOLEKYL.Namn är FN mot SUBSTANS.Namn

**BINDNING(Mnamn, Bindningstyp)** där BINDNING.Mnamn är FN mot MOLEKYL.Namn

**GRUNDÄMNE(Namn, Upptäckare, Kemisk\_beteckning, Atomnummer, Atomvikt)**, där  
 GRUNDÄMNE.Namn är FN mot SUBSTANS.Namn

**INNEHÅLL(Snamn, Mnamn, Procent)**, där INNEHÅLL.Snamn är FN mot SUBSTANS.Namn och  
 INNEHÅLL.Mnamn är FN mot MOLEKYL.Namn

**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**

<u>HästId</u>	<u>HästNamn</u>	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**

<u>Person</u>	<u>HästId</u>	<u>Datum</u>
Lisa	1	120321
Lisa	1	120101
Lily	4	121231
John	4	121231
Lily	1	121231
Lisa	2	120321
John	1	120401

**PERSON**

<u>Namn</u>	<u>Adress</u>
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 personer har ridit hästen Man O'War, ange deras namn och adress?

**Lösningsförslag SQL:**

```
SELECT Namn, Adress
FROM PERSON, RIDNING, HÄST
WHERE Namn=Person
AND HÄST.HästId = RIDNING.HäsId
AND Hästnamn = 'Man O'War'
```

**Lösningsförslag relationsalgebra:**

TEMP  $\leftarrow$  (RIDNING  $\bowtie$ <sub>Person=Namn</sub> PERSON)

$\Pi_{\text{Namn, Adress}} (\sigma_{\text{Hästnamn}='Man O'War'}(\text{HÄST} \bowtie \text{TEMP}))$



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

- b) Översätt följande till *SQL och relationsalgebra*: Vilka personer har ridit alla hästar, det räcker att ange deras namn?

**Lösningsförslag SQL:**

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

**Lösningsförslag relationsalgebra:**

Nämnare  $\leftarrow \pi_{\text{HästId}}(\text{HÄST})$   
 Täljare  $\leftarrow \pi_{\text{Person, HästId}}(\text{RIDNING})$   
 Svar  $\leftarrow \text{Täljare DIVISION Nämnare}$

- c) Vad blir resultatet (rita tabell med rader) om följande SQL-sats exekveras (basera svaret på exempel-relationerna ovan)?

```
SELECT Person
FROM PERSON, RIDNING
WHERE Person=Namn
AND Adress = 'Byvägen 3'
```

**Lösningsförslag SQL:**

```
Person
Lisa
Lisa
Lisa
```

- d) Vad blir resultatet (rita tabell med rader) om följande relationsalgebraiska uttryck exekveras (basera svaret på exempel-relationerna ovan)?

$\pi_{\text{Person}}(\sigma_{\text{Adress}='Byvägen 3'}(\text{RIDNING} \bowtie_{\text{Person=Namn}} \text{PERSON}))$

**Lösningsförslag relationsalgebra:**

**Person**  
Lisa