

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

Tentamen i

Databasteknik

för D1 m fl

tisdag 15 januari 2019

Gäller som tentamen för:
DT105G Databasteknik, provkod 0100

-
- Hjälpmedel:** Ordbok för översättning.
- Poängkrav:** Maximal poäng är 31. För godkänt betyg krävs 20 poäng, varav minst fyra poäng på uppgift 1 och minst fem poäng på uppgift 2. För den som följt kursen hösten 2018 ger varje i tid inlämnad inlämningsuppgift en extra poäng. Den som *inte* gått kursen hösten 2018 får dessa (fem) extrapoäng ändå.
- Resultat:** Meddelas på kursens hemsida eller via e-post senast tisdag 5 februari 2019.
- Återlämning av tentor:** Elektroniskt via Studenttjänster.
- 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 och tvetydiga 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

Här är ett par varuautomater som står utanför kafeterian i Teknikhuset:



De har varit trasiga åtminstone sen i somras, och Sodexo, som äger automaterna, har inte lagat dem. Kanske har de glömt var de ställt dem, och kan inte hitta dem? Vi måste hjälpa dem genom att skapa en databas!

Det som Sodexo vill lagra i databasen är följande:

1. **Kunder.** Kunderna är de företag och andra organisationer där Sodexo har placerat ut automaterna. Varje kund har ett **namn**, till exempel **Örebro universitet**, och ett unikt **kundnummer**, till exempel **83712**.
2. **Automater.** Varje automat har ett unikt **nummer**, till exempel **602827**, och den står hos en av kunderna.
3. **Varor** som finns i automaterna. Varje vara har ett unikt **namn**, till exempel **Coke Zero, 50 cl**, och även ett unikt **varunummer**, till exempel **1283798**.
4. Automaterna **innehåller** varor. Varje automat kan förstås innehålla flera varor, och varje vara kan förstås finnas i flera automater.
5. **Köp** av varor ur automaterna. Ett köp gäller en **vara** ur en **automat**, och görs en viss **tid** på ett visst **datum**.

Uppgift 1 (5 p)

Rita ett ER- eller EER-diagram för den beskrivna databasen. Använd informationen i scenariot ovan, men tänk också på att det ska gå att svara på frågorna i uppgift 3 nedan.

ER- och EER-diagram kan ritas på flera olika sätt. Om du använder en annan notation än kursboken, måste du förklara den notation som du använder.

Uppgift 2 (6 p)

Implementera den beskrivna databasen i relationsmodellen, dvs översätt ER-diagrammet till tabeller.

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 3 (10 p)

Formulera följande frågor i SQL. Definiera gärna vyer eller CTE:er om det underlättar, men skapa inte nya tabeller.

- (2p) Vad är numren på de automater som är placerade hos kunden **Örebro universitet**?
- (2p) Hur många *olika* varor finns i automaterna som står hos kunden **USÖ**? Vi vill veta antalet, inte numren.
- (2p) Hos vilka kunder står det inga automater? För varje sådan kund vill vi veta kundens nummer och namn.
- (3p) Skapa en vy som heter **AntalKöp** och som för varje kund anger kundens nummer, kundens namn och antalet köp som gjorts ut automater som står hos den kunden. Även kunder där inga köp gjorts ska vara med i vyn, med antalet noll.
- (1p) Använd vyn från uppgiften ovan för att söka reda på hos vilken kund det har gjorts flest köp. Vi vill veta namnet på den kunden.

Uppgift 4 (5 p)

Sodexo vill ha en ruta på sin hemsida med namnet på deras bästa kund, alltså den kund där det gjorts flest köp. Det är alltså svaret på SQL-frågan i deluppgift **e** i uppgiften ovan.

De gör en lösning där den SQL-frågan körs varje gång någon tittar på Sodexos hemsida. Frågan tar lång tid att köra, och allt blir väldigt långsamt.

Beskriv några bra sätt som Sodexo kan använda för att få bättre prestanda!

Uppgift 5 (5 p)

När man programmerar numera är det populärast med **objektorienterad programmering**. Samtidigt är det **relationsdatabaser** som är den vanligaste typen av databaser. Om man har ett objektorienterat program, och vill lagra det programmets objekt i en databas, brukar man därför behöva **översätta** från programmets objekt till databasens tabeller.

Ett objektorienterat program innehåller följande klasser, alltså typer av objekt:

- **Fordon**. Varje fordon har ett unikt registreringsnummer (till exempel **OAU686**), en vikt (till exempel **2600 kg**) och en färg (till exempel **röd**).
- Varje fordon kan vara antingen ett motorfordon eller ett omotoriserat fordon. Klassen **Fordon** har därför två subclasser, **Motorfordon** och (kan vi kalla dem) **FordonUtanMotor**. Motorfordon har, förutom samma egenskaper som andra fordon, även en **motorstyrka**, till exempel **800 hästkrafter**.
- Klassen **Motorfordon** är i sin tur uppdelad i två subclasser: **Bilar** och **Motorcyklar**.
- Klassen **Bilar** är i sin tur uppdelad i tre subclasser: **Lastbilar**, **Personbilar** och **Bussar**. Lastbilar har en maxlast (till exempel **10000 kg**), och bussar har ett antal platser (till exempel **20**).

a) (2p) Rita ett EER-diagram som innehåller entitetstyperna ovan, med arv och egenskaper.

b) (2p) Översätt EER-diagrammet till tabeller. 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 vad som är primärnyckel, och vilka referensattribut som finns och vad de refererar till.

c) (1p) Skriv en SQL-fråga som tar fram vikten på alla röda bussar.
