



Försättsblad till skriftlig tentamen vid Linköpings Universitet

Datum för tentamen	2014-01-09
Sal (2) Om tentan går i flera salar ska du bifoga ett försättsblad till varje sal och <u>ringa in</u> vilken sal som avses	G35 G37
Tid	8-12
Kurskod	729G28
Provkod	TEN1
Kursnamn/benämning Provnamn/benämning	Webprogrammering och databaser Tentamen
Institution	IDA
Antal uppgifter som ingår i tentamen	6
Jour/Kursansvarig Ange vem som besöker salen	Eva Ragnemalm
Telefon under skrivtiden	013-282768
Besöker salen ca kl.	ca kl. 09
Kursadministratör/kontaktperson (namn + tfnr + mailaddress)	Anna Grabska Eklund, ankn. 2362, anna.grabska.eklund@liu.se
Tillåtna hjälpmedel	Inga
Övrigt	
Vilken typ av papper ska användas, rutigt eller linjerat	Valfritt
Antal exemplar i påsen	



Försättsblad till skriftlig tentamen vid Linköpings Universitet

Datum för tentamen	2014-01-09
Sal (2) Om tentan går i flera salar ska du bifoga ett försättsblad till varje sal och <u>ringa in</u> vilken sal som avses	G35 G37
Tid	8-12
Kurskod	729G28
Provkod	TEN1
Kursnamn/benämning Provnamn/benämning	Webprogrammering och databaser Tentamen
Institution	IDA
Antal uppgifter som ingår i tentamen	6
Jour/Kursansvarig Ange vem som besöker salen	Eva Ragnemalm
Telefon under skrivtiden	013-282768
Besöker salen ca kl.	ca kl. 09
Kursadministratör/kontaktperson (namn + tfnr + mailaddress)	Anna Grabska Eklund, ankn. 2362, anna.grabska.eklund@liu.se
Tillåtna hjälpmedel	Inga
Övrigt	
Vilken typ av papper ska användas, rutigt eller linjerat	Valfritt
Antal exemplar i påsen	

Linköpings Universitet
Institutionen för datavetenskap
Eva L. Ragnemalm

TENTAMEN

729G28

Webprogrammering och databaser

Datum: **2014-01-09 tid: 8-12**

Jourhavande lärare: **Eva Ragnemalm**

Besöker tentamenslokalen: **c:a kl 9.**

Tel: **013-28 2768**

Hjälpmedel: **inga.**

Poängfördelning:

Uppgift	Poäng
1	5
2	6
3	3
4	8
5	4
6	7
Totalt	33

Betygsgränser: G: 20, VG: 27

1. **Begreppstips:** För varje delfråga svara 1, X eller 2 (bara ett alternativ är rätt): (5p)
- a) Metadata är:
- 1. Meteorologisk data om väder.
 - x. Data som söks fram vid avancerade sökningar i flera databaser.
 - 2. Data om data i databasen.
- b) En databas' instans är:
- 1. Databasens faktiska innehåll vid ett visst ögonblick i tiden.
 - x. Beskrivningen av hur data i databasen ska vara strukturerat.
 - 2. Förändringen av databasens innehåll mellan två olika tider.
- c) En konceptuell datamodell beskriver:
- 1. I vilket format data ska lagras i databasen, t.ex. om det ska vara heltal eller strängar.
 - x. Vilka begrepp i den verkliga världen som ska finnas representerade i databasen.
 - 2. Vilka tabeller databasen ska innehålla.
- d) En sambandstyp i ett ER-diagram sammanbinder:
- 1. En entitetstyp med dess attribut.
 - x. Två entitetstyper med varandra.
 - 2. Två eller fler entitetstyper med varandra.
- e) Ett fullt funktionellt beroende är:
- 1. ett specialfall av funktionellt beroende sådant att determinanten är minimal.
 - x. ett funktionellt beroende där determinantens värde beror av ett annat attribut i relationen.
 - 2. ett funktionellt beroende där man kan ta bort minst ett attribut ur determinanten och fortfarande ha ett funktionellt beroende.

Begrepp, normalisering, SQL: Givet en databas med två tabeller med följande definitioner (date är ett datum-format som inte spelar någon roll i uppgiften). Du kan anta att vi vid bearbetning av data aldrig manipulerar delar av attributen.:

```
CREATE TABLE musik (Lnr int, LNamn varchar (30), LTid float,
    artist int, album varchar (30), producent varchar (20),
    ProdÅr int, PRIMARY KEY (Lnr));
CREATE TABLE artister (gruppNr int, gNamn varchar (30),
    gStart date, gStop date, members varchar (100), PRIMARY
    KEY (gruppNr));
```

Det finns också ett constraint enligt följande:

```
ALTER TABLE musik ADD CONSTRAINT fk_musik_artister
    FOREIGN KEY (artist) REFERENCES artister(gruppNr);
```

2. Givet tabellen musik ovan, antag att den har följande fullt funktionella beroenden (varken mer eller mindre):
- LNr=>LNamn, Lnr=>LTid, LNr=>artist, LNr=>album, Lnr=>producent, LNr=>ProdÅr,
album=>artist, album=>producent, album=>ProdÅr (6p)

- a) Vilka av följande attributmängder är Supernyckel, Kandidatnyckel, Primärnyckel, inte nyckel alls, i tabellen musik?
 {LNr} {LNr, ProdÅr} {album} {album, artist}
- b) Vilken normalform uppfyller tabellen musik (motivera)?

3. Antag att tabellen musik, med definition enligt ovan, innehåller följande data:

LNr	LNamn	LTid	artist	album	producent	prodÅr
1	Den blomstertid nu kommer	2.34	2	Sommarvisor	AllMusik	2011
2	I denna ljuva sommartid	3.23	56	Sommarvisor	AllMusik	2011
3	Euphoria	4.45	234	Loreens Greatest Hits	Bolaget	2012

Ange en tabell Artister, med attribut enligt ovan, och fyll den med ett exempel-innehåll (dvs skriv ut några rader i tabellen) som gör att följande kommandon går bra att genomföra (samma innehåll ska fungera för båda deluppgifterna och kommandona ska verkligen ha effekt): (3p)

- a) `Delete from artister where gruppNr=23;`
- b) `Insert into musik values (4, "Waterloo", 3.45, 12, "Waterloo", PolarMusic, 1974);`

Designuppgift: Musikfirman AllaKanSpela hyr ut musikinstrument till folk som nyss börjat lära sig spela. De vill ha hjälp att hålla ordning på instrumenten, kontrakten och kunderna. Varje instrument har en unik identitetskod, en typ (t.ex. blockflöjt, fiol osv.) och en viss månadshyra som bara uppdateras när man får in instrumentet mellan uthyrningar. Man vill ha kvar information om gamla kontrakt (vem som hyrde vad under vilken period) även efter kontrakttidens slut. Information om vilken hyra som gällde för gamla kontrakt behöver inte sparas. Det händer ganska ofta att samma kund hyr samma instrument igen, då upprättas ett nytt kontrakt för den nya hyresperioden. En kund kan ha flera kontrakt samtidigt (ofta står föräldrarna som kund fast man hyr instrument till barnen). Du kan anta att gamla kontrakt för instrument som inte är uthyrda just nu har ett slutdatum som är passerat. Man kan inte lägga in framtida kontrakt i databasen (startdatum kan aldrig vara större än dagens datum).

Man vill kunna ställa frågor av följande typ till databasen:

- a) Alla blockflöjter är uthyrda men vi behöver veta hur snart vi har en ledig. Sök ut instrument-id och slutdatum för uthyrda blockflöjter (dvs typ="blockflöjt" och dagens datum<slutdatum). Sortera på slutdatum så vi får det sökta datumet först.
- b) Det är dags att inventera. Hur många instrument av de olika typerna har vi? (Lista typ och hur många det finns av den typen, oavsett om de är uthyrda eller ej)
- c) Blockflöjt BF102 har en spricka, senaste kunden måste betala för ovarsamheten. Sök ut namn, adress och telefonnummer till den senaste kunden som hyrde blockflöjten (vi vet

att den inte är uthyrd - vi har den i handen). Bara den senaste kunden ska listas. Denna utsökning får delas upp i flera SQL-kommandon.

4. Rita upp ett ER-diagram för databasen. Markera kardinalitet och deltagande samt nycklar så som vi gjort i kursen. Om du tillför attribut som inte tydligt finns med i beskrivningen måste du motivera varför de behövs. (8p)
5. Konvertera ER-diagrammet till relationsmodellen (tabeller). Markera primärnycklar i relationerna samt främmande nycklar så som vi gjort i kursen. (4p)
6. Formulera sedan de uppräknade frågeställningarna i SQL. (7p)

Läs beskrivningen noga så att du inte missar något. Om du tycker det saknas information eller att du behöver göra ytterligare antaganden, skriv då ner dem. Om ditt ER-diagram eller relationsschema visar effekten av antaganden som inte redovisats och som inte stämmer med uppgiften får du poängavdrag.

Lycka till!