



Relationer og Subforms

Artiklen beskriver/gennemgår de generelle metoder til at oprette relationer mellem tabeller, og senere vise/redigere data i main og subforms. Artiklen er gratis, og henvender sig primært til den mindre øvet bruger.

Skrevet den **05. Feb 2009** af **-anders-** I kategorien **Databaser / Access** | ★★★★★

Indledning:

Jeg vil i denne artikel forsøge at gennemgå de generelle teknikker omkring styring af subforms til at vise relateret data fra mere end en tabel, samt opsætning af korrekte relationer mellem tabeller. I den tid jeg pt. har været aktiv på eksperten har jeg jævnligt oplevet indlæg omkring disse emner, og syntes derfor det var relevant at forsøge at beskrive disse emner overordnet. Artiklen vil gennemgå lidt teori omkring emnet, samt noget praktisk. Artiklen er opdelt i 2 hovedemner, Relationer, tabeldesign e.t.c. Og formulardelen, visning i subforms, mainforms, Underordnet og Overordnet felter, samt oprettelse af nye poster.

For at gøre artiklen mere forståelig tager jeg udgangspunkt i en simpel adressedatabase hvor man kan opretter personer med stamoplysninger, samt efterfølgende tilknyttet disse personer til en eller flere grupper. Hvis du efter at have læst artiklen har lyst til at se en demo er du velkommen til at skrive til anj@bryggemail.dk, så vil jeg sende demoen til dig hurtigst muligt. Undervejs i artiklen henviser jeg til diverse objekter, disse objekter er de samme som i demobasen. Artiklen henvender sig nok til den mindre øvet bruger, men jeg håber at alle kan få glæde af den. Der bliver brugt meget spalteplads på den relationelle del af gennemgangen da denne del er det vigtigste ift. Til teknikken omkring visning af relateret data i subforms.

Generelt om den relationelle styring

Hvis man ønsker at vise relateret data i en subform fungerer det typisk på denne måde:

Man har en mainform, denne mainform har en tabel som postkilde, i tabellen er der så et unikt felt f.eks. et idfelt af typen autonummering, værdien i dette felt bruges til at identificere den aktuelle post unikt. For at kunne styre visningen af relateret data i en subform, og have velfungerende relationer er det nødvendigt at poster i databasens tabeller kan identificeres unik. Vi prøver et praktisk eks.

Opret en tabel, navngiv tabellen tblpersoner, og tilføj nedenstående felter:

PersonID (Datatype angives til autonummering, og feltet sættes til primær nøgle)

FuldeNavn (Datatype angives til Tekst).

Du har nu en tabel hvor hver person kan identificeres unikt via PersonID feltet, dette er fint nok, men du kan sådan set ikke bruge det til noget ifht. At vise relateret data i en subform, derfor er der brug for en ny tabel, så vi kan angive at en person kan være medlem af en eller flere grupper.

Opret en ny tabel, navngiv tabellen tblgrupper, og tilføj nedenstående felter:

GruppeID (Datatype angives til autonummering, og feltet sættes til primær nøgle)

GruppeNavn (Datatypen angives til Tekst).

Du har nu en ny tabel hvor hvert gruppenavn igen kan identificeres unikt via GruppeID feltet.

Ideen er så at få nogle relationer op og køre imellem tabellerne, men før vi begynder på det er det nødvendigt at oprette en tabel mere, eks. går jo ud på at reg. et eller flere gruppemedlemsskaber for en eller flere personer (mange til mange tabellen).

Opret derfor en ny tabel, navngiv tabellen tbldata, og tilføj nedenstående felter:

Person (Datatypen angives til Tal)

Gruppe (Datatypen angives til Tal)

ID (Datatype angives til autonummering, og feltet sættes til primær nøgle).

Det var så tabellerne, før vi kan komme videre er det nødvendigt at lave nogle relationer imellem de 3 tabeller, så datavisning i vores subforms kommer til at virke hurtigt og gnidningsløst, men først en opsummering på tabeldesignet:

Tblpersoner bruges til at registrere personer med stamoplysninger feks. Navn og adresse e.t.c.

Tblgrupper bruges til at registrerer grupper en eller flere personer kan være medlem af.

Tbldata bruges til at "samle" data for de 2 ovenstående tabeller, dvs. Når du melder en person ind i en gruppe eller angiver at en person skal være medlem af en gruppe, samles disse registreringer i tbldata.

I tbldata har du 2 felter Person, Gruppe, det er disse 2 felter der er nøglen når vi senere skal vise relateret data i vores subforms, sagt på en anden måde: de subforms vi senere skal oprette vil trække data fra tbldata vha. At benytte en forsprøgsel til formålet, og enten forsørge på PersonID eller GruppelID, alt efter hvilken visning vi vælger.

Oprettelse af relationer mellem tabellerne:

Nu det tid til at oprette relationer, du behøver ikke at oprette dem manuelt vi lader Access oprette dem næsten automatisk:

Åben tbldata i designvisning, i feltet Person skal du i datatypecomboen vælge guiden opslag, når du har valgt guiden åbens en dialog her skal du angive at feltet person skal slå op i tblpersoner. Medtag felterne PersonID (vigtigt), og feltet FuldeNavn, guiden vil som standard foreslå at PersonID feltet skjules da det er nøglefeltet, dette skal du blot angive.

Gentag samme procedure for feltet Gruppe, her skal du blot vælge tblgrupper som opslagstabel, luk tabellen tbldata og svar ja til at gemme. Nu er de nødvendige relationer oprettet. Du kan nu åbne relationsvinduet, her kan du se de relationer du lige har oprettet via guiden opslag., her kan du højre klikke på en relation for at angive diverse relationsegenskaber. Hvilke egenskaber du angiver har ikke nogen betydning for hvordan dine data vises i subforms (dog ikke hvis du angiver andre joinegenskaber), egenskaber for relationer i øvrigt ligger uden for denne artikel.

Nu er dine tabeller klar til at indtaste poster, det er en god ide at indtaste nogle poster i tblpersoner og tblgrupper før du går videre. Prøv at indtaste 2 eller flere personer i tblpersoner, og indtast en eller flere gruppenavne i tblgrupper. Når du har indtastet en eller flere personer i tblpersoner kan du nu se et lille + tegn helt til venstre i tabellen, hvis du klikker på + tegnet kan du vælge en eller flere grupper den valgte person skal være medlem af disse registreringer gemmes automatisk i tbldata.

Du kan nu gøre nøjagtigt det samme i tblgrupper her vendes det bare om, klik på + tegnet ud for en gruppe, nu kan du vælge en eller flere personer der skal være medlem af den valgte gruppe. Du kan altså nu takket være korrekte relationer registrere "begge veje", det ligemeget om du registrere via tblpersoner eller tblgrupper, dataregistreringerne bliver i begge tilfælde gemt korrekt i tbldata da både Gruppenavn og Fuldenavn kan indetificeres unikt via PersonID og GruppelID.

Hvis du åbner tbldata kan du se de registreringer du lige foretaget, du kan for den sags skyld også registrere poster i tbldata, her skal du så blot vælge både person og gruppe. Tbldata er mange til mange tabellen dvs. i denne tabel vil en person og eller en gruppe ligge flere gange alt efter hvor mange grupper en person er medlem af, men fordelten er at personen kun er repræsenteret en gang i tblpersoner, det samme gør sig gældende for tblgrupper, så både tblpersoner og tblgrupper er en siden af relationen. Denne struktur forhindre så redundante data i opslagstabellerne tblpersoner og tblgrupper. Du har nu en velfungerende database med et korrekt tabeldesign, alle disse herligheder styres nu nemt og smertefrit, udelukkende fordi at posterne kan indetificeres unikt via PersonID og GruppelID. I tbldata har du også et

nøglefelt ID, det er nødvendigt med dette felt da poster i tblldata også skal kunne identificeres unikt. Sagt på en anden måde: Det er ikke muligt at lave relationer mellem tabeller hvis ikke tabellen har et nøglefelt der er angivet som Primær nøgle.

Man kunne spørge hvorfor man ikke blot angiver at FuldeNavn i tblpersoner skal være nøglefelt, og hvorfor ikke angive at GruppeNavn skal være nøglefelt i tblgrupper. Det kunne man også godt, men kan et Navn på en person være unikt, nej før eller siden ville man støde ind i problemer ikke bare i et adressekartotek, men også i mange andre databasemæssige sammenhænge. Men hvad med gruppenavnet, kunne det ikke være unikt, jo sagtens, men forestil dig indholdet af et nøglefelt som feks. "Familie (men ikke fætre og kusiner)", det kan blive en tung sag at forespørge på et sådant nøglefelt, men det kan selvfølgelig lade sig gøre.

Med dette tabeldesign er der nu ikke nogen begrænsninger for hvad du kan gøre med dine data ifht. Til visning af data i diverse main og subforms. Brugeren mugs her på E havde engang et glimrende indlæg i et spørgsmål vedrørende relationelle visning af data, jeg citere (efter hukommelsen):

Betragt dine data som et fjernvarmesystem, men i stedet for at det er varmt der flyder frem og tilbage mellem husstande, er det her blot data der flyder frem og tilbage:o)

Visning af relateret data i main og subforms:

Nu er selve grundstrukturen i databasen færdig, alle tabeller er oprettet, med de relevante felter, og relationer er sat korrekt op, så nu det tid til at designe main og subforms så data kan præsenteres på en nem og overskuelig måde. Allerst skal der oprettes en forespørgsel, denne forespørgsel skal bruges til at vise relateret data i 2 subforms, Personer efter gruppe, og Gruppe efter personer.

Opret en ny forespørgsel vha. at benytte guiden simpel forespørgsel, datakilden til forespørgslen skal være tabellen tblldata, medtag felterne Person, og Gruppe, gem forespørgsel og navngiv den fssubforms.

Opret nu en ny formular, formen skal oprettes som enkeltkolonne, angiv postkilden til tblpersoner. Sørg for at der er plads til en combobox øverst i formen, og en kommandoknap nederst, i midten af formen skal der være plads til at indsætte en subform. Indsæt derefter alle felter fra tblpersoner, du kan skjule PersonID feltet hvis du ønsker det, men feltet skal være på formen. Gem formen og navngiv den frmperson. Det var så mainformen til tblpersoner, nu skal der oprettes en subform der automatisk viser de grupper den aktuelle person er medlem af når du skifter post i mainformen.

Opret en ny form i dataarkvisning vha. at benytte guiden, angiv formens postkilde til forespørgslen fssubforms, medtag feltet Gruppe (opslagsfelt), tilpas formen så den passer til frmperson. Gem formen og navngiv den frmpersonersub. Nu er de nødvendige forms oprettet til visning af gruppedlemskaber for en person. Det er nu tid til at indsætte subformen frmpersonersub på mainformen frmperson, det er her vi skal angive Underordnet og Overordnet felter så data kan vises korrekt for den valgte person.

Indsæt subform og angiv Underordnet og Overordnet felter:

Det er her tryllet begynder, ved at angive nogle ganske simple kriterier kan vi få vist de korrekte data i subformen der matcher den aktuelle person i mainformen.

Åben frmperson i designvisning, klik på på værktøjskasseikonet i værktøjslinjen, sørg for at tryllestaven øverst til højre er aktiveret. Klik nu på ikonet i værktøjskassen "Underformular/underrapport" ikonet er det næst nederst til højre i værktøjskassen. Vælg Anvend en eksisterende formular, og marker formen frmpersonersub på listen, klik på Næste, og klik på Udfør. Nu er subformen indsat på mainformen, og de Underordnet og Overordnet felter er angivet helt automatisk. Gem og luk frmperson, prøv at åbne formen og skifte imellem posterne, du kan nu se at subformen "følger med", og viser de grupper den aktuelle person er medlem af. Hvis du vil tilføje en gruppe for den aktuelle person skal du blot vælge gruppen på comboen i subformen, så bliver det automatisk registreret i tblldata. Hvis du vil slette et gruppedlemskab for den aktuelle person skal du blot højreklikke på gruppenavnet i subformen, og vælge slet post. Denne handling sletter kun registreringen i tblldata, gruppen bliver ikke slettet i tblgrupper.

Du kan blot nu oprette en ny mainform i enkeltkolonne, angiv formens postkilde til tblgrupper, tilføj alle felter fra tblgrupper, gem formen og navngiv den frmgruppe, formater formen som frmperson. Opret en ny subform i datakarvisning, angiv formens postkilde til fssubforms, og tilføj feltet person (opslagsfelt), gem formen og navngiv den frmgruppersub. Indsæt subformen på frmgruppe som beskrevet ovenfor, men angiv kilden til frmgruppersub, gem og luk frmgruppe. Det er nu også muligt at registrere "den anden vej", væg blot en person på comboen i frmgruppersub, og registreringen bliver gemt automatisk i tbldata.

Du kan nu få vist relateret data to veje Personer efter gruppe, og Grupper efter personer alt efter hvilken mainform du åbner. Alle disse herligheder er designet og virker uden at du har skrevet en eneste linje kode.

Forklaring af Underordnet og Overordnet felter:

Visning af data i subforms kan laves uden at angive ovenstående vha. at benytte diverse Select statement (SQL), men ovenstående metode er Access "indbygget" måde at gøre det på, holder man styr på sine nøgler i tabellerne er metoden 100 % skudsikker. Generelt kan man sige at:

Det underordnet felt er feltet i tbldata der svare til nøglefeltet i en siden af relationen, hvis man tager tblpersoner er det overordnet felt PersonID i tblpersoner, og det underordnet felt er Personfeltet i tbldata., det samme gør sig gældende i tblgrupper. Begge felter i tbldata er opslagsfelter, hvert opslag består af 2 kolonner den skjulte kolonne (0) som er Nøglen og den synlige kolonne (1) som er det man ser, men det hele køre udelukkende på de skjulte nøglefelter. Hvis du fjerner et af nøglefelterne i en af tabellerne braser det samme, og vil ikke længere virke.

Prøv at tjekke dine underordnet og overordnet felter i frmperson ved at følge nedenstående:

Åben frmperson i designvisning, højreklik på subformen og vælg egenskaber, klik på fanebladet Data. Her kan du nu se de underordnet og overordnet felter Access automatisk har angivet, de svare nøjagtigt til ovenstående gennemgang. Du kan også selv angive denne egenskab, og eller ændre det. Prøv at klikke ude til højre i det underordnet felt, nu åbnes en dialog her kan du se selve sammensætningen af tabellerne der for dine subforms til at virke korrekt. Man kan sågar også få Access til at foreslå det automatisk vha. at klikke på knappen Foreslå. Hvis du arbejder med en database og viser data i main og subforms, og det fejler, er det typisk de overordnet og underordnet felter man skal kikke på, det ofte her det går galt.

Da vi indsatte subforms på mainforms var tryllestaven aktiveret på værktøjskassen, hvis du slår den fra er det netop her du selv kan gå ind og definere dine under og overordnet felter.

Navigation i mainforms:

Har du angivet at navigationsknapper i mainforms skal vises kan du sagtens benytte disse til at bladere mellem posterne på mainformen, men der findes en mere "elegant" måde at gøre dette på. Der er ofte indlæg på E med ordlyden "hvordan går jeg til en bestemt post ved at vælge den i en combobox". Prøv at følge nedenstående gennemgang, det faktisk ikke så galt:

Åben frmperson i designvisning, klik på værktøjskassen, og sørg for at tryllestaven er aktiveret. Klik på combobox ikonet på værktøjskassen og indsæt en combobox øverst i din mainform. Når comboen er sat ind starter Access en guide der spørger hvordan comboen skal virke her skal du vælge mulighed 3 "Søg efter en post i formularen på basis af den værdi jeg har valgt i kombinationsboks" Klik på Næste, nu åbnes en ny dialog hvor du skal angive rækkeilden til comboen, vælg felterne PersonID og FuldeNavn, og klik på næste. Nu vises det faktiske indhold i comboen du lige har oprettet, bemærk at nøglefeltet igen er skjult, men det er faktisk dette felt vi bruger til at navigere i mainformen. Klik på Næste og Klik på udfør. Nu er comboboxen færdig, luk mainformen og svar ja til at gemme.

Prøv nu at åbne frmperson og vælg en person i comboen øverst, nu gæes der hurtigt nemt og smertefrit til den person du har valgt, og subformen følger med og viser de grupper den valgte person er medlem af.

Du kan gøre nøjagtigt det samme i frmgruppe, når comboen er indsat og designet kan du vælge en gruppe

på din combo øverst, der gøres som ovenstående til den aktuelle post og subformen viser alle personer der er medlem af den valgte gruppe.

Hvordan virker disse comboboxe:

Som nævnt tidligere har du endnu ikke selv skrevet en eneste linje kode, men Access har faktisk skrevet noget kode for dig automatisk da vi indsatte comboboxene på de to mainforms. Prøv at åbne frmperson i designvisning, højreklik på din combo øverst og vælg egenskaber. Prøv at klikke på fanebladet Data, her kan du se den select som Access automatisk har genereret da du indsatte comboen. Prøv at klikke på fanebladet Hændelser, du vil se at der ud for comboens egenskab Efter Opdatering står [Hændelsesprocedure], det er her Access har skrevet koden der for din combo til at fungere som navigator imellem posterne på din mainform. Klik på de 3 små ... prikkes til højere for proceduren, nu åbnes VBA vinduet, og du kan se hvordan koden er skrevet, den ser sådan ud:

```
Private Sub Kombinationsboks27_AfterUpdate()  
    ' Søg efter den post, der svarer til kontrolementet.  
    Dim rs As Object  
  
    Set rs = Me.Recordset.Clone  
    rs.FindFirst "[PersonID] = " & Str(Nz(Me![Kombinationsboks27], 0))  
    If Not rs.EOF Then Me.Bookmark = rs.Bookmark  
End Sub
```

Da vi senere skal vise en nyoprettet post vha. at benytte et serverfilter er det nødvendigt at angive FilterOn False i ovenstående kode, kodelinjen skal stå aller øverst i koden, se nedenstående:

```
Private Sub Kombinationsboks27_AfterUpdate()  
Me.FilterOn = False  
    ' Søg efter den post, der svarer til kontrolementet.  
    Dim rs As Object  
  
    Set rs = Me.Recordset.Clone  
    rs.FindFirst "[PersonID] = " & Str(Nz(Me![Kombinationsboks27], 0))  
    If Not rs.EOF Then Me.Bookmark = rs.Bookmark  
End Sub
```

Som du kan se er det nøglefeltet PersonID der spørges på og ikke personens navn, det er derfor at PersonID feltet skal være på formen evt. skjult, men det skal være der for at koden kan spørge på værdien i feltet. Hvis du ikke vil anvende en brugerdefineret navigering som beskrevet ovenfor behøver du slet ikke PersonID feltet på mainformen, frmpersonersub virker selv om du fjerner feltet fra mainformen, subformen "snakker" kun med de underliggende postkilder altså tabeller og eller forsørgsler, her skal nøglefelterne naturligvis være.

Om man ønsker at benytte denne form for navigation er naturligvis et spørgsmål om smag og behag, men jeg syntes at det fungerer godt, og det ikke specielt besværligt at få det til at virke, når bare tryllestaven er aktiv skriver Access som sagt koden automatisk.

Har man mange poster kan det være lidt uoverskueligt at vælge fra en combo, man kan hvis man ønsker det kan man benytte en liste i stedet for som navigator i mainformen. Det virker på nøjagtig samme måde som comboen, man skal blot vælge en liste fra værktøjskassen i stedet for en combo. Det er faktisk meget anvendeligt i mange sammenhænge.

Det var det, nu køre tabellerne, mainforms og subforms virker korrekt, med eller uden brugerdefineret navigation. Men hvad med oprettelse af nye personer og grupper. Hvis jeg opretter en ny person i frmperson direkte hvornår er posten så gemt, og hvornår ved Access at den nye post er gemt. Og hvordan reff. Jeg til en subform på en mainform Dette vil jeg forsøge at belyse i det sidste afsnit.

Opret og gemme nye poster i tblpersoner og tblgrupper:

Du kan altid oprette nye personer og grupper i de dertil indrettede tabeller direkte i tabelvinduet. Du kan også efterfølgende melde såvel personer og grupper ind på tværs af hindanden i tbldata direkte i tabelvinduet, men dette er ikke den optimale måde at gøre det på. Det er altid god skik at arbejde med disse ting i diverse forms ikke mindst hvis din database senere skal distribueres ud til en eller flere slutbrugere.

Som beskrevet tidligere kan du feks. Angive at en person skal være medlem af en eller flere grupper ved at vælge dette på subform i opslagfeltet for den aktuelle person i mainformen. Men dette kan kun lade sig gøre hvis posten på forhånd er gemt og Access ved at posten er gemt. Opretter du en ny person eller gruppe direkte i din mainform er posten ikke gemt før du feks. Går til en anden post. Opretter du en ny post direkte i mainformen og efterfølgende direkte går til subform og vælger en gruppe vil det give en nøglefejl fordi Access endnu ikke ved at posten er oprettet. Der findes mange måder at omgås dette kendte problem, nedenstående vil jeg vise en måde det kan gøres på.

Opret ny person direkte i mainform vha. at benytte en popup form:

Nedenstående eks. viser hvordan du kan oprette en ny post til tabellen tblpersoner direkte i frmperson vha. at benytte en popupform, og få posten gemt så det muligt at angive et eller flere gruppemedlemskaber med det samme i frmpersonersub, uden at lukke og genåbne frmperson manuelt.

Opret en ny form i enkelkolonne, angiv formens postkilde til tblpersoner og indsæt alle felter fra tblpersoner. Klik på fanebladet andre og angiv at formen skal være popup og modal. Popup egenskaben gør at formen altid ligger øverst på evt. andre åbne forms. Modalegenskaben gør at man ikke kan begynde på alle mulige andre ting før man er færdig med at oprette den nye person. Under egenskaber for formen, fanebladet Data angiver du at formen kun kan benyttes til at oprette nye poster med. Opret derefter 2 kommandoknapper og indsæt dem nederst på formen navngiv knapperne cmdgem og cmdannuller, titeltekst &Gem, &Annuller. & tegnet gør at man kan benytte ALT som genvejstast. Luk formen og navngiv den frmnyperson. Nu skal der så for første gang skrives noget brugerdefineret kode bag på de 2 knapper samt på formens Efter Opdaterings event.

Først knappen cmdannuller:

Åben frmnyperson i designvisning, højreklik på cmdannuller og vælg egenskaber. Klik på fanebladet Hændelser, ved knappens On Click event skal du åbne VBA vinduet imellem Privat Sub og End sub skal du skrive denne enkle kode

```
'Annuller oprettelsen'  
Me.Undo  
Lukker den aktive form'  
DoCmd.Close acForm, Me.Name
```

Så knappen cmdgem:

Igen ved knappens On Click event skal du skrive nedenstående enkel kode:

```
'Lukker formen'  
DoCmd.Close acForm, Me.Name
```

Til sidst formens After Update event skal du skrive nedenstående kode:

```
'Denne kode går til den nyoprettet post i frmperson, og viser'  
'de oplysninger man har angivet for den nye person'  
'Først genåbner vi frmperson og går til den post vi lige har oprettet'  
DoCmd.OpenForm "frmperson", , , "PersonID = " & Me!PersonID
```

'Nu er posten oprettet og gemt, og formen frmperson viser den post vi lige har oprettet'

```
'så nu skal comboen på frmperson opdateres, og samtidig vise den nye post'  
'Altså den nyoprettet persons navn'  
'Her opdateres comboen'  
Form_frmperson!Kombinationsboks27.Requery
```

```
'Her vises så den nyoprettet person navn i comboen'  
'Det er det unikke personid (første kolonne i comboen (0)der er nøglen'  
Form_frmperson!Kombinationsboks27 = Form_frmperson!PersonID
```

```
'Her sættes fokus på subformen frmpersonersub hvor det er muligt at'  
'angive et eller flere gruppemedlemskaber for den nye person. Bemærk'  
'at man her reff. til kontrolelementet frmpersonersub, og ikke til selve'  
'formnavnet'  
Forms!frmperson!frmpersonersub.SetFocus  
'Code End'
```

Det ser måske lidt voldsomt ud, men det meste er faktisk kun kommentare til koden. Luk frmnyperson og svar ja til at gemme.

Så hele koden til at gemme posten og efterfølgende finde den i frmperson ligger altså på frmnyperson på hændelsen Efter Opdatering. Hvorfor ikke angive koden på cmdgem. Når koden ligger på formens after update event vil den altid blive kørt uanset hvordan brugeren lukker formen. Man kunne sagtens undlade knappen cmdgem og nøjes med cmdannuller, knappen er blot medtaget for syns skyld :o).

Nu det til til at prøve det af i praksis:

Åben formen frmperson i design, indsæt en kommandoknap nederst i formen, navngiv knappen cmdnyperson, titeltekst &Ny person. Bag knappens On Click event skal du skrive nedenstående enkle kode:

```
'Åbner formen frmnyperson, formen åbnes som standard'  
'Kun nye poster tillades'  
DoCmd.OpenForm "frmnyperson"
```

Luk formen og svar ja til at gemme. Åben frmperson, og klik på cmdnyperson, nu åbnes frmnyperson, indtast oplysninger for den nye person, luk formen på den måde der passer dig bedst. Du vil nu stå på den nye post i frmperson du lige har oprettet, posten er gemt og der er sat fokus på subformen så nu det bare at melde personen ind i en eller flere grupper.

Du kan bruge ovenstående metode på nøjagtig samme måde til at oprette grupper direkte i frmgruppe, du skal blot oprette en ny form frmnygruppe, og tilpasse egenskaberne for formen, titeltekst på knapper e.t.c. I demobasen er denne form oprettet.

Metoden som beskrevet ovenfor kan virke noget omstændig, indrømmet, men den er skudsikker, der vil ikke opstå nøglefejl når du opretter nye poster til tblpersoner og tblgrupper, og efterfølgende registrere direkte i subforms. Der findes helt sikkert andre måder at gøre dette på, betragt det som input til problematikken og ikke som en universal løsning/anbefaling.

Hvordan referer man til en subform der er indsat på en mainform:

Når du har indsat en subform på en mainform er den ikke længere i Access's forstand en formular, men i reference forstand et kontrolelemte på mainformen på lige fod med feks. En tekstbox eller en etiket e.t.c. Dette betyder at du skal referere til subformen på nøjagtigt samme måde som feks. En tekstboks. I gennemgangen af oprettelse af ny poster benytter jeg faktisk en sådan referance:

```
Forms!frmperson!frmpersonersub.SetFocus. I dette eks. reff. Jeg til kontrolelementet frmpersonersub på mainformen frmperson, hvis jeg ville reff. Til selve formnavnet skulle det se sådan ud: Forms!frmperson.SetFocus. Hvis du vil reff. Til subformen i mainformen direkte altså ikke via en anden form kan du blot
```

benytte metoden: Me!frmpersonersub. SetFocus

Afslutning:

Det var så hvad jeg havde valgt at bringe på bane omkring relationer og visning af relateret data i main og subforms. Artiklen er lang, indrømmet, men det er komplekse ting at beskrive, det svært at gøre det meget kortere, til gengæld er artiklen gratis:o).

Er du en erfaren Access bruger/udvikler er der næppe noget nyt, men for den mindre øvet som måske endnu ikke har set lyset omkring relationer skulle der dog være lidt inspiration at hente.

Artiklen er ikke tænkt som en entydig anbefaling til hvordan man laver en adressedatabase, dette kan jo laves på mange andre måder, men udelukkende tænkt som et praktisk eks. jeg kunne for den sags skyld have taget udgangspunkt i næsten hvad som helst. Har du spørgsmål til artiklen er du velkommen til at skrive til anj@bryggemail.dk så vil jeg forsøge at besvare efter bedste evne.

God fornøjelse.
Hilsen Anders.

Kommentar af nmk d. 24. Mar 2006 | 1

Kommentar af paddington d. 29. May 2006 | 2

Wupti, og nu mangler vi bare at du også lave alt arbejdet for mig.... DAMN en god artikel!!

Kommentar af pavon d. 11. Apr 2006 | 3

Mangler at kunne vælge Genial. Lige noget jeg kunne bruge som ny Access "Haj"