

SAS[®] Programming 1: Essentials

Quick Guide

– hjælp i dagligdagen til nye SAS
programmører

SAS® Programming 1: Essentials Quick Guide is created by SAS Knowledge & Education in Denmark.

Copyright © 2009 SAS Institute Inc. Cary, NC, USA. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

Prepared: June 2009

Indholdsfortegnelse

DATA og PROC step

<i>Regler og syntax i SAS programmer.....</i>	<i>4</i>
<i>Udskriv indholdet af tabeller.....</i>	<i>5</i>
<i>DATA step.....</i>	<i>6</i>
<i>Læs fra og skriv til Excel regneark.....</i>	<i>7</i>
<i>Indlæs tekstfil.....</i>	<i>8</i>
<i>Kombinér tabeller</i>	<i>9</i>
<i>Opret SAS-formater.....</i>	<i>10</i>
<i>Deskriptiv dataanalyse.....</i>	<i>11</i>
<i>Summerede rapporter.....</i>	<i>12</i>
<i>Summerede data.....</i>	<i>13</i>

Generel funktionalitet

<i>WHERE statement</i>	<i>14</i>
<i>SAS Data set options</i>	<i>16</i>
<i>SAS-funktioner</i>	<i>17</i>
<i>SAS-formater.....</i>	<i>19</i>
<i>Output Delivery System (ODS).....</i>	<i>21</i>

<i>Information om SAS software</i>	<i>22</i>
--	-----------

Regler og syntax i SAS programmer

Emne	Regel/SAS syntax	Eksempel
SAS step - Start	Begynder med DATA- eller PROC- statement.	<pre>proc print data=orion.sales; run;</pre>
SAS step - Slut	Afsluttes med et RUN; eller QUIT;	<pre>proc print data=orion.sales; run;</pre>
Statements	Begynder med et keyword og slutter med et semikolon .	<pre>var first_name last_name ;</pre>
Frit format	En/flere blanke eller specieltegn adskiller ord. Et statement kan fylde flere linjer. Flere statement kan stå på samme linje.	<pre>proc means data=orion.sales; class gender; var salary; run;</pre>
Kommentar	Starter med stjerne og slutter med semikolon.	<pre>* ---- en kommentar ---- ; * var salary gender ;</pre>
Kommentar	Starter med /* og slutter med */.	<pre>/* ---- en kommentar ---- var salary gender */</pre>
Numeriske kolonner	Anvendes til at gemme på tal, der skal regnes på. Gemmes som floating point på 8 bytes (standard). Kan indeholde 16 til 17 betydende cifre.	
Tekstkolonner	Kan indeholde alle tegn. Længden kan være fra 1 til 32.767 bytes. Du kan bestemme længden med et LENGTH statement i et DATA step.	<pre>length job_title \$ 20;</pre>
Navne: SAS data set og variable	Op til 32 tegn langt. Ingen specialtegn eller blanke i navnet. Skal starte med bogstav eller understregning. Efterfølgende karakterer kan være bogstaver, tal eller understregning. Må ikke indeholde Æ, Ø eller Å. Der tages ikke hensyn til store og små bogstaver ved efterfølgende brug.	<pre>data _tabel2; set orion.sales; _2008_loen = salary; koen = gender; run;</pre>
Specialtegn og blanke i navne	Navne som ikke overholder reglerne ovenfor kan benyttes ved at anvende apostroffer, efterfulgt af bogstavet N.	<pre>options validvarname=any; data _tabel2; set orion.sales; 'løn 2008'n = salary; run;</pre>

Udskriv indholdet af tabeller

1. Oversigt over tabelstruktur: Kolonnenavne, type, formater, informater, labels, antal rækker m.v.
2. Dan simpel listerapport.
3. Sortér tabel.
4. Dan grupperet listerapport med grand- og subtotaler m.v.

<pre> /* Udskriv indholdet af tabeller */ libname orion 'c:\kursus\prg1'; proc contents data=orion.employee_donations; run; proc contents data=orion._all_ nodes; run; proc print data=orion.employee_donations noobs; var Employee_ID Total; run; proc sort data=orion.employee_donations out=donations; by paid_by descending qtr4; run; title1 "Donationer fra de ansatte"; footnote1 "Dannet den, %sysfunc(today()),dandfwdx18.);"; proc print data=donations noobs label split='/' n='Antal = ' uniform; by paid_by; pageby paid_by; id paid_by employee_id; var qtr1 qtr2 qtr3 qtr4; sum qtr1 qtr2 qtr3 qtr4; sumby paid_by; label employee_id='Ansættelses-/nummer' paid_by ='Betalings-/type'; format qtr1-qtr4 commax10.; run; title; </pre>	<p>Oversigt over alle kolonner i tabel.</p> <p>Oversigt over alle tabeller i bibliotek.</p> <p>Dagsdato på dansk i fodnoten.</p> <p>Label option skal med for at få kolonnebeskrivelser.</p> <p>Nulstil alle overskrifter.</p>
--	--

DATA step

Et datastep vil typisk indeholde følgende fire blokke af kode i nedenstående anbefalede rækkefølge:

1. Læs og begræns input-data.
2. Manipulér data.
 - a. Rens dataindhold.
 - b. Dan nye kolonner.
3. Bestem rækker og kolonner i den nye tabel.
4. Beskriv kolonneoverskrift og – indhold.

<pre> data work.sales; length job_group \$10; set orion.sales; where salary > 0 and hire_date > '01jan1990'd; country = upcase(country); if index(lowcase(job_title),'manager') then do; job_group='Manager'; if country='US' then bonus=salary*0.10; else bonus=salary*0.12; end; else do; job_group='Employee'; bonus=salary*0.15; end; total=sum(salary,bonus); age_hired=intck('year',birth_date,hire_date); if total >= 20000; keep total country employee_id salary bonus job_group age_hired birth_date; format total salary bonus commax20. age_hired 3.0 birth_date ddmmmyyp10.; label total = 'Løn med Bonus' age_hired = 'Alder ved ansættelse'; run; </pre>	<p>1. Læs og begræns input-data:</p> <ul style="list-style-type: none"> • Dan tabellen work.sales. • Læs data fra orion.sales. • Begræns antal rækker. <p>2. Manipulér data:</p> <ul style="list-style-type: none"> • Rens dataindhold. • Dan nye kolonner. <p><i>Forskel mellem 2 datoer (antal år).</i></p> <p><i>Begræns rækker vha. ny kolonne.</i></p> <p>3. Bestem rækker og kolonner i den nye tabel.</p> <p>4. Beskriv kolonneoverskrift og indhold.</p>
--	---

Læs fra og skriv til Excel regneark

Læs Excel regneark	
<pre>libname orionxls excel 'C:\kursus\prg1\sales.xls'; proc contents data=orionxls._all_ nods; run; proc contents data=orionxls.'Australia\$'n; run; proc print data=orionxls.'Australia\$'n; run; data work.AU_Females; set orionxls.'Australia\$'n; where Gender='F'; run;</pre>	<p>Opret adgang til et regneark.</p> <p>Oversigt over faneblade i regnearket.</p> <p>Kolonneoversigt.</p> <p>Udskiv indholdet fra et faneblad.</p> <p>Dan en SAS-tabel fra et faneblad i et regneark.</p>
Skriv til Excel regneark	
<pre>libname newxls excel 'C:\kursus\orion_data.xls'; proc copy in=orion out=newxls; run; proc datasets lib=newxls kill; run; proc copy in=orion out=newxls; Select Budget Country; run; proc delete data=newxls.budget; run; data newxls.budget; Set orion.budget; run; libname newxls clear;</pre>	<p>Opret et nyt regneark.</p> <p>Kopier alle tabeller til et regneark.</p> <p>Slet alle faneblade i et regneark.</p> <p>Kopier udvalgte tabeller til et regneark.</p> <p>Slet et faneblad i et regneark.</p> <p>Dan et nyt faneblad i et regneark.</p> <p>Frigiv regnearket, så andre kan åbne det.</p>

Indlæs tekstfil

Dan SAS-tabel ved indlæsning af tekstfil:





```
data ny_tabel;
    infile 'fysisk placering og filnavn';
    length tekst $ 12;
    input tal tekst ;
run;
```

Indlæs kommasepareret fil	
<pre>data salary; infile 'c:\kursus\prg1\sales.csv' dlm=','; length gender \$ 1 salary 8 first_name \$ 12 last_name \$ 18 ; input employee_id first_name last_name gender salary ; format salary commax10.; run;</pre>	<p>Kommasepareret fil.</p> <p>Bestem kolonnernes rækkefølge, type og længde i SAS-tabellen.</p> <p>Indlæs kolonner fra venstre mod højre, som de findes i tekstfilen.</p>
Indlæs kommasepareret fil med værdier, som valideres og konverteres	
<pre>data salary; infile 'c:\kursus\prg1\sales.csv' dlm=','; input employee_id :8. first_name :\$12. last_name :\$18. gender :\$1. salary :8. job_title :\$25. country :\$2. birth_date :date9. hire_date :mmddyy10.; format salary commax10. birth_date hire_date ddmmyyp10.; label birth_date = 'Fødselsdato' gender = 'Køn'; run;</pre>	<p>Kolon betyder, at indlæsningen sker sådan:</p> <ol style="list-style-type: none"> 1. SAS læser værdier mellem skilletegn. 2. Informatet bruges til konvertering. 3. Informatet bestemmer kolonnelængden.

Kombinér tabeller

Tabeller kan kombineres med følgende tre metoder:

1. DATA step med SET statement.
2. PROC APPEND (kun 2 tabeller).
3. DATA step med MERGE statement.

<pre>proc sort data=orion.Employee_US out=Employee_US; by Employee_Id; run; proc sort data=orion.Employee_AU out=Employee_AU; by Employee_Id; run; data Employee_ALL; set Employee_US Employee_AU(rename=(Region=Country)); by Employee_Id; run;</pre>	<p>Sortér tabeller, hvis rækker skal læses i ordnet rækkefølge.</p>  <p>Kolonnen Region omdøbes til Country.</p> <p>Rækker læses i ordnet rækkefølge.</p>
<pre>proc append base=Employee_All data=orion.Employee_DK force; run;</pre>	 <p>Brug FORCE, hvis kolonne-definitionen på de 2 tabeller ikke er ens.</p>
<pre>proc sort data=Employee_ALL; by Employee_Id; run; proc sort data=orion.Employee_Phone out=Phone; by Employee_Id; run; data Employee_Phone Error_Phone; merge Employee_ALL(in = E) Phone(in = P); by Employee_Id; if E = 1 then output Employee_Phone; if P and not E then output Error_phone; run;</pre>	<p>Sortér altid tabeller først.</p>   <p>E og P er kolonner med værdier 0 eller 1, og medtages ikke i de nye tabeller.</p>

Opret SAS-formater

1. Opret format med PROC FORMAT.
2. Angiv, hvor brugerdefinerede formater findes.
3. Anvend FORMAT statement i DATA eller PROC step. Se mere under emnet **SAS formater**.

```
proc format lib = orion.formats;
  value $country  'AU'      = 'Australia'
                  'US' , 'us' = 'United States'
                  other      = 'Miscoded';
  value salary_group
    low    - < 0      = 'Incorrect Salary'
    0      - < 50000  = 'Low Salary'
    50000  - 100000  = 'Medium Salary'
    100000 < - high  = 'High Salary';
run;

option fmtsearch = (orion work);

proc freq data=orion.sales;
  tables Country Salary;
  format Country $country.
           Salary salary_group.;
run;

proc format fmtlib lib=orion;
run;
```

Specielle værdier

- Other: Alle andre værdier.
- Low: Minus uendelig.
- High: Plus uendelig.
- Missing: Manglende værdier.

Søg efter formater i orion- og work-biblioteker i angivne rækkefølge.

Dan rapport med indhold af formater.

Deskriptiv dataanalyse

Procedurer til at undersøge om dine data er valide.

<pre>proc freq data = orion.sales order=freq; tables Country / missing nocum; tables Country*gender / nocum norow nocol nopercent; run; proc univariate data = orion.sales; class Country; var Salary; run; proc means data = orion.sales sum mean max maxdec=2; class Country; var salary; run;</pre>	<p>Frekvens uden akkumerede data 2-vejs frekvenstabel.</p> <p>Vælg kolonner, der skal undersøges med VAR statement.</p> <p>Vælg gruppering med CLASS statement.</p>
--	---

Deskriptive statistiske beregninger

SAS 9.1.3 <http://support.sas.com/onlinedoc/913/getDoc/da/proc.hlp/a000146729.htm>

SAS 9.2 <http://support.sas.com/documentation/cdl/en/proc/61895/HTML/default/a000146729.htm#a003281428>

Summerede rapporter

Med PROC TABULATE kan du beregne, gruppere og danne rapport med tilpasset layout.

<pre>proc tabulate data = orion.sales format = commax12.; class country; table country , n; run;</pre>	Frekvenstabel pr. land.
<pre>proc tabulate data = orion.sales format = commax12.; class country gender; var salary; table country , gender * salary; run;</pre>	Krydstabel med summer.
<pre>proc tabulate data = orion.sales format = commax12.; class country gender; var salary; table all country all , (gender all) * salary * (sum mean) / box = 'Lønstatistik Orion Star'; keylabel sum = 'Sum' mean = 'Gennemsnit' all = 'Total'; label country = 'Land' gender = 'Køn' salary = 'Løn'; run;</pre>	<p>Krydstabel med summer med layoutstyring.</p> <p>Skift overskrift på beregninger med keylabel statement.</p>

Tilgængelige beregninger i PROC TABULATE

SAS 9.1.3 <http://support.sas.com/onlinedoc/913/getDoc/da/proc.hlp/a000146762.htm#a000083741>

SAS 9.2 <http://support.sas.com/documentation/cdl/en/proc/61895/HTML/default/a000146762.htm#a000083741>

Summerede data

Med PROC MEANS kan du beregne og gruppere data, som du gemmer i en tabel.

<pre>proc means data = orion.sales noprint nway; class country; var salary; output out = sales_total (drop = _type_) sum = salary_sum mean = salary_avg; run; proc means data = orion.sales noprint; class country gender; var salary; output out = sales_total sum = salary_sum mean = salary_avg; run; proc print data = sales_total noobs; where put(_type_,binary2.) = '01'; run; proc print data = sales_total noobs; where put(_type_,binary2.) = '10'; run; proc print data = sales_total noobs; where put(_type_,binary2.) = '11'; run;</pre>	<p>NOPRINT: Dan ikke rapport. NWAY: Ingen grand total.</p> <p>Summeret tabel med én grupperingskolonne.</p> <p>Summeret tabel med flere grupperingskolonner. Resultattabellen indeholder alle summeringsniveauer.</p> <p>Sum pr. køn.</p> <p>Sum pr. land.</p> <p>Sum pr. land og køn.</p>
---	--

WHERE statement

Anvend WHERE statement i DATA step eller PROC step til begrænsning af rækker ved indlæsning af data:

```
WHERE sex = 'F' and age > 15;
```

```
WHERE sex = 'F'; WHERE ALSO age > 15;
```

Regler

- Skal være inde i et DATA eller PROC step.
- Kun ét WHERE statement pr. step.
- Et eller flere WHERE ALSO pr. step.
- Man kan bruge parenteser og SAS-funktioner til komplekse udtræk.

Eksempel på anvendelse af WHERE statement

Eksempel	Kommentar
<code>where gender ne 'M' and salary >= 50000;</code>	
<code>where gender ne 'M' or salary >= 50000;</code>	
<code>where country = 'AU' or country = 'US';</code>	
<code>where country = 'AU' or 'US';</code>	Alle rækker medtages.
<code>where country ne 'AU' and country ne 'US';</code> <code>where country not in ('AU', 'US');</code>	Brug IN-operator ved mere end en værdi.
<code>where country ne 'AU' or country ne 'US';</code>	Alle rækker medtages.
<code>where salary between 50000 and 100000;</code> <code>where 50000 <= salary <= 100000;</code>	
<code>where salary not between 50000 and 100000;</code>	
<code>where employee_id is null;</code> <code>where employee_id is missing;</code>	Giver samme resultat mod SAS-tabeller.
<code>where job_title contains 'Rep';</code>	

<code>where job_title like '%Rep%';</code>	
<code>where name like 'T_M%';</code>	
<code>where name =: 'T';</code>	Start med T.
<code>where name in: ('T','U');</code> <code>where substr(name,1,1) in ('T','U');</code>	Start med T eller U.
<code>where upcase(gender)='F' and (age=11 or age=10);</code>	Kun 10 og 11 årige piger.
<code>where upcase(gender)='F' and age=11 or age=10;</code> <code>where (upcase(gender)='F' and age=11) or age=10;</code>	11 årige piger og <u>alle</u> 10 årige.
<code>where (gender = 'F' and age = 11) or</code> <code>(gender = 'M' and age = 10) or</code> <code>salary >= 1000000;</code>	

SAS Data set options

Options kan bruges i DATA og PROC step for en tabel.

Her er de mest anvendte options.

<pre>data Employee_DK; set Employee_ALL(firstobs = 10 obs = 40 where = (country = 'DK' rename = (country = Land keep = (Land Employee_Id in = A); run;</pre>	<ul style="list-style-type: none"> • Læs fra række 10. • Læs højst til række 40. • Læs kun rækker med DK. • Omdøb kolonnen Country til Land. • Indlæs kun nævnte kolonner. • Opret styringskolonne A.
<pre>proc print data = Employee_ALL (firstobs = 10 obs = 40 where = (country = 'DK' rename = (country = Land keep = (Land Employee_Id); run;</pre>	<ul style="list-style-type: none"> • Læs fra række 10. • Læs højst til række 40. • Læs kun rækker med DK. • Omdøb kolonnen Country til Land. • Indlæs kun nævnte kolonner.

Alle SAS Data set options

SAS 9.1.3 <http://support.sas.com/onlinedoc/913/getDoc/da/lrdict.hlp/a002295655.htm>

SAS 9.2 <http://support.sas.com/documentation/cdl/en/lrdict/61724/HTML/default/a002295655.htm>

SAS-funktioner

Anvend funktioner i DATA step eller PROC step:

```
WHERE lowercase(sex) = 'f';
```

```
Navn=PROPCASE(name); /* kun DATA step */
```

Regler

- Der er altid parentes () efter funktionsnavnet.
- Der er 0, 1 eller flere argumenter.
- Argumenter adskiller du med kommaer.
- En funktion returnerer altid en værdi.

Eksempel på anvendelse af funktioner

Resultat	Eksempel	Argumentværdi
Hugo Jensen	Navn=PROPCASE(Name);	hugo jensen
HUGO JENSEN	Navn=UPCASE(Name);	hugo jensen
Hugo Jensen	Navn=CATX(' ', Fnavn, Enavn);	Hugo Jensen
100	Total=SUM(Revenue, Expenses);	72 28
15	Avg=MEAN(Q1, Q2, Q3);	10 . 20
1	Missing=NMISS(A, B, C);	1 . 7
Antal dage fra 1. januar 1960 til dags dato.	Now=TODAY();	
2009	Year=YEAR(Hiredate);	17956
1	Quarter=QTR(Hiredate);	17956
2	Month=MONTH(Hiredate);	17956
28	Day=DAY(Hiredate);	17956
7	Weekday=WEEKDAY(Hiredate);	17956

9	Week=WEEK(Hiredate, 'v');	17956
Antal år mellem dags dato og 28. februar 2009 inkl. decimaler.	Years=INTCK('Year',TODAY(),Hiredate);	dags dato 17956
17956	Date=MDY(month,day,year);	2 28 2009

Alle funktioner

SAS 9.1.3: <http://support.sas.com/onlinedoc/913/getDoc/en/lrdict.hlp/a000245860.htm>

SAS 9.2: <http://support.sas.com/documentation/cdl/en/lrdict/59540/HTML/default/a000245860.htm>

SAS formater

Tilknyt formater til kolonner i DATA step eller PROC step:

```
FORMAT      Salary      commax20.2
            Name        $20.
            Hiredate     ddmmmyyp10.;
```

Regler

- Der er altid punktum i et format.
- Tallet foran punktum er længden og tallet efter punktum er decimaler (hvis numerisk).
- \$-tegn angiver, at det er et tekstformat.
- Tekstformater kan bruges til tekstkolonner, numeriske formater til numeriske kolonner.

Hyppigt anvendte formater

Værdi	Format	Formateret værdi	Forklaring
32456.2	8.2	32456.20	
32456.2	commax10.2	32.456,20	Dansk komma notation.
32456.2	comma10.3	32,456.200	
Hugo Jensen	\$20.	Hugo Jensen	
Hugo Jensen	\$7.	Hugo Je	
-73	ddmmmyyp10.	20.10.1959	
0	ddmmmyyd10.	01-01-1960	
18000	ddmmmyyb10.	13 04 2009	
18000	dandfwdx18.	13. april 2009	
101451234	z10.	0101451234	Indsætter foranstillede nuller.
0.55	percent9.	55 %	
1125.5	eurox10.2	€1.125,50	Dansk komma notation.
1125.5	euro10.2	€1,125.50	
Jensen	\$char20.	Jensen	Beholder foranstillede blanke.

Jensen	\$upcase20.	JENSEN	
18000	dandfdwn7.	mandag	
18000	dandfmm9.	april	
18000	dandfwkx31.	mandag, den 13. april 2009	

Alle formater

SAS 9.1.3: <http://support.sas.com/onlinedoc/913/getDoc/en/lrdict.hlp/a001263753.htm>

SAS 9.2: <http://support.sas.com/documentation/cdl/en/lrdict/59540/HTML/default/a001263753.htm>

Output Delivery System (ODS)

Ofte anvendte ODS-destinationer:

- LISTING SAS Output-vindue
- HTML Danner HTML-fil, vises i SAS' Results Viewer
- PDF Danner PDF-fil, vises Adobe PDF-reader
- RTF Danner RTF-fil, vises MS-Word
- XML Danner XML-fil, kan bl.a. åbnes i SAS Web Report Studio

<pre>ods _all_ close; ods listing; ods html file='c:\temp\Report.html' style=EGDefault; ods pdf file='c:\temp\Report.pdf'; ods rtf file='c:\temp\Report.rtf'; ods xml file='c:\temp\Report.xml'; proc freq data = orion.sales; tables Country; run; ods xml close; ods rtf close; proc gchart data = orion.sales; vbar3d country / sumvar=salary type=mean; run; quit; ods _all_ close; ods listing; ods html;</pre>	<p>Luk alle ODS-destinationer.</p> <p>Åbn flere ODS-destinationer og angiv fysisk placering og filnavn.</p> <p>Luk 2 af de 5 åbne ODS-destinationer.</p> <p>Åbn mindst én ODS-destination, så efterfølgende rapporter vises uden ODS-statement.</p>
---	---

Information om SAS software

Generel hjælp

- SAS OnlineDoc 9.1.3 for the Web <http://support.sas.com/onlinedoc/913/docMainpage.jsp>
- SAS Customer Support Center <http://support.sas.com>
- Alle SAS produkter og løsninger <http://www.sas.com/products/index.html>

Danmark

- SAS Institute Danmark <http://www.sas.com/offices/europe/denmark/index.html>
- SAS nyhedsbrev <http://www.sas.com/offices/europe/denmark/2801.html>
- SAS kurser www.saskurser.dk

Konferencer

- SAS Forum Danmark www.sasforum.dk
- Analytics, Danmark <http://www.sas.com/events/aconf/>
- SAS Global Forum, USA <http://support.sas.com/events/sasglobalforum/previous/online.html>
- Data Mining, USA <http://www.sas.com/events/dmconf/>
- Forecasting, USA <http://www.sas.com/events/fx/>

Dokumentation

SAS9 documentation kan downloades <http://support.sas.com/v9doc>

Anbefalede bøger (kan downloades gratis som PDF):

ISBN	SAS9 Documentation
*978-1-59047-754-0	Base SAS 9.1.3 Procedures Guide, 2nd Ed. Volumes 1-4
*978-1-59047-840-0	SAS 9.1.3 Language Reference: Concepts, 3rd Ed. Vol 1-2
*978-1-59994-098-4	SAS 9.1.3 Language Reference: Dictionary, 5th Ed. Vol 1-4
978-1-59047-922-3	SAS(R) Certification Prep Guide: Base Programming for SAS9
978-1-58025-791-6	Step by Step Programming with Base SAS Software
	SAS 9.1 Companion for (Operating System)
	SAS/ACCESS 9.1 for (Database)