

Örebro universitet
Institutionen för teknik
[Thomas Padron-McCarthy \(thomas.padron-mccarthy@oru.se\)](mailto:thomas.padron-mccarthy@oru.se)

Tentamen i Databasteknik

för D1, SDU1 m fl

tisdag 10 januari 2012

Gäller som tentamen för:

DT1012 Datateknik A, Databasteknik, provkod 0100
DT1007 Datateknik A, Tillämpad datavetenskap, provkod 0310

Hjälpmedel:	Inga hjälpmedel.
Poängkrav:	Maximal poäng är 34. För betyget 3 respektive G krävs 21 poäng. För den som följt kursen hösten 2011 ger varje i tid inlämnad inlämningsuppgift en extra poäng. Den som <i>inte</i> gått kursen hösten 2011 får dessa (fem) extrapoäng ändå.
Resultat och lösningar:	Meddelas på kursens hemsida eller via e-post senast tisdag 31 januari 2012.
Återlämning av tentor:	Efter att resultatet meddelats kan tentorna hämtas på universitetets centrala tentamensutlämning.
Examinator och jourhavande:	Thomas Padron-McCarthy, telefon 070 - 73 47 013.

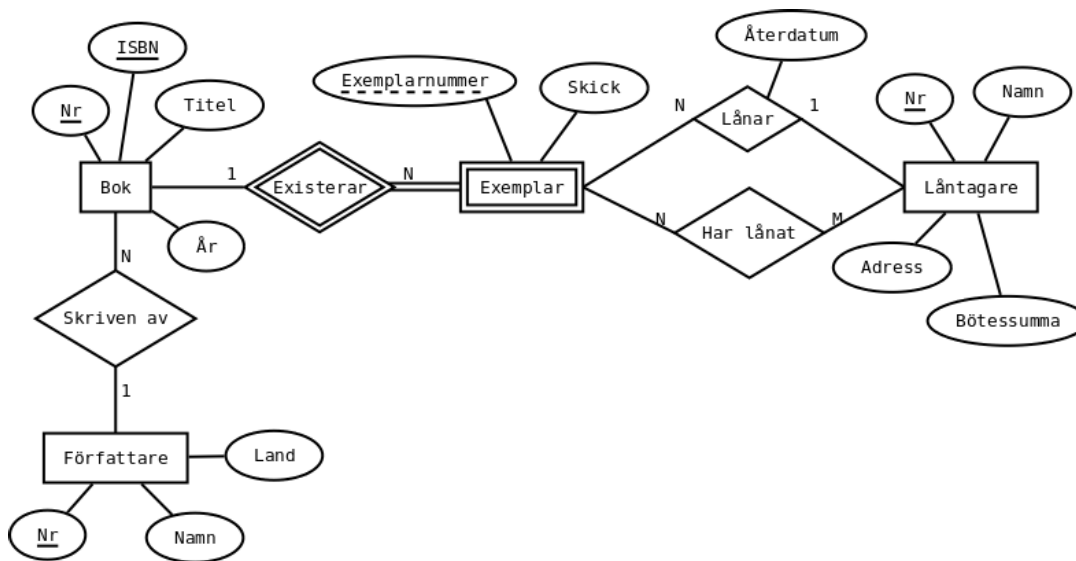
- Skriv tydligt och klart. Lösningar som inte går att läsa kan naturligtvis inte ge några poäng. Oklara formuleringar kommer att misstolkas.
 - Skriv den personliga tentamenskoden på varje inlämnat blad. Skriv *inte* namn eller personnummer på bladen.
 - Skriv bara på en sida av papperet. Använd inte röd skrift.
 - Antaganden utöver de som står i uppgifterna måste anges. Gjorda antaganden får inte förändra den givna uppgiften.
 - Skriv gärna förklaringar om hur du tänkt. Även ett svar som är fel kan ge poäng, om det finns med en förklaring som visar att huvudtankarna var rätt.
-

LYCKA TILL!

Scenario till uppgifterna



Universitetsbiblioteket i Örebro behöver en databas. Man ska lagra böcker, låntagare och utlånade böcker. Det här är ett ER-diagram för databasen:



Här är några förklaringar om de saker man vill lagra i databasen:

- **Böcker.** En bok är en viss utgiven bok, till exempel kursboken med titeln **Databasteknik**. Alla böcker har ett unikt ISBN-nummer, till exempel **978-91-44-04449-1**, men eftersom ISBN-nummer är så långa har biblioteket också infört ett unikt internt nummer.
- Varje bok finns i ett antal **exemplar**. Det är exemplaren som är de fysiska böcker som står i hyllorna på biblioteket. Till exempel kanske biblioteket har köpt in tre exemplar av boken **Databasteknik**. Varje exemplar har ett skick, som till exempel kan vara **som ny** eller **nästan förstörd**.
- **Författare.** Vi lagrar författare i databasen, och vilka författare som skrivit vilka böcker.
- **Låntagare** är personerna som lånar böckerna.
- I databasen ska det också lagras vilka låntagare som just nu har var och en av de utlånade böckerna, och även alla låntagare som tidigare någon gång lånat böckerna.
- När en låntagare lämnar tillbaka en bok efter återdatumet, måste hon betala böter, och därför håller vi reda på hur mycket **böter** varje låntagare har (kvar) att betala.

Uppgift 1 (6 p)

a) Om du jämför med hur världen ser ut på riktigt, finns det (åtminstone) en sak i ER-diagrammet som är fel. Vilken?

b) Implementera den beskrivna databasen i relationsmodellen, dvs översätt ER-diagrammet till tabeller. (Utgå från ER-diagrammet, även om det inte stämmer med världen.)

Du behöver inte skriva **create table**-kommandon i SQL, men du ska ange vilka relationer som finns och vilka attribut varje relation innehåller. Ange också alla kandidatnycklar, vilken av dessa som är primärnyckel, samt vilka referensattribut som finns och vad de refererar till.

Implementationen ska vara bra.

Uppgift 2 (5 p)

Medan de väntade på att du skulle bli klar med uppgift 1 gjorde personalen på biblioteket en egen databas, så att de nödortfigt kunde lagra böckerna och vilka som lånar dem. Den består av den här tabellen, som heter **BöckerOchLån**:

ISBN	Titel	Exemplarnummer	Skick	Låntagare	Adress	Återdatum
978-91-44-04449-1	Databasteknik	1	Dålig	Bo Ek	Vägen 4	2012-01-19
978-91-44-04449-1	Databasteknik	2	Som ny	Anna Berg	Stigen 7	2012-02-01
978-91-44-04449-1	Databasteknik	3	Som ny	<i>null</i>	<i>null</i>	<i>null</i>
978-91-44-07606-5	Vägen till C	1	Bra	Bo Ek	Vägen 4	2012-01-19
978-91-44-07606-5	Vägen till C	3	Som ny	Cassius Clay	Torget 2	2012-01-29

a) Vilka kandidatnycklar finns i tabellen?

b) Vilka fullständiga funktionella beroende finns i tabellen?

c) Vilken är den högsta normalform som tabellen uppfyller, av 1NF, 2NF, 3NF och BCNF? (Ledtråd: Det är inte BCNF.) Motivera svaret!

d) Beskriv ett problem som finns i den här tabellen, men som man skulle slippa om den hade uppfyllt BCNF.

Uppgift 3 (10 p)

Formulera följande frågor i SQL. Definiera gärna vyer om det underlättar. (Använd dina tabeller från uppgift 1.)

- Jag heter **Thomas Padron-McCarthy** och är författare. Vilket land kommer jag ifrån?
- Jag heter **Thomas Padron-McCarthy** och har skrivit en bok. Vad har den boken för titel?
- Hur många exemplar av boken med titeln **Databasteknik** finns just nu inne, alltså är inte utlånade? Vi vill veta antalet.
- Finns det några böcker som är med i databasen, men där vi inte har några exemplar? Vi vill veta ISBN-nummer och titel på dessa böcker.
- Vad är ISBN-numret och titeln på den bok som biblioteket har i flest antal exemplar?

Uppgift 4 (5 p)

- (3p) Databasen innehåller tiotusentals låntagare och miljoner böcker, författare och lån. De tre SQL-frågorna a, b och c i uppgiften ovan körs ofta (men med andra konstanter, till exempel andra personnamn än **Thomas Padron-McCarthy**) och behöver snabbas upp. Vi märker att det inte finns några index alls i databasen, inte ens på nycklar. Vilka index bör man skapa för att just de frågorna ska gå snabbt att köra? Motivera svaret!
- (1p) Vad är det för skillnad på ett index och en primärnyckel?
- (1p) Hur kan index göra att samma SQL-frågor går snabbare att köra? Om SQL-koden inte har ändrats, borde det väl ta precis lika lång tid att köra frågan som innan man skapade indexen?

Uppgift 5 (3 p)

Vilken databashanterare bör vi välja för vår databas? Nämn tre olika databashanterare, och tala om varför, eller varför inte, de passar för den här tillämpningen.

Uppgift 6 (2 p)

Vi kommer på att författare och låntagare alla är *personer*, med nummer och namn, och vill införa denna generalisering i ER-diagrammet. Rita om hela ER-diagrammet i scenariot till ett EER-diagram med arv enligt detta. Tänk på att sambandstyperna ska gå till rätt entitetstyp.

Uppgift 7 (3 p)

När en bok lämnas tillbaka ska noteringen om att den är utlånad (den som syns i sambandstypen **Lånar**) strykas ur databasen. Skriv en trigger (med **create trigger**) som gör att det då automatiskt görs en notering i databasen om att boken varit utlånad (sambandstypen **Har lånat**).
