

Försättsblad till skriftlig tentamen vid Linköpings universitet



Datum för tentamen	2019-01-08
Sal (1)	<u>U1(4)</u>
Tid	8-12
Utb. kod	732G16
Modul	TEN1
Utb. kodnamn/benämning Modulnamn/benämning	Databaser: design och programmering Tentamen
Institution	IDA
Antal uppgifter som ingår i tentamen	10
Jour/Kursansvarig Ange vem som besöker salen	Eva Ragnemalm
Telefon under skrivtiden	070-190739
Besöker salen ca klockan	ca kl 10.00
Kursadministratör/kontaktperson (namn + tfnr + mailaddress)	Annelie Almquist, annelie.almquist@liu.se, tel 013-282934
Tillåtna hjälpmedel	Svensk - kinesisk, svensk-engelsk, svensk-norsk ordbok (vid behov)
Övrigt	
Antal exemplar i påsen	

Linköpings Universitet
Institutionen för datavetenskap
Eva Ragnemalm

TENTAMEN

732G16 Databaser Design och programmering

Datum: 2019-01-08 kl 8-12

Jourhavande lärare: Eva Ragnemalm

Tel: 070-1907391

Besöker tentamenslokalen: c:a kl 10

Hjälpmedel: Vid behov: ordbok för översättning från svenska till modersmål.

Följ instruktionerna på tentaomslaget. Svara inte på flera uppgifter på samma papper.

Uppgift	Poäng
1	6
2	2
3	1
4	1
5	4
6	6
7	6
8	20
9	6
10	16

Betyg: G: 40p VG: 60p Max: 78p

1. **Begrepp:** Redogör kort för (nämna minst en likhet och en skillnad): 6p
- a. funktionellt beroende och fullt funktionellt beroende
 - b. ER-modell och relationsmodell
 - c. relationsalgebra och SQL
2. **Begrepp:** Förklara kort vad ACID står för (förklara respektive term). 2p

Kombinationsuppgift (Design, Begrepp, Fysiska databasen, Transaktionshantering, SQL, Normalisering): följande innehåller information till uppgift 3-9.

Postorderfirman ListigaPrylar har beställt ett nytt datorsystem för att registrera beställningar, varor och kunder. Firman säljer speciellt användbara prylar av alla de slag och de finns i ett gammalt katalogsystem med artikelnummer, namn och beskrivning, och pris. Man vill också hålla reda på hur mycket man har i lager av respektive pryl (vara). Man behöver inte lagra information om gamla beställningar, utan systemet används bara för att plocka varor på lagret och skicka dem.

En klantig programmerare gjorde program som hanterade två filer med följande namn och innehåll;

Beställning (kundnr, namn, gatuadress, postnummer, postort, mobilnummer, artikelnr för beställd vara, varans pris, antal av varan)

Vara (artikelnr, namn, beskrivning, pris, lagerhylla, antal i lager)

Programmeraren tänker att en beställning gäller en viss pryl (vara) och vill samma kund beställa flera olika varor får de väl göra det i olika beställningar.

3. Hur ska programmeraren välja att identifiera olika poster i filen Beställning (dvs hur ska man välja primärnyckel i relationen Beställning) för att göra det möjligt att lagra flera beställningar från samma kund? 1p
4. Eftersom gammal beställningsinformation inte sparas så tas poster bort ur filen Beställning när försändelsen är skickad. Då uppstår en viss typ av problem, nämligen att kundens kundnummer, namn och adress försvinner ur filen när den sista beställningen levererats. Vad kallas detta problem (engelska eller svenska termen)? 1p
5. Programmeraren skapade ett program för dem som plockar på lagret, där Vara uppdateras med det förändrade antalet artiklar i lager när de plockat varan. Han skapade också ett separat program som användes vid leveranser av nya varor till firman, som uppdaterar Vara med antalet nya artiklar som inkommit. Programmen arbetade mot samma fil, Vara. Efter en tid började man misstänka att personalen på lagret slarvade, för ibland stämde inte antalet i lager med vad som verkligen fanns. Men personalen bedyrade sin ärlighet.
Vad tror du problemet var, och vad har en databashanterare som löser problemet? Beskriv kort: hur måste programmeraren skriva sitt program för att problemet ska undvikas? 4p

6. Antag att Beställning och Vara enligt ovan är tabeller i en relationsdatabashanterare som har SQL som databasspråk. 6p
- Skriv ett SQL-kommando som listar kundnummer, artikelnummer, beställt antal samt varans lagerhylla för alla beställningar från kunden med kundnr 123.
 - Skriv ett SQL-kommando som räknar ihop hur många varor av respektive sort (per artikelnr) som är beställda just nu, och listar artikelnummer och totalt antal beställda sådana varor under rubrikerna "Artikel" respektive "Antal".
7. Normalisera Beställning som om det var en relation, med den primärnyckel du identifierade i uppgift 2 ovan. Rita ut fulla funktionella beroenden och kandidatnycklar. Motivera sedan varje förändring eller brist på förändring utifrån normaliseringsvillkoren och var tydlig med vilka ffb och attribut som är relevanta i varje steg. Välj själv hur långt du vill normalisera och motivera ditt val. 6p
8. Antag att du inte får införa ett unikt beställningsnummer för beställningar och att en viss pryl (vara) kan finnas på flera lagerhyllor. Observera också att kundens gatuadress, postnummer och postort tillsammans utgör kundens adress. Gör en bättre design av ListigaPrylars databehov (dvs samma krav som i ovanstående beskrivning men undvik problemen som identifieras i tidigare uppgifter). Gör ett ER-diagram. Markera nycklar med understrykning, deltagande med enkel/dubbel linje och kardinalitet med 1/N (som vi gjort i kursen). 20p
9. Filen Varor blir med tiden ganska stor och man söker väldigt ofta i den på varans namn och inte artikelnumret. Programmeraren har hört talas om databashanterarnas index och undrar vilken typ han borde använda. Redogör för tre olika typer av index och hur de skulle kunna användas (fokusera på poststorlek och antal indexposter för de olika typerna). 6 p
10. **Relationsmodell:** Du har följande ER-diagram som representerar databehov för en leksaksgrossist som köper leksaker från tillverkare och levererar till leksaksaffärer. Konvertera det till relationsmodell. Markera primärnycklar med understrykning och främmande nycklar med pil från referensattributet/en till refererad relation (som vi gjort i kursen). Namnge relationer och attribut så att den som rättar begriper var de kommer ifrån. 16 p

Lycka till!

