



## Lär känna MS SQL 2008 / Övning

---



### Observera

Övningar som finns tillgängliga är till för att du ska kunna testa dina kunskaper och träna på dem. Det är helt upp till dig när du vill genomföra och om du vill genomföra övningarna. Observera att du vid ett visst tillfälle måste redovisa dina kunskaper (i avsnittet som denna övning ingår i) muntligt för en handledare/lärare och då klarar du dig bäst om du bearbetat alla övningarna.



### Tips

Se kurshemsidan för användbara länkar/referenser för att genomföra denna övning.



### Förberedelse

Innan du påbörjar arbetet bör du se till att du läst igenom de eventuella föreläsningsanteckningar, bokhänvisningar, föreläsningsmaterial och länkar som finns tillgängliga till detta avsnitt.

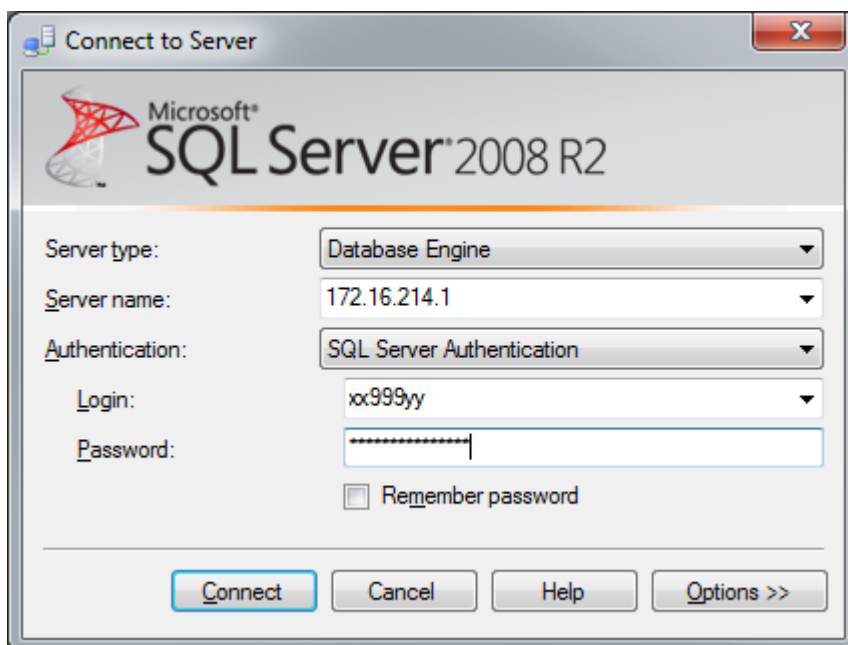
**Läs genom hela övningen innan du börjar arbeta med databasen.**

**Denna övning är inte obligatorisk.**

## Lär känna MS SQL Server 2008. En första övning.

Anslut till MS SQL Server 2008 som finns i den virtuella miljön. Falken har IP nr 172.16.214.1

1. Starta din koppling VPN till det virtuella nätet vpn200.lnu.se.
2. Starta programmet MS SQL Server 2008 Management Studio
3. Ändra **Server name**: till IP nummer enligt bilden.
4. Ändra **Authentication** till SQL Server Authentication
5. Lägg in **ditt** kontonamn i fältet Login. Du kan behöva lägga in domännamnet ex: Kupan\xx999yy (dvs ditt kontonamn).  
I fältet password anger du det lösenord du har tilldelats. Se kurswebben om du är osäker.
6. Avsluta med att välja **Connect**.

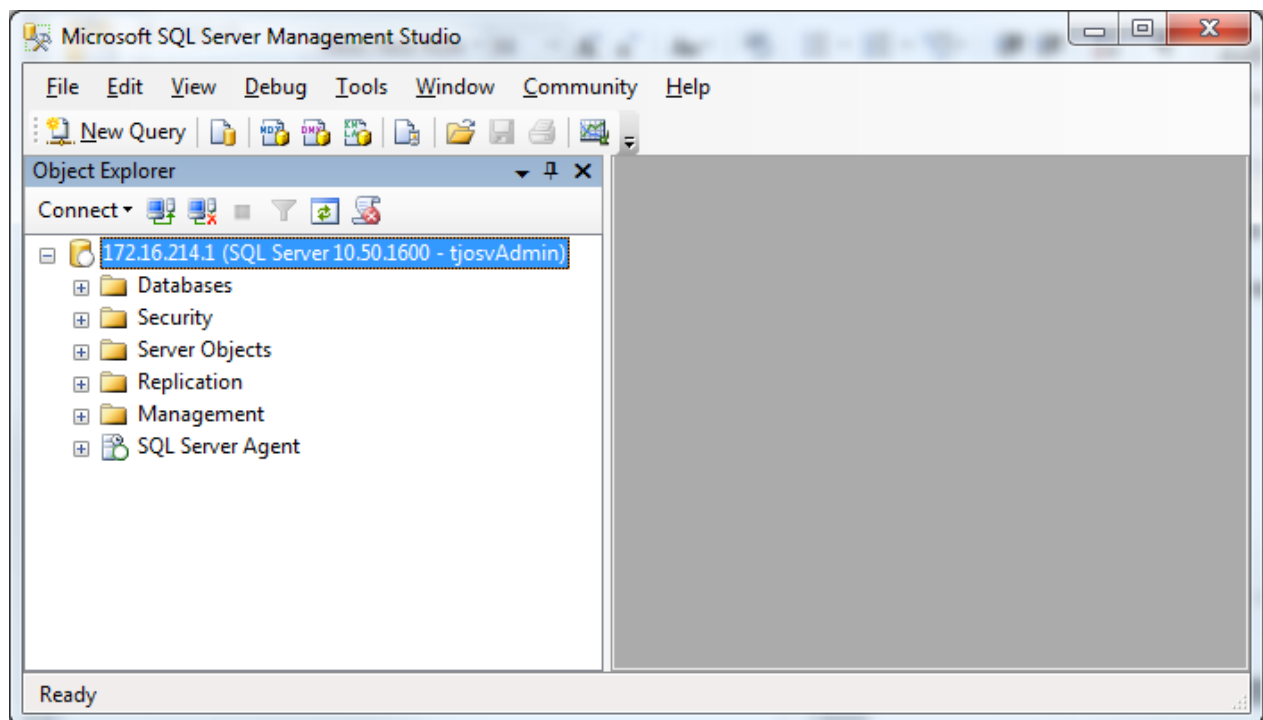


### OBS!

Om du använder MS SQL Server 2008 Management Studio Express hemma och ansluter till Viggen så kan det innebära vissa problem men det mesta bör gå bra när du kör mot MS SQL Server 2008 på Viggen.

Om du har Express eller Compactversionen installerad hemma och vill installera en högre version av MS SQL Server 2008 ska du först avinstallera Express/Compact versionen.

När sedan MS SQL Server 2008 Management Studio startat så ser det ut på ungefär följande sätt:



## Namnsättning på din databas

För att vi lätt ska hitta igen de databaser som du skapar och för att din databas inte ska kollidera med namnen på andra databaser som finns i MS SQL så vill vi att du använder ett specifikt system för namnsättning på dina databaser:

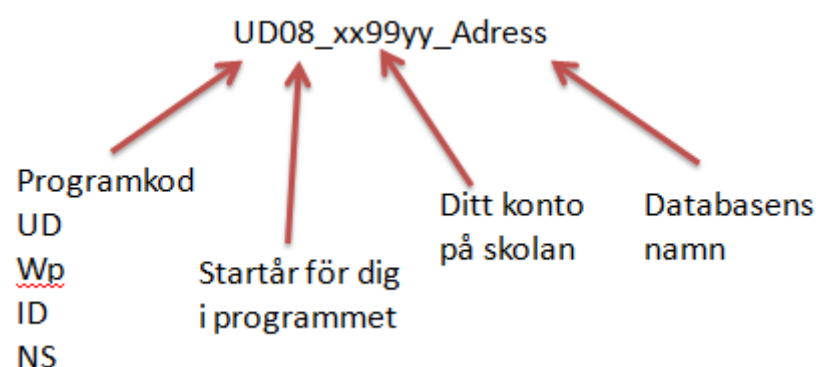
Börja alltid med förkortningen på det program du går: ID, UD WP, IT etc. Lägg sedan på det startår som du har när du började på programmet. Exempelvis UD12, WP12 eller fristående. Lägg sedan dit ditt konto och databasens namn Exempelvis:

**UD12\_xx99yy\_Address**

**WP12\_xx99yy\_Address**

**FRI12\_xx99yy\_Address**

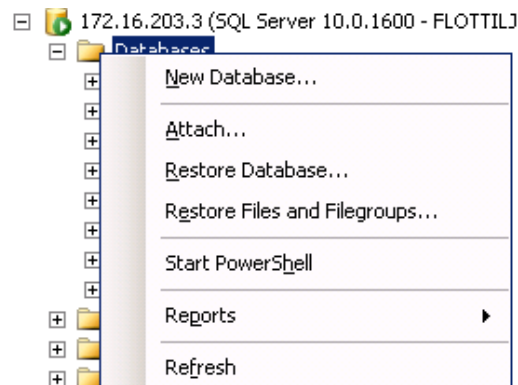
Byt ut xx99yy mot ditt kontonamn.



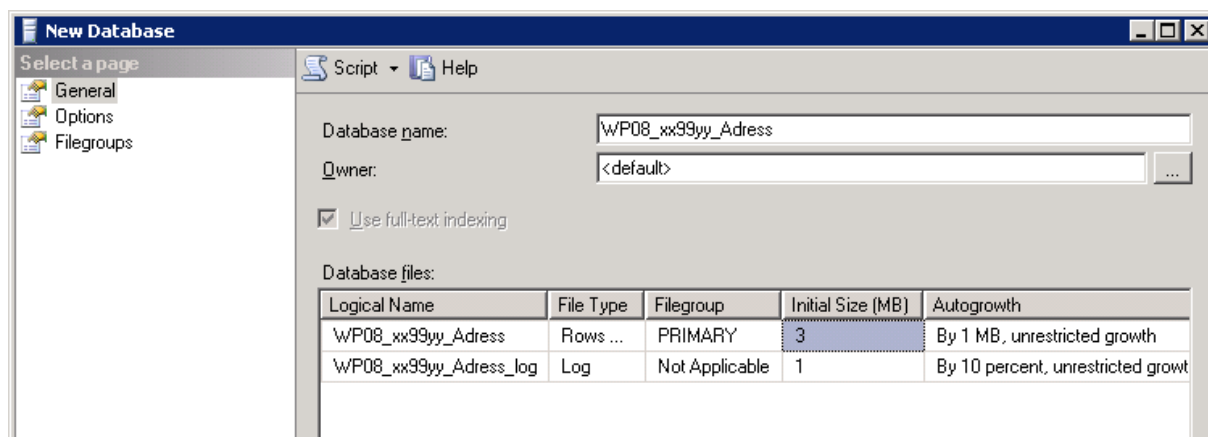
# Skapa Databas

Din första uppgift är att skapa en enkel databas i utbildningssyfte för att bekanta dig lite med MS SQL Server. Databasen kommer att innehålla två tabeller. Se nedan. Vi kallar databasen för Adress eftersom det ska läggas in Adressuppgifter i den.

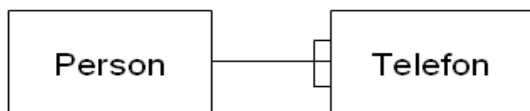
1. Högerklicka på Databases och välj New Database.
2. Ange namnet på databasen. Ange din kod för programmet, lägg till startåret, därefter lägger du på ditt konto och slutligen databasens namn: Exempelvis:  
**WP07\_xx99yy\_Adress.**



**OBS!** Byt ut xx99yy mot ditt kontonamn. Som du ser skapas två filer. En data fil och en logfil.



4. Avsluta med att välja **OK** nederst i bilden så skapas databasen. Du hittar den nya databasen längst ned i objektslistan ( i vänstra delen av Management Studio). Om inte Högerklicka välj Refresh.
5. Nästa steg är att skapa tabellerna (vilket du gör i nästa punkt) som du ska använda dig av . Så här ser datamodellen ut:



Den första tabellen du kommer att skapa ska innehålla ett antal personer som du har kontakt med. Tabellen vill vi kalla för Person och ska ligga i databasen. Vi har tänkt att följande fält ska ingå i tabellen:

Fältnamn	Innehåll	Fältlängd	Fälttyp	
Pid	Primärnyckel	Räknare	Heltal	Int
Enamn	Efternamn	25 tkn	Text	VarChar
Fnamn	Förnamn	20 tkn	Text	VarChar
Adress	Gatuadress	45 tkn	Text	VarChar
Postnr	Postnumret	6 tkn	Text	VarChar
Ort	Ortsnamnet	25 tkn	Text	VarChar
Anteckning	Anteckningar	200 tkn Text	VarChar	

6. Expandera den databas som du vill skapa den nya tabellen i. Högerklicka sedan på Table-objektet och välj New Table.
7. Skriv in följande:

	Column Name	Data Type
	PID	int

8. Klicka på Nyckeln när raden med PID fortfarande är markerad. Du skapar då en Primary Key på det fältet.



Klicka på Nyckeln

9. Bläddra ned till Column Properties och ställ in Identity Specifikation enligt nedan. Observera vad som händer med Identity Increment och Identity Seed när du väljer Yes på Is Identity.


Column Properties	
Collation	<database default>
Computed Column Specification	
Condensed Data Type	Int
Description	
Deterministic	Yes
DTS-published	No
Full-text Specification	No
Has Non-SQL Server Subscriber	No
Identity Specification	Yes
(Is Identity)	Yes
Identity Increment	1
Identity Seed	1

10. Fortsätt och lägg in de andra fälten enligt följande bild:

	Column Name	Data Type	Allow Null
🔑	PID	int	<input type="checkbox"/>
	Enamn	varchar(25)	<input type="checkbox"/>
	Fnamn	varchar(20)	<input type="checkbox"/>
	Adress	varchar(45)	<input type="checkbox"/>
	Postnr	varchar(6)	<input type="checkbox"/>
	Ort	varchar(25)	<input type="checkbox"/>
	Anteckning	varchar(200)	<input type="checkbox"/>

11. Behöver du ändra ett fält så markerar du fältet och gör de ändringar som du anser behövs göras
12. Nu är det dags att spara tabellen och namnge den. Stäng tabellen så får du också ange namnet på tabellen. Använd namnet Person på tabellen. Det blir då med automatik dbo.Person
13. Skapa en ny tabell med namnet Telefon som har följande utformning:

Fältnamn	Innehåll	Fältlängd	Fälttyp
TelID	Primärnyckel	Räknare	Heltal
Telnr	Hela telenr	20 tkn	Text
Pid	Fk	Tal	Heltal

	Column Name	Data Type	Allow Nulls
	TelID	int	<input type="checkbox"/>
	Telnr	varchar(20)	<input type="checkbox"/>
	PID	int	<input type="checkbox"/>

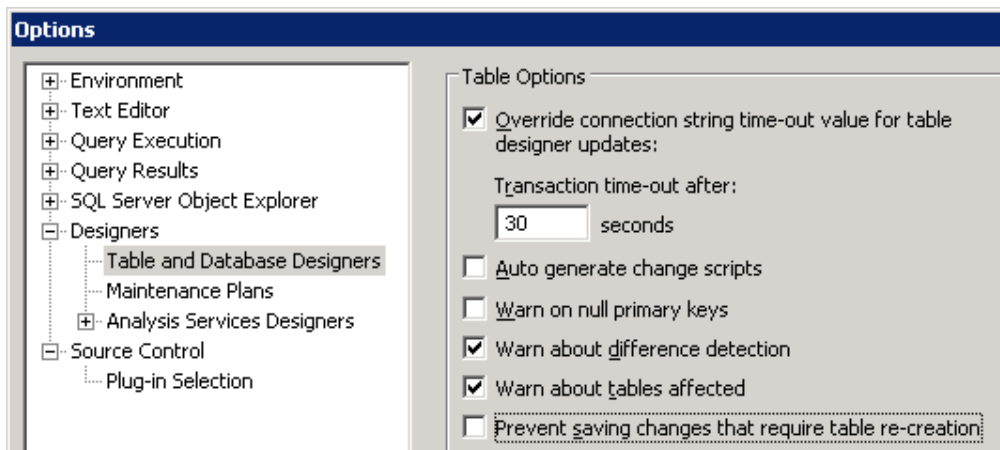
Telid är primärnyckel för tabellen.

**Se till att du har ställt Identity på fältet (Finns under Column Properties!!)**

Pid är främmande nyckel för koppling till person. En person kan ha flera olika telefoner. Det är viktigt att fältet Pid är Int efter som Pid i person är Int. Pk och Fk måste vara samma.

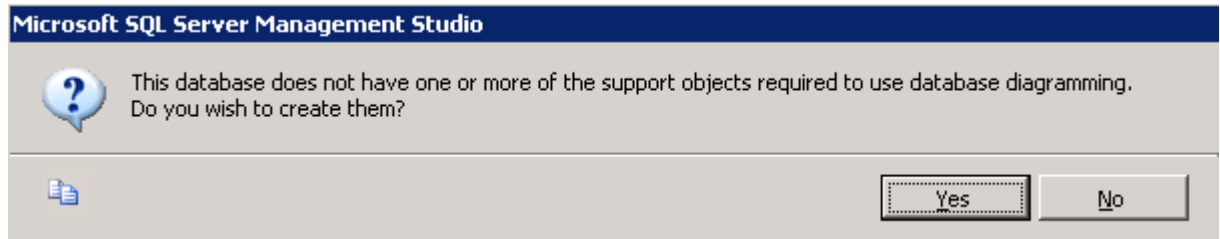
Telnr är fältet där du matar in aktuella telefonnr.

14. Spara tabellen under namnet Telefon. Stäng tabellen.  
OBS! Om du ska ändra i någon av tabellerna är de bara att öppna tabellen i designläge och genomföra ändring. Kontrollera dock inställningen under Tools-Options så att Prevent saving.. är avböckad. Annars kan du inte spara en ändring av tabellkonstruktionen:



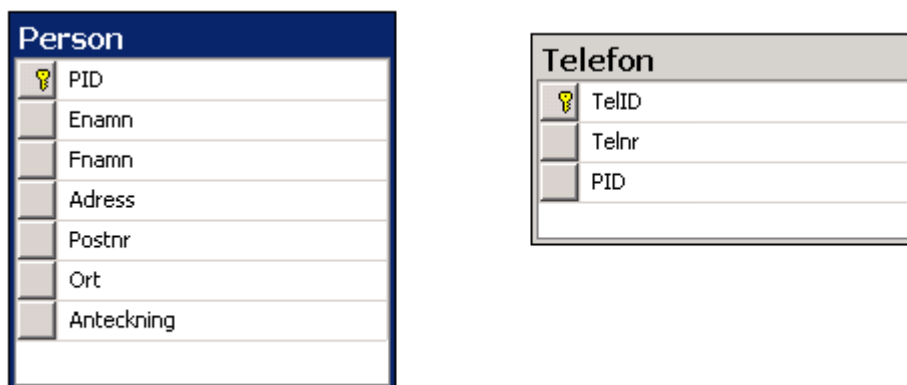
15. Mellan tabellen Person och Telefon finns fältet Pid i Telefon som ska hålla samman de båda tabellerna och som sedan ska visas vilka telefonnummer en viss person har. Det du nu ska göra är att du ska skapa en Fk i Fältet Pid i Telefon.
16. Markera Database Diagrams i din databas. Följande fråga kan då dyka upp och du väljer då Yes.

17.

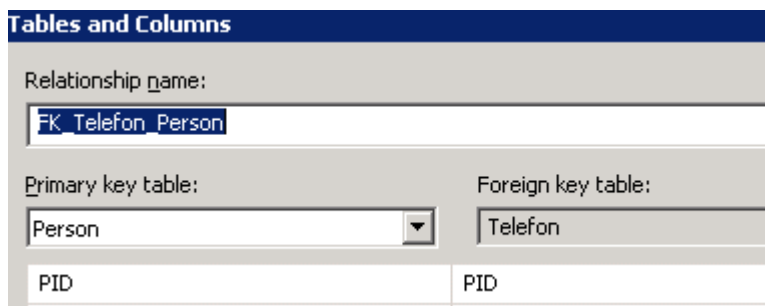


18. Högerklicka på Database Diagrams och välj New diagram

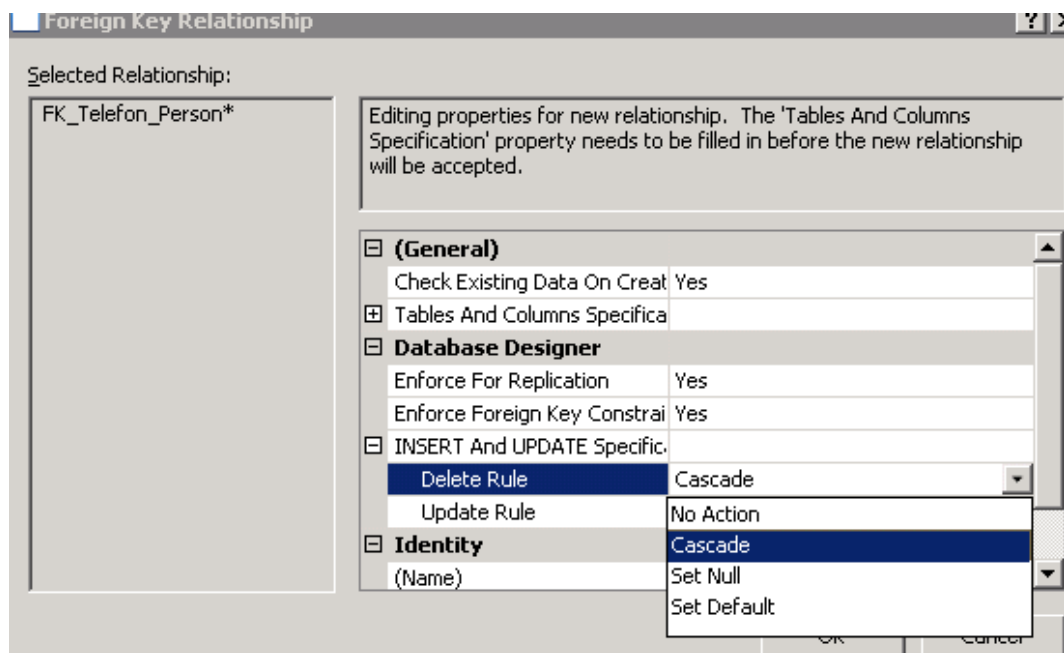
19. Markera Person och välj Add.  
Markera Telefon och Välj Add  
Välj Close och du har följande utseende.



20. Peka på PID i Telefon och dra den till PID i Person. Detta gör att du skapar en Fk (Foreign Key)



21. Välj **OK** så stängs den ena dialogrutan och den andra visas. Ställ in funktionaliteten för Fk enligt följande på Delete Rule i avsnittet INSERT And UPDATE Specifikation:



22. Välj Cascade. Du har följande möjligheter:

**No action** innebär att du inte kan radera en post i tabellen Person utan att du först har raderat alla barnposter i tabellen Telefon. Du kan heller inte lägga till en Telefon utan att motsvarande post finns i Person.

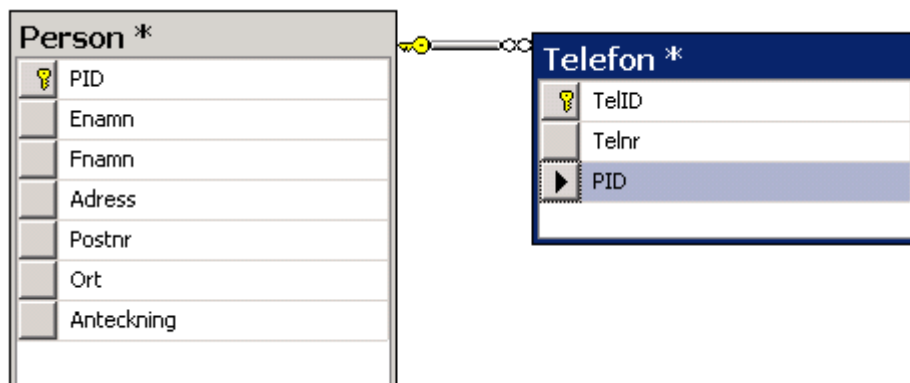
**Cascade** innebär att om du raderar en post i tabellen Person så raderas alla barnposter i tabellen Telefon.

**Set Null** innebär att om du raderar en post i tabellen Person så läggs Nullvärde in fältet Pid på alla barnposter i tabellen Telefon. Då måste du ange att du också tillåter Nullvärden i fältegenskapen.

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

23. Under Update Rule hittar du samma val som i föregående uppgift. Vid en ändring av primärnyckel på föräldersida kommer uppdatering automatiskt ske av Fk på barnsida om du anger Cascade. Anger du No Action tillåts inte att du ändrar Pk på föräldersida. Regel. Använd denna endast om du har talande nyckel på föräldersida. Undvik att använda talande nyckel och då sätter du denna till Restrict.

24. Spara genom att klicka på knappen OK.

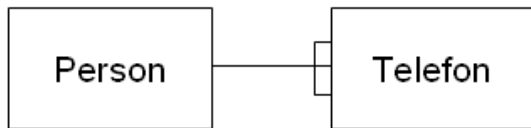


25. Spara diagrammet genom att stänga och döpa det till Adress. Du måste också spara tabellerna eftersom Fk påverkar tabellerna.



## Arbeta med poster i tabeller

Följande datamodell ligger till grund för de två tabeller du nu arbetar med.



Du ska nu få lägga till några poster i Person och några i Telefon

1. Öppna tabellen Person genom att högerklicka på den och välj **Edit 200 Rows**.
2. Markera fältet **Enamn**.  
Skriv: Johansson och tryck på tab så markeras fältet Fnamn.
3. Skriv: **Anna** och tryck på tab. Fortsätt att lägga in:  
**Ringvägen 23** och tryck på tab  
**393 51** och tryck på tab  
**KALMAR** och tryck på tab
4. Gå till nästa rad så bör det bli nu så att du får ett nr i PID på första raden. Det bör vara nr 1.
5. Lägg in ytterligare två poster:  
Karlsson Stina Åsgatan 23 393 64 KALMAR Historielärare  
Karlberg Stig Kyparevägen 12 393 63 KALMAR
6. Vill du ta bort en rad i tabellen så markerar du postväljaren längst till vänster på den aktuella raden. Tryck på Delete.
7. Du ska nu få lägga in ett par telefonnr på Stig Karlberg. Du bör därför komma ihåg vad han har för Pid.  
Öppna tabellen Telefon.
8. Markera fältet Telnr.  
Skriv: **0480-123456**  
Skriv: **3** som ska motsvara Pid för Stig Karlberg. Tryck på tab så markeras nästa fält.
9. Lägg in en rad till med Stigs mobiltelefon – **070 754 8965**.
10. Prova att lägga in en post där du anger ett Pid som inte finns i Person – vad händer?

11. Att skriva några **SQL-frågor**. Högerklicka på din databas. Välj New Query. (Om du väljer New Query på verktygsfältet måste du börja med Use och namnet på din databas som SQL-sats. Ex: **Use UD08\_xx99yy\_Address**)  
I fönstret som öppnas skriver du dina SQL frågor och väljer Execute (F5) på verktygsfältet. Studera resultatet. Execute (F5) exekverar (kör) din SQL kod. Avsluta varje SQL sats med detta.

```
SELECT Enamn, Fnamn  
FROM Person  
ORDER BY Enamn
```

Studera resultatet. Visar allas efternamn och förnamn sorterade på efternamn

```
SELECT Fnamn, Enamn, Ort  
FROM Person  
ORDER BY Fnamn
```

Studera resultatet. Visar alla förnamn, efternamn och ort sorterade på förnamn.

```
SELECT Fnamn, Enamn, Telnr  
FROM Person  
INNER JOIN Telefon  
ON Person.Pid=telefon.Pid
```

tryck F5 och studera resultatet. Visar förnamn, efternamn och telefonnr för de personer som har telefon registrerad i tabellen telefon

```
SELECT Fnamn, Enamn, Telnr  
FROM Person  
LEFT OUTER JOIN Telefon  
ON Person.Pid=telefon.Pid
```

Tryck F5 studera resultatet. Visar alla förnamn och efternamn för alla personer och telefonnr för de som har telefonnr registrerad i telefon.

```
SELECT Fnamn, Enamn, Telnr  
FROM Person  
RIGHT OUTER JOIN Telefon  
ON Person.Pid=telefon.Pid
```

Tryck F5 och studera resultatet. Visar alla poster ur telefon med förnamn och efternamn, telefonnr för de personer som har telefonnr registrerad i telefon.

```
SELECT Enamn, Fnamn  
FROM Person  
WHERE Enamn='Karlberg'
```

Studera resultatet. Visar efternamn och förnamn för de som har Karlberg i fältet Enamn.

12. Du är nu klar och kan avsluta MS SQL Server Management Studio och även dina andra program som du har igång för denna övning.