

Referentiell Integritet RI, (Enforce Foreign Key Constraints)

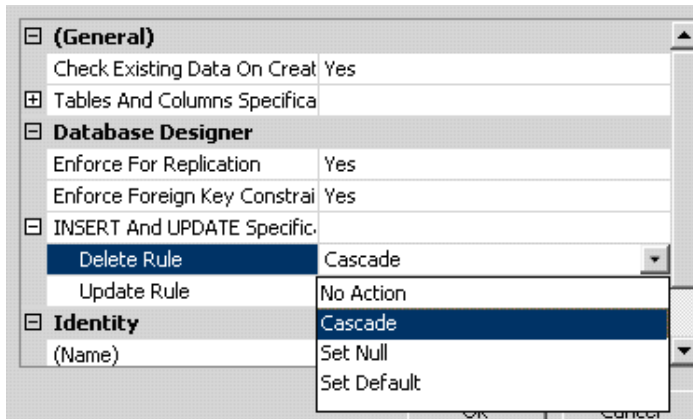
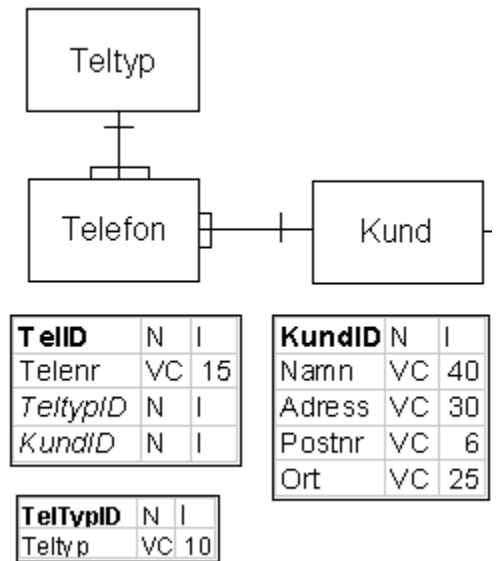
RI är ett verktyg som du kan använda i de flesta databaser för att definiera hur relationer mellan tabeller ska fungera. Det är alltså relationen mellan två tabeller som du påverkar. Funktionen med Referentiell Integritet kan vara, och är oftast, olika i olika databashanterare.

Antag att du har följande utformning av tabeller enligt bilden.

Man kan ha en relation mellan exempelvis Kund och Telefon där Kundid (Fk) i Telefon inte är något annat än ett vanligt fält och inga övriga bestämningar har gjorts på fältet. Då kan du inte använda något regelverk. Se också beskrivning på index för detta.

I MSSQL kan du bestämma att fältet KundID ska vara en Fk till Kundid i Kund. I den bestämningen skapas ett index i Telefon på Kundid bland annat därför att det ska gå snabbt att hitta de relaterade posterna i Telefon.

Funktionerna som du har tillgång till med RI i MSSQL är då följande:



No action innebär att du inte kan radera en post i föräldertabellen utan att du först har raderat alla poster i barntabellen. Du kan heller inte lägga till en post i barntabellen utan att motsvarande förälder finns i föräldertabellen..

Cascade innebär att om du raderar en post i föräldertabellen så raderas alla poster i barntabellen.

Set Null innebär att om du raderar en post i föräldertabellen så läggs Nullvärde in Fk-fältet på alla poster i barntabellen. Då måste du också ange att Null är tillåtet i fältet.

Set Default innebär att det värde som finns som Default värde för fältet kommer att sättas vid radering av förälderpost.

Delete Rule

Om du väljer No action på din Fk Telefon-Kund så innebär det att du kan inte lägga in en telefon utan att motsvarande kund finns i Kundtabellen. Alltså måste du lägga in ett värde på Kundid och det värdet måste finnas i Kundid (Pk) på Kund. Du kan heller inte radera en Kund som har telefoner. Du måste då först radera telefonerna på den kunden - sen kan du ta bort kunden.

Väljer du Cascade så har det samma funktioner men om du raderar kunden så kommer databasen att radera de telefon som är registrerade på den kunden. Alltså som har samma Kundid som posten i Kund.

Update Rule

Update Rule har *med ändring/uppdatering av Pk och Fk i relationen*. Det har alltså inget att göra med ändringar på namn, adress eller telefonnummer att göra.

Betydelsen av Update Rule har att göra med vad som ska hända om du ändrar Kundid i Kundtabellen och hur det ska påverka Telefontabellen.

Om du väljer No action på din Fk Telefon-Kund så innebär det att du inte kan ändra Kundid i Kundtabellen. Du kan ändra i telefon men bara till ett annat Kundid som finns i Kund. Om du inte har telefoner på en kund så kan du ändra på kunden och det påverkar då inte telefontabellen.

Väljer du Cascade så kan du ändra ett Kundid i Kund och databasen kommer då automatiskt att ändra Kundid på alla relaterade poster i Telefon.

Regel: Använd No action om du har räknare (Identity). Du kommer då aldrig att själv ändra Kundid i tabellen. Om du har talande nyckel som Pk, ett organisationsnr, namn eller annan liknande Pk så kan det bli så att du vill ändra värdet på Pk. Då ska du ha Cascade.

I Relationen mellan Kund-> Telefon bör man då sätta Cascade på Delete Rule och No Action på Update Rule.

I relationen mellan Teltyp -> Telefon då. Ja för det första ska en telefon inte läggas in utan att man anger vad det är för teltyp. Då bör det vara en korrekt Fk med no action/cascade. Vad sk då hända om man ska radera en teltyp. Ska alla telefoner raderas också som har den teltypen? **NEJ!!!!** Så får det inte vara. Om man tar bort mobil som typ så försvinner alla telefonnummer som är mobiler. Det måste då vara No action i den relationen. Om man ska ta bort en teltyp så måste man först lägga om alla de telefonnummer som använder den typen till en annan typ. Eller enklast. Döp om den teltypen som ska byta namn. Då kommer alla som hade ursprungstypen att få den nya teltypen.

Om du har satt No action som du gör i relationen Kund-Kategori och det sker efter det att det finns ett antal poster i Kund. Då måste du först lägga in data i Kundtabellen för de kategorier som finns i Kategori – annars kan du inte sätta dit en Fk. I lab 2 skapar du kategori med 1, 2 eller 3 som Pk i Kategori. Då måste du ha 1, 2 eller 3 som värde i kategori i Kundtabellen. Annars kan du aldrig bygga en Fk.