

# DATABAS RAPPORT

By Denisé Larsson

---

## Modeller :

Analysera kravet och notera vilka objekt jag behöver. Objekt som jag valde var:

- Ort
- Utbildning
- Klass (Utbildningsledare tar hand om en klass)
- Anställd (Utbildningsledare och lärare, istället för 2 tabeller)
- Utbildningskurs (Blev en relations tabell, mellan Utbildning och Kurs)
- Kurs
- KlassKurs (Lärare har hand om en klass kurs)
- Studerande
- Betyg (relations tabell mellan klasskurs och studerande)
- Inaktiv studerande (när en elev skrivs ut, ska det föras över hit)
- Inaktiv betyg (här tänkte jag att man inte behöver klarat en hel utbildning för att för att få ett slutbetyg, därför blev det inaktiv betyg som kan hämtas när eleven inte är aktiv längre)

När objektet är framtagen skapade jag en infologisk modell för att kolla vilka relationer dem har till varandra och även kunna visa upp detta för kunden om det var så dem ville ha det. Fick feedback från kunden och fick små justera, därefter normaliserade jag för att se att det inte har några dubbel lagringar fick då lägga till några extra kolumner som till exempel datum till "InAktivBetyg" tabellen etc.

## PROCEDURE:

Jag valde att lägga till tabell med inaktiva studeranden och inaktiv betyg. Eftersom kravet var att man ska kunna automatiskt överföra elevernas information efter avslutat utbildning till en tabell och hämtar betyg när en elev inte längre är aktiv blev de 2 Procedure metoder på detta.

## CONSTRAINT:

Jag har använt dessa constraint (*PRIMARY KEY*, *FOREIGN KEY*, *NOT NULL*) när jag skapade tabellerna för att vara säker att alla primärnycklar och främmande nycklar knyter ihop. Jag hade kunnat göra det med ALTER TABLE men för att vara säker och inte glömma det så har jag gjort det här. Men när jag skulle INSERT data fick jag lägga till "*SET IDENTITY INSERT*" Studerande ON/OFF" på varje tabell förre/efter, eftersom jag senare valde att ge id nummer till vissa tabeller som hade "Id INT IDENTITY(1,1) PRIMARY KEY". Använde en CHECK för att kolla så att Anställd Id är mellan 1-1000.

**TRIGGERS:**

Jag valde att göra en trigger på när man lägger till ett betyg eller ändra så uppdateras datumet automatiskt. Var svårt och komma på något bra här, tänkte först att man kunde använda trigger till flytta elever som har avslutat en utbildning men det blev en procedurer där istället.

**VYER:** Här skapade jag en vy som visar lärareId, KursId, start/ slut datum, betyg och vilken elev. Eftersom det också var med i kravet gjorde jag en vy som visar dessa information.

**JOIN, GROUP, BY, HAVING m.m:**

Valde att visa hur många som studera i en utbildning, vilka kurser och betyg som dem har fått visas.

**Sammanfattning:**

Under projektets gång har den största biten varit att skapa den infologiska modellen och sätta upp rätt data. Har fått rita många infologiska modell för att säkerställa att det blir så bra som möjligt i början. När det kommer till att lägga in data gjordes mycket i början innan db byggdes, märkte även här att jag kunde gjort insert i tabellerna i efterhand när jag behövde en viss data istället för att dropa db och köra den om och om igen efter jag korrigerade data texten.

Jag tror att det är viktigt och analysera korrekt i början annars blir det mycket extra jobb när man väl har skapat databasen. Jag har också valt att lägga in mycket data för att inte begränsa testningen allt för mycket. Fick ändra vissa relationer mellan olika tabeller ett par gånger för att få rätt på data.

När det kommer till att skapa procedure, trigger, vyer, group by etc fick jag testa mig fram och valde utifrån kravet. Hade kunnat skapa hur mycket som helst men kravet är alltid det viktigaste och får med, allt annat görs om det finns tid över.

Koppling mellan databas och java gick smidigt, kul att se kopplingen och vad det faktiskt gör. Ser verkligen fram emot kommande kurser :)

**Lärdomar:** Det har varit mycket error och hitta rätt kombinationer, men efter alla fel man har gjort vet jag förhoppningsvis när man är ute och jobba vad jag inte borde göra. Många ändringar under projektets gång och många sql kod fick slängas fast jag hade suttit med det länge, det var många gånger lättare och ta bort och skriva om än att försöka lägga till en massa onödigt för att få det och funka.

