

---

---

**Kapitel** | **Modul**  
**7**

<b>1 Moduler</b> .....	<b>2</b>
<b>2 Modulfönstret</b> .....	<b>2</b>
<b>3 Globala Moduler</b> .....	<b>3</b>
3.1 Skapa Global Modul .....	3
3.2 Spara Global Modul.....	3
<b>4 Variabler, Konstanter och Datatyper</b> .....	<b>4</b>
4.1 Variabler .....	4
4.2 Datatyper .....	5
4.3 Deklaration av Variabler .....	6
4.4 Konstanter .....	7
<b>5 Globala Procedurer</b> .....	<b>8</b>
5.1 Skapa en Global Procedur .....	9
<b>6 Kompilera Procedurer</b> .....	<b>13</b>
<b>7 Använda Procedur</b> .....	<b>13</b>
<b>8 Formulär-/Rapportmoduler</b> .....	<b>15</b>
8.1 Skapa en Formulär-/Rapportmodul .....	15



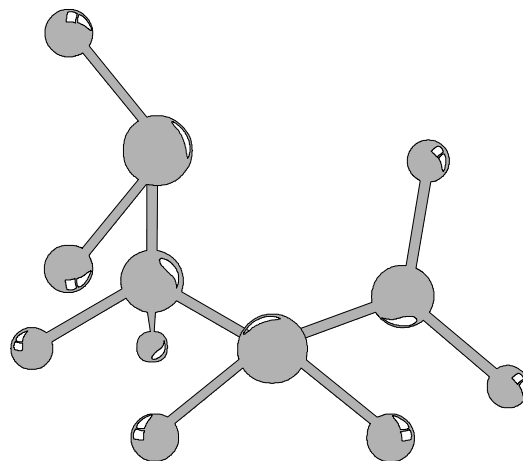
# Modul

Alla procedurer lagras i så kallade *Moduler* som även dessa namnges. Varje modul består av en deklarationsdel och därefter av ett antal procedurer. Deklarationsdelen innehåller satsen *Option Compare Database*. Satsen info-gas automatiskt och medför att textjämförelser utförs enligt den sorteringsordning som aktuell databas använder.

Moduler är ett bra sätt att administrera sina procedurer. Du kan t. ex. samla alla procedurer som hör till ett visst formulär i en modul.

Detta kapitel kommer att gå igenom följande ämnesområden:

- Vad är en modul?
- Skapa globala moduler.
- Vad är variabler, konstanter och datatyper?
- Skapa globala procedurer.
- Kompilering av procedurer.
- Skapa formulär-/rapportmoduler.



# 1 Moduler

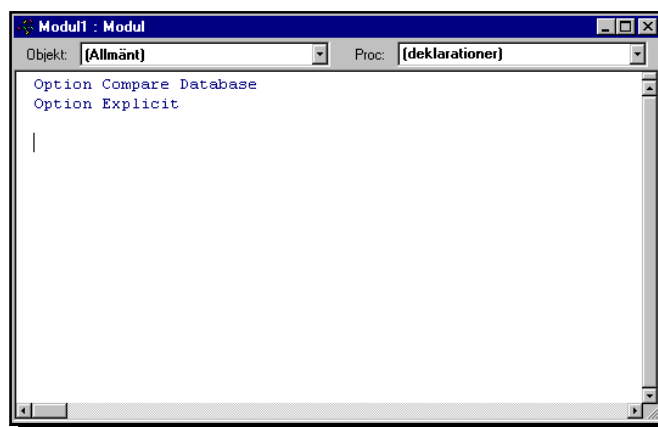
Det finns två olika sorters moduler du kan skapa. Vi ska nu redogöra för skillnaderna mellan dessa.

- **Globala moduler** skapar du som egna databasobjekt i databasen. Procedurerna i dessa moduler kan du anropa från uttryck, makron, händelser eller från procedurer i andra moduler. Globala moduler är tillgängliga från alla databasobjekt i databasen.
- **Formulär-/rapportmoduler** Varje rapport eller formulär du skapar i databasen innehåller automatiskt en moduldelen, som i sin tur innehåller de procedurer du kopplar till de olika händelseegenskaperna i formuläret/rapporten. Dessa moduler visas inte som databasobjekt i databasfönstret och dess procedurer och moduler namnges inte. Det enda sättet att studera eller redigera dessa procedurer är genom att öppna formuläret/rapporten i designläget och därefter, via egenskapsrutan, ta fram Visual Basic-koden.

## 2 Modulfönstret

Den globala modulen skapar du som ett eget databasobjekt. Du öppnar då ett modulfönster på samma sätt som när du öppnar ett makrofönster. Modulfönstret ser ut på följande sätt:

FIGUR 7.1



## 3 Globala Moduler

När du skapar en ny procedur kan du göra det antingen i en modul som finns sedan tidigare eller så skapar du en ny modul samtidigt med den nya proceduren.

### 3.1 Skapa Global Modul

Procedurer sparas i moduler, så innan du kan skriva in din Visual Basic-kod måste du ha en modul att spara proceduren i. Du ska nu skapa en modul som vid ett senare tillfälle skall ta emot din Visual Basic-kod.

Gör så här:

- Se till att databasen **C:\Access97-FK\Animalens Djursjukhus 4** är öppen.
- Klicka på fliken **Moduler** i databasfönstret.
- Klicka därefter på knappen **Ny**.

Nu öppnas modulfönstret och du kommer automatiskt in i deklarationsdelen som är redo att ta emot Visual Basic-kod.

### 3.2 Spara Global Modul

Du ska nu spara din modul för att kunna använda den vid ett senare tillfälle i detta kapitel.

Gör så här:

- Välj **Spara** i **Arkiv**-menyn.

- I rutan som nu visas döper du modulen till **Global**.
- Klicka på knappen **OK** för att verkställa sparningen av modulen.

## 4 Variabler, Konstanter och Datatyper

Innan du börjar skriva din kod i modulfönstret ska du få en förklaring på några av de benämningar som ingår när du skapar en procedur. När du skapar en procedur innehåller denna *Variabler*, med andra ord värden som varierar beroende av olika händelser som sker i proceduren.

Variablerna kan vara av en viss *Datatyp* vilket innebär att de innehåller en viss typ av värde. En numerisk variabel kan t. ex. inte innehålla vanlig text utan endast siffror.

Proceduren innehåller troligen också *Konstanter*, det vill säga värden som är konstanta genom hela procedurkörningen oavsett vad som inträffar.

### 4.1 Variabler

När du kör en procedur behöver du kanske lagra ett värde för att senare i proceduren jämföra detta mot ett annat värde eller använda värdet i en beräkning. Värdet sparas då i en *variabel*. En variabel har på samma sätt som ett fält i en tabell ett namn och är av en viss datatyp.

När du namnger en variabel måste du följa vissa regler:

- Namnet måste börja med en bokstav.
- Namnet måste bestå av bokstäver, siffror eller understryknings-tecken. Punkter och mellanslag godtas inte.
- Namnet får vara högst 40 tecken långt.
- Namnet får inte vara ett ord som används internt av Access.

## 4.2 Datatyper

En variabel är också av en viss datatyp. Om du inte direkt tilldelar variabeln en datatyp tilldelas den automatiskt datatypen *Variant*. Variabler med denna datatyp kan lagra numeriska värden, datum/tid och teckensträngar. Om du vill att din variabel ska utföra en beräkning med denna datatyp, så konverterar Visual Basic värdet automatiskt till ett numeriskt värde.

Om du däremot vet att du vill lagra en viss typ av värde i en variabel är det bättre att ange den rätta datatypen för variabeln direkt i koden.

### Datatyper som finns att välja mellan

Datatyp	Lagrar värde
<i>Integer</i> (heltal)	-32 768 till 32 767
<i>Long</i> (långt heltal)	-2 147 483 648 till 2 147 483 647
<i>Single</i> (Flyttal med enkel precision)	ger 7 siffrors noggrannhet och kan lagra tal för negativa värden inom området -3,402823E38 till -1,401298E-45 samt för positiva värden och 0 inom området 1,401298E-45 till 3,402823E38.
<i>Double</i> (flyttal med dubbel precision)	ger 16 siffrors noggrannhet och kan lagra tal för negativa värden inom området -1,797823E38 till -1,401298E-45 samt för positiva värden och 0 inom området 1,401298E-45 till 3,402823E38.
<i>String</i> (sträng)	0 till 65 535 byte.
<i>Variant</i>	Valfritt numeriskt värde upp till gränsen för Double.