

Projekt 2 för DB:HT2018 - Frågespråk

Läs igenom hela detta dokument noggrant innan ni påbörjar arbetet med Projekt 2.

1. Allmänt

Ett antal frågor ska besvaras med hjälp av SQL och relationsalgebra. SQL-delen ska skrivas i MS Access 2016 (OBS: versionen!) och ni använder en databas som är en implementation av en given relationsmodell. Databasfilen hämtas på kurswebbplatsen (se nedan) och är populärad med testdata som **inte** ska ändras. *Relationsalgebradelen* ska skrivas i text, och mot samma relationsmodell.

2. Inlämning:

Allra sista inlämningsdag 2019-01-19, 23:59, dvs delkurstillfällets sista dag.

Det som står nedan gäller även eventuella restinlämningar.

Inlämningen ska *skickas till Slutinlämning: Projekt 2 - XX grupper på kurswebbplatsen som en zippad fil*, där XX står för er redovisningshandledare. (För ev. restinlämningar gäller Restinlämning Y: Projekt 2 - XX grupper, där Y står för vilken restinlämning det är, dvs 1, 2, 3 osv.) (Skicka endast dit och *inte* med kopia till någon lärare.)

Följande ska finnas med:

- .accdb-fil, dvs er kopia av den givna databasfilen, kompletterad med er grupps SQL-frågor
- .pdf-fil med frågorna formulerade dels i relationsalgebra (för fråga 1- 5), dels i SQL, plus två skärmdumpar av en körning av varje fråga. Den ena skärmdumpen skall vara från ACCESS-körningen, den andra från SQLtutor-körningen. SQLtutor (<https://sqltutor.dsv.su.se/prod/>) är ett webbaserat verktyg som utför extra kontroll av SQL-frågor från Projekt 2. Verktöget finner ni på adressen ovan. Gruppera ihop frågan i naturligt språk med sin översättning till relationsalgebra och SQL plus skärmdumparna av körningarna (Access resp. SQLtutor) så att allt står på samma ställe. Har ni kommentarer så ange dem också per fråga. I denna fil ska även alla gruppmedlemmars namn plus epostadress anges.

Namnge filerna enligt nedan.

Namngevningssmall: exempel för en grupp 07:

- zip-filen db_HT2018_Proj2_grupp07.zip
- accdb-filen: db_HT2018_Proj2_grupp07.accdb
- pdf-filen: db_HT2018_Proj2_grupp07.pdf

3. Förberedelser

Innan ni börjar ska alla ha läst kap. 1-5 och 9 i kompendiet "Introduction to Access", och kört igenom övningarna. Allt kan göras gemensamt. Det kan också vara bra att kursivt läsa igenom kap. 6-7 i kurskompendiet, bok, föreläsningsbilder, lektionslösningförslag, etc.

4. Beskrivning av uppgiften

Ni ska med hjälp av frågespråken SQL och relationsalgebra besvara ett antal frågor mot en fördefinierad databas skapad efter en given relationsmodell. Databasen är populär med tillräcklig och väl varierad testdata, på ett sådant sätt att svaren illustrerar skillnaden mellan olika frågetyper och operatörer i SQL. Fokus i denna uppgift är inte att modellera information, eller lägga in information i en databas, utan uppgiften går helt ut på att skriva frågor som söker fram information ur en databas, utan att behöva ta hänsyn till att olika sätt att modellera en och samma domän gör olika typer av frågor mer eller mindre svåra att besvara. Databasen är modellerad och populär så att alla frågetyper ska kunna jämföras, och resultat på frågor besvarade med samma frågetyp ska kunna jämföras. Man ska få ut intressanta (dvs icke-tomma) resultat på sina frågor, och inte felaktigt få samma rader som resultat när man använt olika operatörer på grund av att databasen innehåller för lite testdata eller testdata med för lite variation.

Ni ska arbeta i MS Access 2016, med en databasfil som hämtas från kurswebbplatsen: *Databas för Projekt 2* (finns i blocket för Projekt 2). Filen heter db_ht2018_Proj2_gruppXX.accdb. När ni sparar ner filen ska ni byta ut XX mot ert gruppnummer, enligt mallen ovan. Varje grupp har ett antal frågor att besvara (de följer här nedan). **De fem första frågorna ska besvaras med både SQL och relationsalgebra, övriga frågor bara med SQL** (gruppen kan frivilligt besvara även övriga frågorna med relationsalgebra, och får då kommentarer till dessa i rättningsrapporten). SQL-lösningarna ska sparas i databasfilen, och vara testkörda mot databasen (i både ACCESS och SQLtutor) så att ni kan verifiera att de fungerar som ni avsett. Numrera lösningarna i enlighet med uppgiftstexten. Frågorna är valda för att illustrera olika operatörer och frågetyper i SQL och relationsalgebra, t ex att öva på selektion, projektion, olika typer av join, hur DISTINCT-operatören och dubblettrader fungerar i relationsalgebra jämfört med i SQL, unionskompatibla operatörer osv. Ingen grupp får exakt samma frågemängd, utan får varianter på samma frågetyper. Det kan t.ex. röra sig om samma frågetyp ställd mot olika tabeller, eller med avseende på olika kolumner. OBS: ni ska INTE använda syntaxen med INNER JOIN i SQL utan använda samma syntax för JOIN som gått igenom på föreläsningarna – OUTER JOIN kan användas om och när denna operatör behövs.

SQL-projektet ska besvaras genom att skriva SQL-satser (dvs queries) till databasen. Plus att dokumentera dessa textuellt (i en .pdf-fil) tillsammans med frågelydelsen och skärmdumpar av exekvering av frågan (i Access resp. SQLtutor). För att få övning i SQL ska ni skriva er kod i SQL-fönstret (SQL view), och alltså inte använda det grafiska SQL-verktyget, som dessutom är begränsat och inte kan hantera en del komplexa SQL-satser. Enklare frågor kan normalt lösas med en enda SQL-sats. För komplexa frågor kan det dock ibland vara lämpligt att använda ytterligare någon SQL-sats (dvs query) som vy (view). Poängen är att man då bryter ner den komplexa satsen till mindre satser, och kan sedan enklare kontrollera att varje del (sats/query/view) returnerar förväntat svar innan man skriver en huvudsats som då använder delarna som om de vore tabeller i databasen (tabellerna i databasen kallas f.ö. bastabeller). Följ uppgiftstexten noggrant, bl.a. ska *endast* de kolumner som anges i respektive fråga visas i resultaten, inga andra! Numrera varje SQL-sats tydligt och entydigt när ni sparar, så att rättaren enkelt ser vilken uppgift som avses.

Relationsalgebradelen (dvs fråga 1-5 och övriga frågor frivilligt) redovisas genom tydliga skriftliga lösningar som sedan sammanställs och sparas i den ovan nämnda .pdf-filen. Vi rekommenderar starkt att ni använder de operatorstecken (grekiska symboler) vi använt under undervisningen. Om man inte vill använda de grekiska symbolerna, är det tillåtet att t ex skriva PROJEKTION eller PI istället för π , SELEKTION eller SIGMA istället för σ , och så vidare för alla övriga operatörer, övrig syntax för relationsalgebran måste dock beaktas.

*Uppgifter markerade med * är icke obligatoriska (men försök att lösa dem ändå).

Frågor för grupp 11

- 1 F Ta fram namn, adress, postnummer och telefon på personer som bor i Kista!
- 2 K Ta fram kurskod, namn och pris på kurser på 3 eller 4 veckor!
- 3 C Ta fram personnummer på lärare som har undervisat i rummet med id 3!
- 4 H Ta fram personnummer, namn och ort på lärare som har undervisat i rummet med id 1!
- 5 C Ta fram personnummer, namn och tjänsterum på lärare som har undervisat i rummet med id 3!
- 6 D Ta fram personnummer på studenter som har deltagit i något kurstillfälle i Sirius!
- 7 E Ta fram namn, adress, ort och telefon på studenter som har deltagit i något kurstillfälle i Jupiter!
- 8 B Ta fram antal kurstillfällen som varje rum har använts till! Visa rummets namn, rummets kapacitet och antalet!
- 9 D I. Ta fram namn och kapacitet samt antal kurstillfällen per rum! II. Ta fram namn och kapacitet på det rum som har använts flest gånger! Använd lösningen av I (vyn) som källa i II!
- 10 E Ta fram namn och kapacitet på rum som har använts för kurstillfällen av minst tre OLIKA kurser!
- 11 B Ta fram namn och ort på studenter som inte deltog i något kurstillfälle med startdatum under 2010!
- 12 D Ta fram namn och kapacitet på rum som aldrig har använts av läraren Bo Åkerman!
- 13 E Ta fram namn och kapacitet på rum som har använts för kursen "Logik" eller för kursen "Java, fortsättningskurs" (eller för båda)!
- 14 E Ta fram namn och kapacitet på rum som har använts för både kursen "Logik" och kursen "Java, grundkurs"!
- 15 B *Ta fram namn och kapacitet på rum som har använts av samtliga lärare från Södertälje!
- 16 C *Ta fram antal gånger som varje student har deltagit i kursen "Java, grundkurs"! Visa studentens personnummer, namn och ort, samt antalet! Även studenter som inte har deltagit i den kursen skall ingå i resultatet (med 0 eller NULL i antalkolumnen).
- 17 C Ta fram information om kurstillfällen som ingen funktionshindrad student har deltagit i! Visa kurskod, kursnamn, startdatum, rummets namn och lärarens namn!