

**Block 1: From questionnaire to SPSS saved file**

**1.3.3.2 Introduction to SPSS syntax**

[Updated **28 March 2013** from earlier version of **3 Sep 2010**: screenshots still **SPSS 18**]

**Previous session:** [1.3.3.1 Preparing the ground](#)

Readers of the following notes are assumed to be familiar with basic operations in **Windows** and **MS-Word** (eg opening and saving files, dragging elements around, dragging edges and corners of panes, copying and pasting elements or text etc.).

**SPSS syntax**

Although SPSS now has a graphic user interface (GUI) with drop-down menus for inveterate mouse clickers, SPSS also has its own special English-like language known as **syntax** (the set of instructions telling SPSS what to do). For basic operations, SPSS syntax is much quicker and easier to use and the language is intuitive and easy to understand, eg:

```
FREQUENCIES      SEX HAPPY .
CROSSTABS        SEX BY HAPPY .
```

Individual SPSS programs are called and driven by **commands** which must start in the **first column** of the line. Within a command there can be one or more **sub-commands** which can start anywhere after the command, continuing on following lines if necessary, but **not in the first column**. SPSS also has certain reserved **keywords** which cannot be used for anything else. All commands (just like sentences in English) must end with a **full stop (period)**.

From release 17 onwards, SPSS automatically allocates colour-coded text to distinguish **commands**, **subcommands** and **keywords** when writing syntax files. The syntax examples in this document tally with the colours used in SPSS: the exercises in these tutorials have been prepared with SPSS/PASW 18 (and some with SPSS 19) but they all work with SPSS 15 (and probably also SPSS 11) but the [comparative screenshots](#) will be different.

**commands**                      **specifications** and / **subcommands**                      **keywords**

In my examples I use the same colour-coding as SPSS. SPSS is case-insensitive (except for user-written comments, labels and documnetnts) and you can type your commands in either UPPER or lower case. If you use the menus which appear as you type commands into the syntax file, everything will be displayed in UPPER CASE.

I use **UPPER CASE** to introduce or explain general formats for each **command** or **sub-command** and **lower case** for actual **SPSS syntax** needed to run the analyses. Thus the above commands would appear as:

```
frequencies sex happy .
```

.. will produce:

Statistics			
		sex Sex of Respondent	happy Q.53 How [happy] are you these days?
N	Valid	932	926
	Missing	0	6

... which tells you how many cases there are for each variable, followed by two frequency tables, one for each variable

**sex Sex of Respondent**

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	1 Men	389	41.7	41.7	41.7
	2 Women	543	58.3	58.3	100.0
	Total	932	100.0	100.0	

**happy Q.53 How [happy] are you these days?**

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	1 Not too happy	57	6.1	6.2	6.2
	2 Fairly Happy	516	55.4	55.7	61.9
	3 Very happy	353	37.9	38.1	100.0
	Total	926	99.4	100.0	
Missing	System	6	.6		
Total		932	100.0		

**crosstabs sex by happy .**

[by is a reserved keyword in SPSS]

**Case Processing Summary**

	Cases					
	Valid		Missing		Total	
	N	Percent	N	Percent	N	Percent
sex Sex of Respondent * happy Q.53 How [happy] are you these days?	926	99.4%	6	.6%	932	100.0%

... which tells you how many cases are valid or missing for the combination of both variables, followed by a contingency table containing the counts for each combination.

**sex Sex of Respondent \* happy Q.53 How [happy] are you these days? Crosstabulation**

Count

		happy Q.53 How [happy] are you these days?			Total
		1 Not too happy	2 Fairly Happy	3 Very happy	
sex Sex of Respondent	1 Men	24	230	131	385
	2 Women	33	286	222	541
Total		57	516	353	926

An example using a **subcommand** would be:

**crosstabs sex by happy /cells row .**

[by and row are reserved keywords in SPSS]

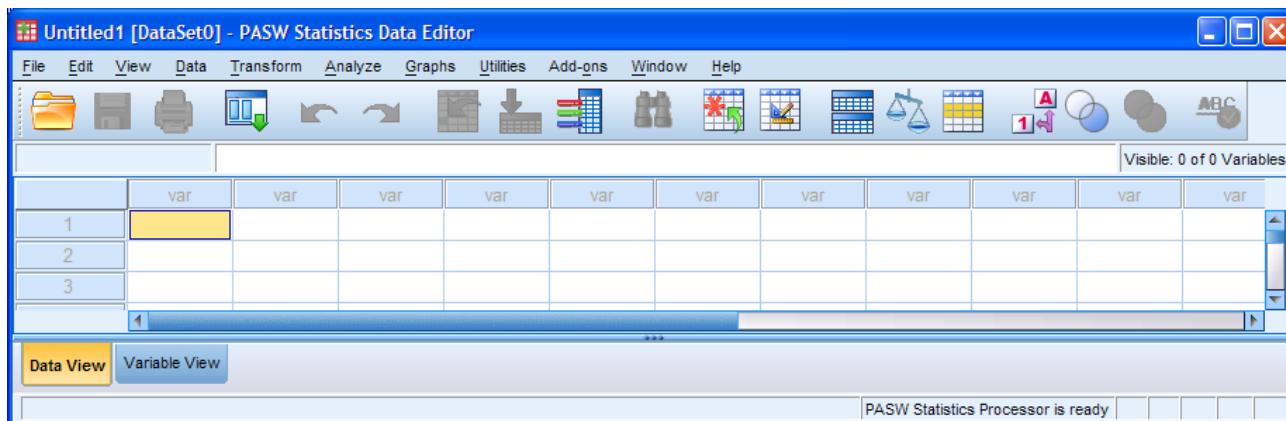
.. which produces the same table, but with **row** percentages (which make it easier than simple counts to compare men and women).

**sex Sex of Respondent \* happy Q.53 How [happy] are you these days? Crosstabulation**

% within sex Sex of Respondent

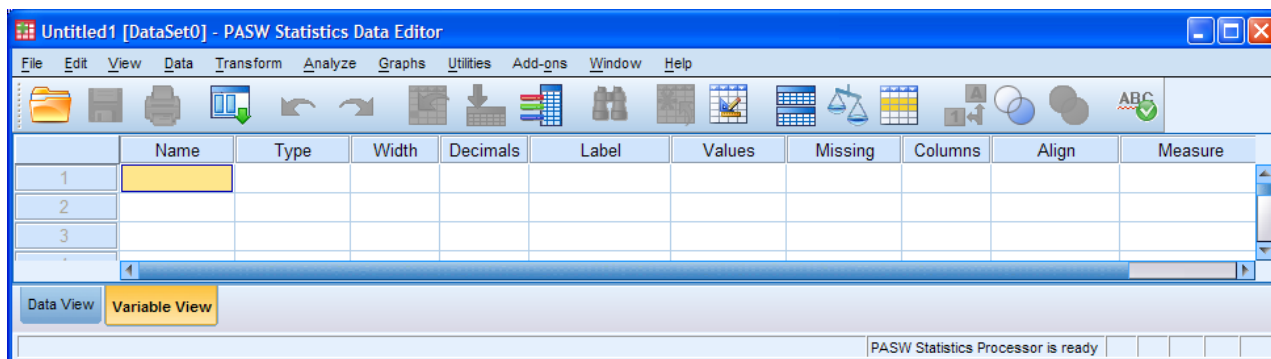
		happy Q.53 How [happy] are you these days?			Total
		1 Not too happy	2 Fairly Happy	3 Very happy	
sex Sex of Respondent	1 Men	6.2%	59.7%	34.0%	100.0%
	2 Women	6.1%	52.9%	41.0%	100.0%
Total		6.2%	55.7%	38.1%	100.0%

SPSS works with its own **Data Editor**. When you first open SPSS the Data Editor will be blank:



Initial Data Editor in **Data View**

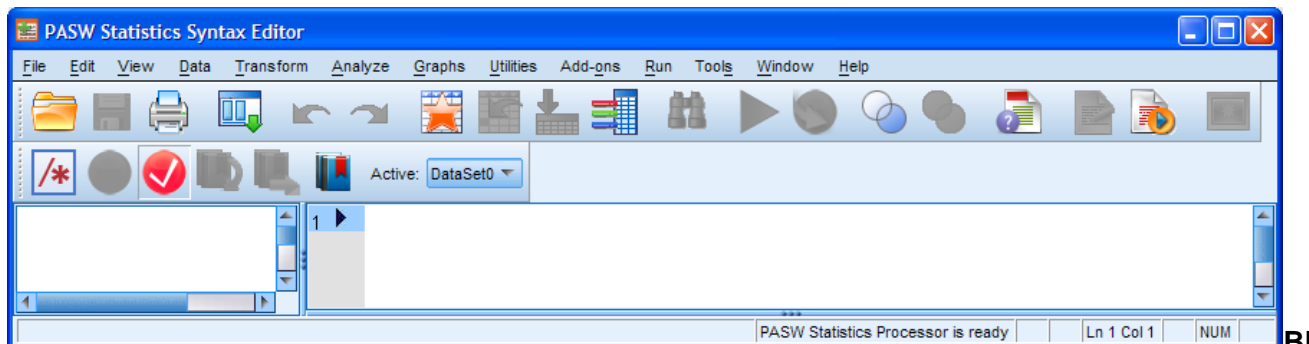
... in which your **variables** will be arranged in the columns and the **cases** in the rows. This is exactly the same format as the data matrix I described in the first tutorial and also as used in Excel spreadsheets (from which SPSS can also import data direct, but that's another story). When starting from scratch this will be empty, but it can also import previously saved SPSS files or files containing data from elsewhere.



Initial Data Editor in **Variable View**

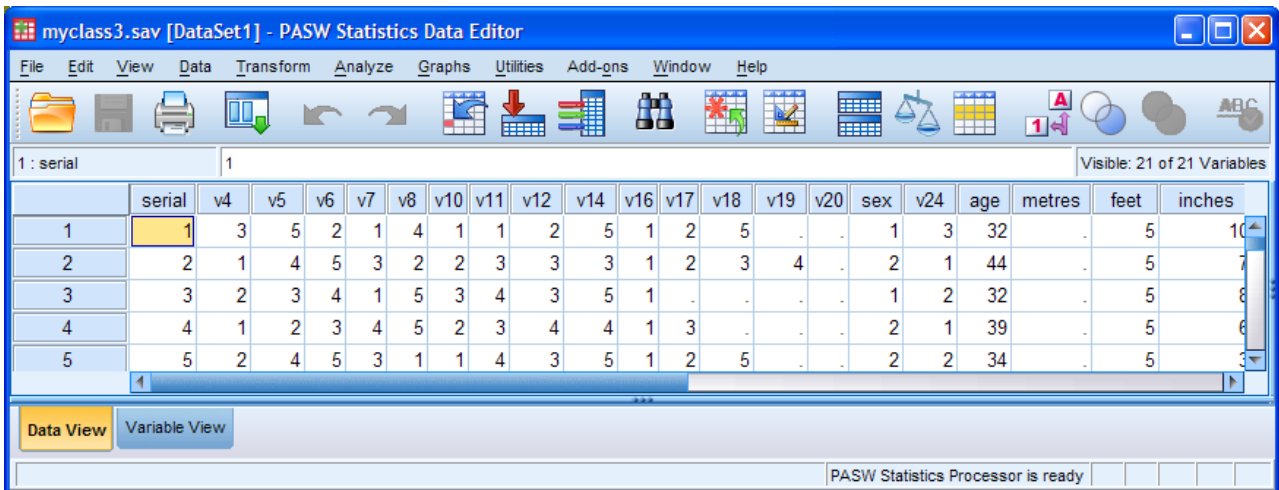
... in which the variable names will appear in the **Name** column and various additional information about them in the other columns.

SPSS commands are written in a special **Syntax Editor** which not only colour codes as you type, but also flags dubious or incomplete syntax in **red** (a bit like Spellcheck)

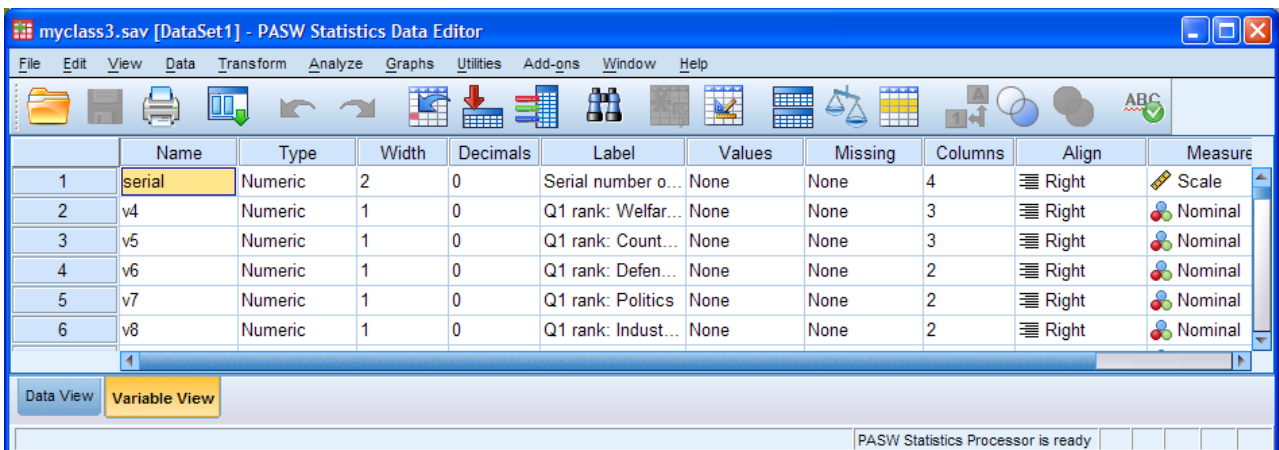


Blank SPSS Syntax Editor

Taking the pre-course questionnaire as an example, once you have used SPSS to define your variables, add dictionary information (missing values, variable labels, value labels) and then read in the raw data from file [myclass.txt](#), the SPSS Data Editor in **Data View** will look something like this:



... and in **Variable View** like this:

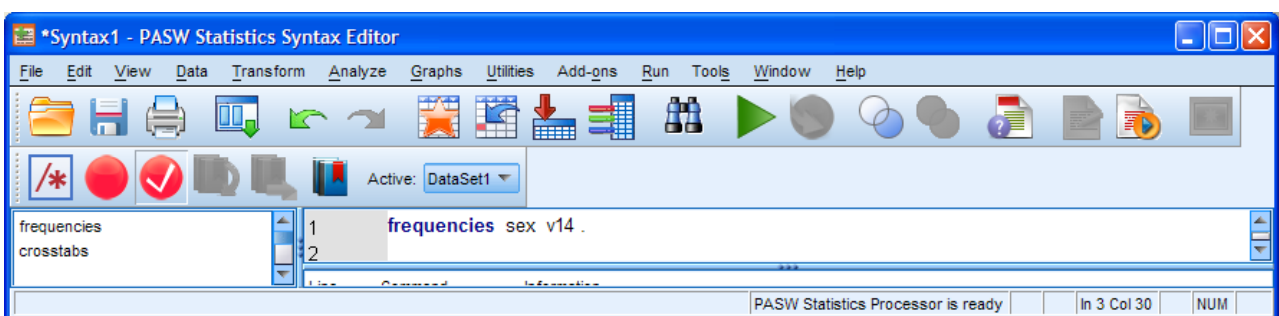


How many male and female students there are in the sample? (Question 5: **sex** ) How satisfied or dissatisfied are they with the running of the National Health Service? (Question 3: **v14** )

We can answer these questions by opening a new syntax file and writing::

**frequencies sex v14 .**

You write your instructions in the right hand pane of the **syntax window**.



... and then run the command to produce.

**sex Q5: Sex of respondent**

	Frequency	Percent	Valid Percent	Cumulative Percent
Valid Male	50	29.6	29.6	29.6
Female	119	70.4	70.4	100.0
Total	169	100.0	100.0	

**v14 Q3: Satisfaction with running of NHS**

	Frequency	Percent	Valid Percent	Cumulative Percent
Valid Very satisfied.	2	1.2	1.2	1.2
Quite satisfied.	13	7.7	7.8	9.0
Neither	27	16.0	16.3	25.3
Quite dissatisfied	64	37.9	38.6	63.9
Very dissatisfied	60	35.5	36.1	100.0
Total	166	98.2	100.0	
Missing System	3	1.8		
Total	169	100.0		

Are there any differences between them on satisfaction - dissatisfaction? One way of answering this question is with a contingency table:

**crosstabs sex by v14 /cells row<sup>1</sup> .**



**sex Q5: Sex of respondent \* v14 Q3: Satisfaction with running of NHS Crosstabulation**

% within sex Q5: Sex of respondent

		v14 Q3: Satisfaction with running of NHS					
		Very satisfied.	Quite satisfied.	Neither	Quite dissatisfied	Very dissatisfied	Total
sex Q5: Sex of respondent	Male	0.0%	8.2%	20.4%	32.7%	38.8%	100.0%
	Female	1.7%	7.7%	14.5%	41.0%	35.0%	100.0%
	Total	1.2%	7.8%	16.3%	38.6%	36.1%	100.0%

<sup>1</sup> tells SPSS to display row percent in the cells

## Types of commands

There are four basic types of command:

### Data definitions . . .

tell SPSS where to find data, and how to interpret them, in order to create variables. They also allow the labelling of data. (eg. **DATA LIST**, **GET FILE**, **MISSING VALUES**, **VARIABLE LABELS**, **VALUE LABELS**)

### Data transformations . . .

change the values of existing variables or create new variables. They can also be used to restrict analysis to a subset of cases. (eg. **RECODE**, **COMPUTE**, **SELECT IF**)

### Procedures . . .

read the data and generally produce output in the form of tables, charts and statistics. (eg. **DISPLAY**, **LIST**, **FREQUENCIES**, **CROSSTABS**, **MEANS**, **T-TEST**)

### Utilities . . .

allow you to insert documentation, titles, subtitles and comments and to give information about the contents of a data file (eg: **DOCUMENT**, **TITLE**, **SUBTITLE**, **COMMENT**)

Commands are read sequentially, so the order in which you enter them is important.

Whenever SPSS is called, it must have a Data Editor to be working on. This file is known as the **active file** and must have at least one variable in it. It can be a blank file that you complete yourself or an existing one that you open and download/import into it. You will need to tell SPSS where to find data and how to read them before you can transform them or carry out any analysis. SPSS will only read the data, perform transformations and produce output when it encounters a **procedure** command or the special command **EXECUTE**.

A sequence of commands will generally take the form:

**Data Definitions**

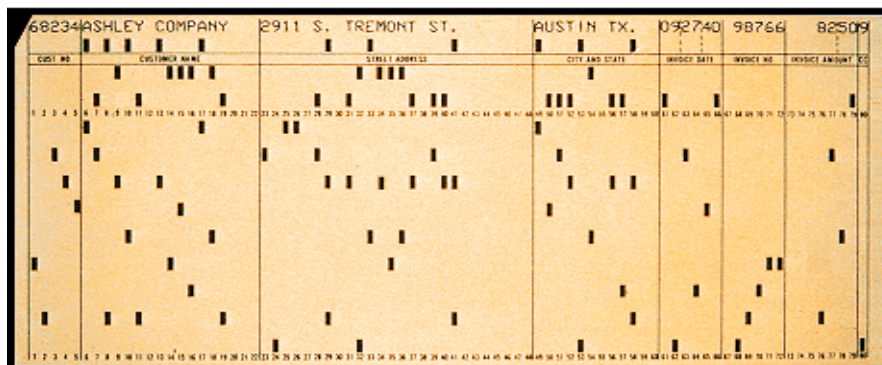
**Data Transformations**

**Procedures**

**Utilities** may be entered where appropriate.

In the early days, SPSS syntax used 80-column Hollerith cards in which columns 1-15 were reserved for **commands**, columns 16-72 for **specifications** and **sub-commands** and columns 73-80 reserved for numbering the cards in sequence. This was in case someone dropped a tray of anything up to 2,000 cards and got them all mixed up. I've seen it happen and once did it on purpose in front of the class. Gasps of dismay and horror all round, but they got the point!

From Computer Desktop Encyclopedia  
© 2000 The Computer Language Co. Inc.



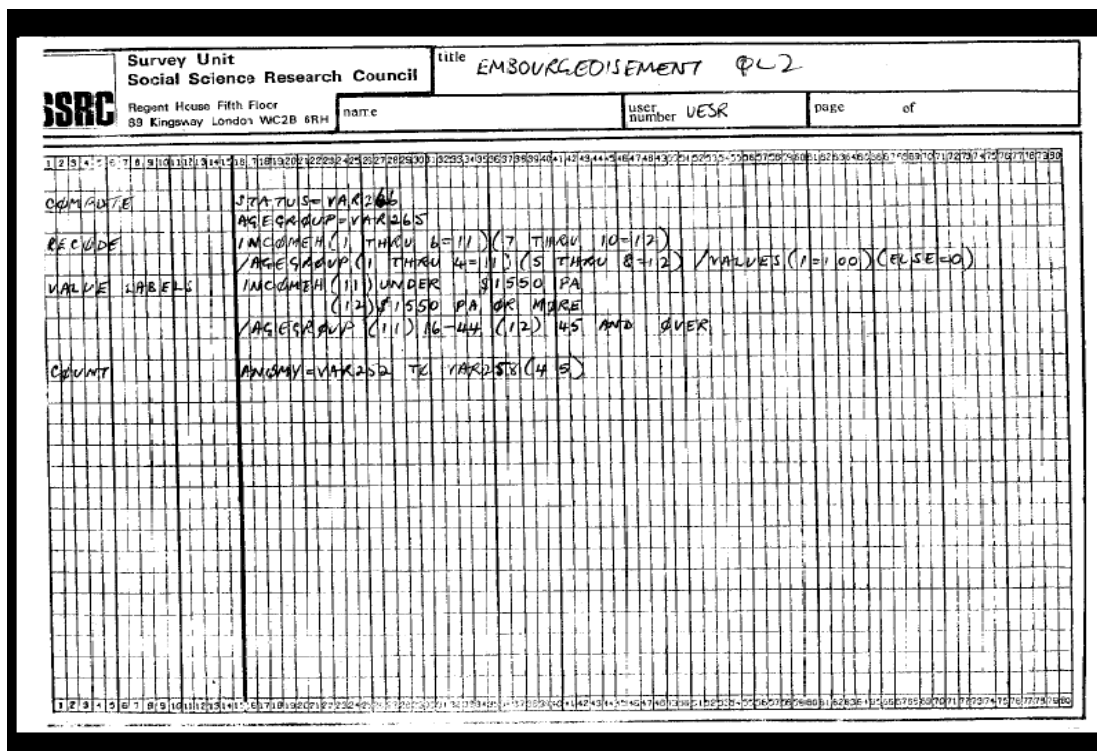
80-column  
Hollerith card

A typical small SPSS job would look something like this:


1	16	77
GET FILE	MYCLASS.SYS	001
FREQUENCIES	SEX V14	002
CROSSTABS	SEX BY V14	003
FINISH		004

[Numbering is a bit pointless on such a small job, but some runs would have several hundred cards.]

To make sure jobs were correctly written up in draft before punching, it was useful to have special coding sheets with a heavy line between columns 15 and 16:



Draft SPSS run to test Goldthorpe-Lockwood *embourgeoisement* theory

This layout is no longer needed because the SPSS knows that a **command** starts in column 1 of a line (and that anything which does not start in column 1 must therefore be a **specification**, a **sub-command** or a continuation line). However, it is still helpful for beginners to keep their **commands** and **sub-commands** visually separate inset their sub-commands and specifications by using several spaces or the [TAB] key .

Thus it is difficult to define hard and fast rules, but the following general rules apply:

1. **Commands** must begin in column 1 of the line in which they appear (i.e. against the left hand margin.)
2. **Sub-commands** can be continued over more than 1 line, but continuation lines must be indented by at least 1 column or must use a + sign if the text is too long to fit on one line.
3. **Commands** can be entered in either **UPPER** or **lower** case.
4. Parts of **commands** must be separated by **spaces** or by symbols such as ' / ', ' = ' or ' , ' where appropriate.

Even if you use the drop-down menus, SPSS automatically generates and saves the syntax, and (optionally) displays it in the output.

Some useful commands (eg **LIST** and **DISPLAY**) are not available via the drop-down menus, but still get colour-coded.

I have occasionally used additional colour-coding for **critical comments** on the way SPSS works (or sometimes does not) and **[bold black]** for special (combinations of) keys on the keyboard (eg **[CTRL]+C** to copy, **[CTRL]+V** to paste **[CTRL]+S** for interim save and **[CTRL]+R** to run SPSS).

For clarity, except for very short sets of commands, I usually write my syntax outside SPSS in a \*.doc or \*.txt file and then copy it across to SPSS. I also indent sub-commands, specifications and continuation lines using tabs, but PASW 18 and SPSS 19 sometimes have problems interpreting these if copied from Word (when the original syntax has in turn been copied into Word from my old WordStar4 files or from EDT programs for the Vax), so it's better to use spaces. Once you get the hang of it, you can simply use one or more spaces in column 1.

Labels in SPSS syntax need to be enclosed in single primes: if you write syntax in Word you need to turn off the "smart quotes" option. Moreover SPSS is not case specific except for user-written labels, so you can work entirely in lower case.

#### **End of tutorial:**

Now let's have a shot at writing syntax for real.

**Next session:** [1.3.3.3 First shot at writing SPSS syntax](#)

[\[Back to Block 1 menu\]](#)