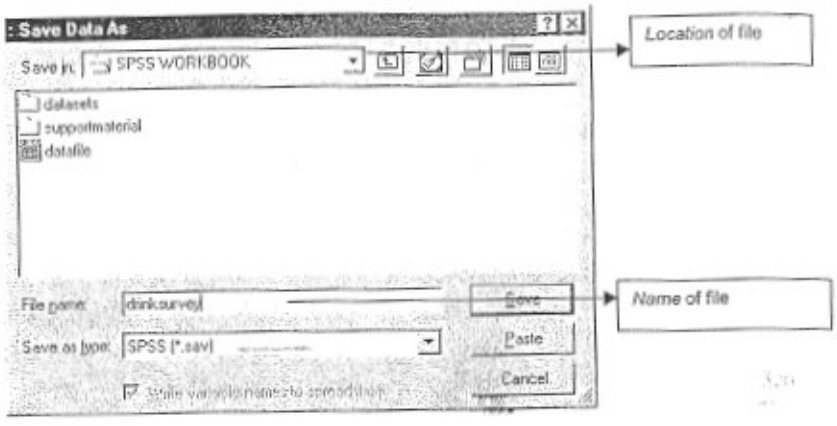
Figure 1.3c **Save Data** dialog box



Be careful that the version of the data file you save is the version you want to *keep*! For instance, a common mistake students make is to select a subset of cases (say, only the males in the sample) and then save the datafile later on without 'turning off' the selection. The next time they come to analyse the data file, they discover to their consternation that only the selected cases are left! A 'safety net' is to save your later version under a slightly different file name; for example, **vers2**. Then, if you realise you have made a mistake, you can always go back and resurrect the earlier version of the data file: **vers1**.

Labelling variables

An SPSS data file requires both the data and information about the variables. The data is visible in the **Data View** window, and the information about the variables is visible in the **Variable View** window of the Data Editor.

When the SPSS Data Editor is in **Data View** format, each column in the grid represents a *variable*. SPSS requires each that each variable must have a unique 'name' that identifies that variable separately from any other variable. These variable names should be intelligible and adhere to the SPSS rules for naming variables. – SPSS requires that variable names should be short words (limited to a maximum of eight characters). It is good practice for these variable labels to be one of three types.

- *Self-explanatory* labels stating what information the variable signifies (for example, **sex**, **age**, **faculty**)
- An *acronym* that helps jog the researcher's memory about what the variable is (for example, **hoh** for 'Head Of Household' or **denom** for a religious denomination coding)
- Just a list of *letters and numbers in sequence* (for example, *V1, V2, V3, V4, ... Vn*).

Which is best depends largely upon personal preference and convenience. Some researchers use the question numbers as variable names, for example, **que1** as a variable name for the first question in a survey.

There are some conventions that must be followed when assigning variable names.

- A variable name must begin with a *letter* (not a number)
- A variable name cannot have a *blank space* within the name (for example **dog day** would not be acceptable but **dog_day** would be acceptable)
- A variable name cannot be more than *eight characters* (a character is a letter, number or the symbols *@*, *#*, *_* or *$*)
- A variable name cannot have the *special characters* !, ? and * or other special characters except those listed in the previous bullet point
- A variable name cannot be one of the words SPSS uses as *keywords* (for example, AND, NOT, EQ, BY and ALL)
- A variable name cannot end with a *full stop*.

Coding

The data that one wishes to put into a computer package for analysis has to be transformed from its completely non-computerised form (which could be answers written onto a questionnaire, entries on a form or application, a personnel or student record file, etc.) into a shape that can be input into a computer. Usually, this means that the information is converted into number values – a process called *coding*. Coding operations can be carried out in several ways depending on the type of information you are dealing with. There are two types of data – *quantitative* and *qualitative*.

Quantitative data

If the information is already in number form, the coding is fairly simple; just the *direct* transfer of the numeric value. For instance, on the 'Drinking questionnaire', Robert is 45 years old, the code for **age** is simply **45**. Similarly, the codes for pints of beer and cider, glasses of wine and measures of spirits are all coded directly on the 'Drinking questionnaire'.

Numbers with decimal points can be coded with or without the decimal point. For instance, Fred answered on the 'Drinking questionnaire' that he spent £4.55 on alcohol last weekend. This could be coded as **4.55** or **0455** (the first case is the version of coding we are using with the questionnaire). The default for SPSS is eight digits with two places to the right of the decimal point. If more digits or decimal places are required this can be changed in the **Variable View** window of the Data Editor (which is explained below).

These days, thanks to the power of modern computers and the ability to manipulate data after input into SPSS, one should always code in the most detail practicable. When coding numbers, this means that the number is the code – you should not amalgamate the number values together into larger categories at the coding stage. The reason for this is that, if the values are lumped together *before* input into SPSS, it will be impossible to get back to the detailed information of the real number. That is, you have unnecessarily thrown away some information that could prove vital at some future date. As you will see when you look at data manipulation in Module 3, it is a simple matter to aggregate numbers together after data has been input into SPSS.

For instance, as an example of bad coding procedure, some people would have coded people's ages on the 'Drinking questionnaire' *directly into categories*.

1 = Younger than 18
2 = 18 to 24
3 = 25 to 40
4 = 41 to 64
5 = 65 or older

We would advise that you avoid prematurely categorising quantitative data such as these. What if, later on, you discover that you really need to compare the drinking habits of those aged less than 21 with those aged between 21 and 29? It will be impossible because your age categories will not allow it. If you had coded age directly, you could easily amalgamate up into the categories required for your analysis. It is always possible to move from a *more* detailed coding to a *less* detailed coding: the reverse cannot be done! (If you are familiar with scaling or levels of measurement, you may have noted that so far we have been talking about the *quantitative* levels of measurement: interval and ratio scales. Levels of measurement will be discussed in more detail at the beginning of Module 2).

Qualitative data

Quite often, the information you want to code may not be number values, but instead in the form of *categories*.

- These can be *mutually exclusive*, binary 'black or white' categories where one category implies the absence or opposite of the other; for example, *YES* or *NO*
  or (from the 'Drinking questionnaire') *1 = Male, 2 = Female*
- There can also be *more than two* categories – a set of categories where each is a different type of a common characteristic:
  for example, Car Colours: *BLACK/RED/BLUE/GREEN/YELLOW*, etc.
  or (from the 'Drinking questionnaire')
  Faculty: *Social Sciences = 1; Arts = 2; Science = 3; Engineering = 4; Other = 5*

Here, of course, the number codes are only labels for the categories and do not have any arithmetic value in themselves (for example, *Engineering*, coded '4', is not twice as much a faculty such as *Arts*, coded '2'!). These sorts of categorial data can be described as being at the *nominal* level of measurement.

There is an in-between situation where you can have categories that fall into an *increasing* or *decreasing* order. The number codes used to represent the categories show the ordering effect but do not literally represent a true amount or quantity. For instance, people could be asked to rank a sensation by its pain:

tickles/uncomfortable/hurts/hurts a lot/excruciating!!!
1 2 3 4 5

We could agree that *'only tickles'* (1) is less pain than *'uncomfortable'* (2), which is less pain than *'hurts'* (3) and so on ($1 < 2 < 3 < 4 < 5$), but we would probably find it hard to agree that three *'tickles'* and an *'uncomfortable'* equals one *'excruciating!!!'* ($1 + 1 + 1 + 2 = 5$)!

In the 'Drinking questionnaire' there is an example of this sort of **ordinal** measurement where the numbers imply a decreasing or increasing order, but the numbers themselves signify only less or more and not any definite amounts:

Do you

1 = 'Never drink alcohol', 2 = 'Drink rarely', 3 = 'Drink moderately', 4 = 'Drink frequently', 5 = 'Drink heavily'?

(The nominal and ordinal levels of measurement will be discussed in more detail at the beginning of Module 2.)

'String' data

While numbers are normally used to record information, it is possible to enter in letters – or, more exactly, *alphanumeric codes*. For instance, there could be good reasons to enter in people's own names; we have done this on the 'Drinking questionnaire' with the variable *name* which has respondents' names. Sometimes an interview schedule may contain 'open-ended' questions where you might want to record exactly what a person said in response to a question *verbatim* instead of converting what they said into a number code.

These alphanumeric codes are called *string* variables (technically, they are in 'A' format). Even if the codes of a variable declared as a string variable are numbers, they cannot have arithmetic operations performed on them as normal number-based codes unless they are specially altered.

Alphanumeric codes are fairly rare, especially in datasets designed for statistical analysis. They are more common in datasets which originate from a personnel file or the like. They have been mentioned here since you may encounter them, but we recommend that you avoid setting up variables as string variable if possible.

An important note: SPSS will permit alphanumeric codes (for example, words such as respondents' names) to be entered into the Data View window only if the variable has been defined as a *string variable*. To change the variable type, switch to the **Variable View** format of the Data Editor. Click on the **Type** cell of the variable you wish to change, and alter the setting to **String**. (The use of **Variable View** is covered in detail below in the section on SPSS operations to 'label' and 'refine' a dataset.)

Refining the data set

At this point after the data has been coded and input into SPSS, you could, in theory, move straight into a statistical analysis of some sort. In practice, however, the basic dataset is usually 'refined' before we can consider it to be completely ready for analysis. This 'refining' can consist of three types of operations.

- Attaching special descriptions or labels that help explain the *form* and *content* of the dataset
- Carrying out *validation or consistency checks* to remove or control errors and missing information in the dataset
- Tagging *missing* or *invalid codes* with special labels so that these incorrect codes aren't mixed in with valid values when analyses are being carried out.

Labelling the data

Attaching special descriptions or labels that help explain the form and content of the dataset is essential. While it makes no difference to the actual numerical analyses, it is a wise practice to document the structure of a dataset and to attach descriptions of the meanings of variable names and the codes of the variables. All of this may seem clear and straightforward to you – the person who set up the dataset – but this might not be the case for someone else who will have to try and analyse the data at some future date! Also, features of the data that seem straightforward to you *today*, might not be so obvious or straightforward to you *some weeks or months later*. Consequently, SPSS allows you to 'document' a dataset as you set it up. The information on each of the variables is presented in the **Variable View** window of the Data Editor.

Variable labels

A common feature for documenting a dataset is a provision that allows one to attach a 'descriptor' or 'label' to the short variable names. The variable names are restricted to eight characters, which may be obscure and require clarification. For instance, in the 'Drinking questionnaire' there is a variable named **fac**, which may be rather vague to someone not familiar with the dataset. But, if a descriptor/label is attached – for example, **fac** = *Faculty at university or college* – the meaning of the variable becomes much more clear. These details can be entered in column Label in the **Variable View** window.

Value labels

A similar problem exists for the individual codes of a variable. Again, a common feature of many data analysis packages is a provision for attaching a 'descriptor' or label to individual codes. For instance, the variable **fac** has six codes – 1, 2, 3, 4, 5 and 9 – that go with it. It would be easy to forget what these six codes mean. But, if descriptors/labels are attached, their meanings are much clearer:

1 = *Social sciences*, 2 = *Arts*, 3 = *Science*, 4 = *Engineering*,
5 = *Other*, 9 = *Not applicable*.

Once a variable and its values are 'labelled', these descriptors will appear on any printout where the variable name or the number codes would appear. For instance, without labelling, a tabulation of **fac** for 500 cases could look like this:

Variable: **fac**	
1	57
2	186
3	98
4	132
5	4
9	23
Total	500

Unless one was very familiar with the dataset, this would be completely obscure. Now, with the descriptors, we obtain a much less vague tabulation:

Variable: **fac**, *Faculty*	
1 *Social sciences*	57
2 *Arts*	186
3 *Science*	98
4 *Engineering*	132
5 *Other*	4
9 *Not applicable*	23
Total	500

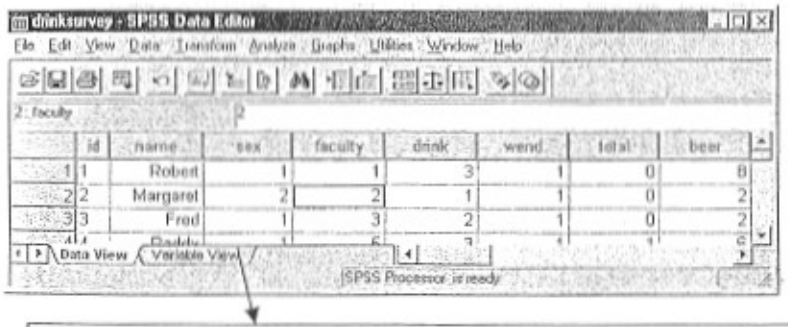
These labels do not alter the mathematical calculations of the programme in any way. The effect they do have is to make the printout much easier to understand.

SPSS operations to 'label' and 'refine' a dataset

Now, let us go through the operations required to fully label new variables in a SPSS-data file. There are a number of ways to set up variable labels and values in SPSS V10. (The procedures using SPSS V10 are slightly different to older versions of SPSS – see Appendix 2, p. 56, for examples using SPSS V9.)

As already described, the **Variable View** format of the Data Editor gives details of each of the variables whose values are entered in **Data View** format. While it is technically possible to complete some statistical procedures without details of each of the variables, it is good practice for the SPSS user to supply and complete the **Variable View** details. To change to **Variable View** format, click on the tab labelled **Variable View** at the bottom of the left-hand side of the Data Editor window (Figure 1.4a). Notice how the grid switches when you do this.

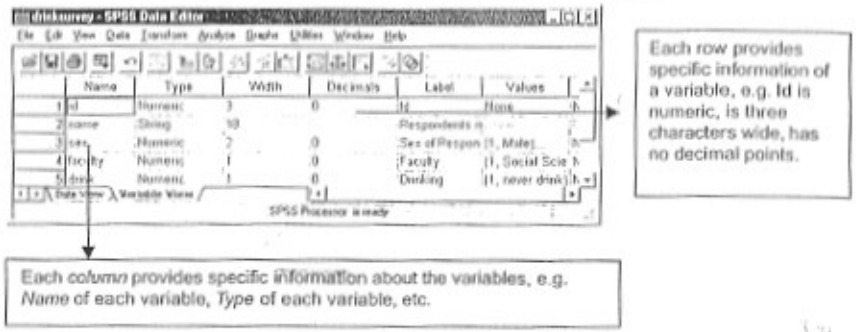
Figure 1.4a Opening **Variable View** in the Data Editor



The *rows* in **Variable View** format contain details of each variable, beginning with the variable name (Figure 1.4b). Each *column* provides specific information about the variables. For example, the **Names** of the variables are presented in the first column, **Type** of each variable is in the

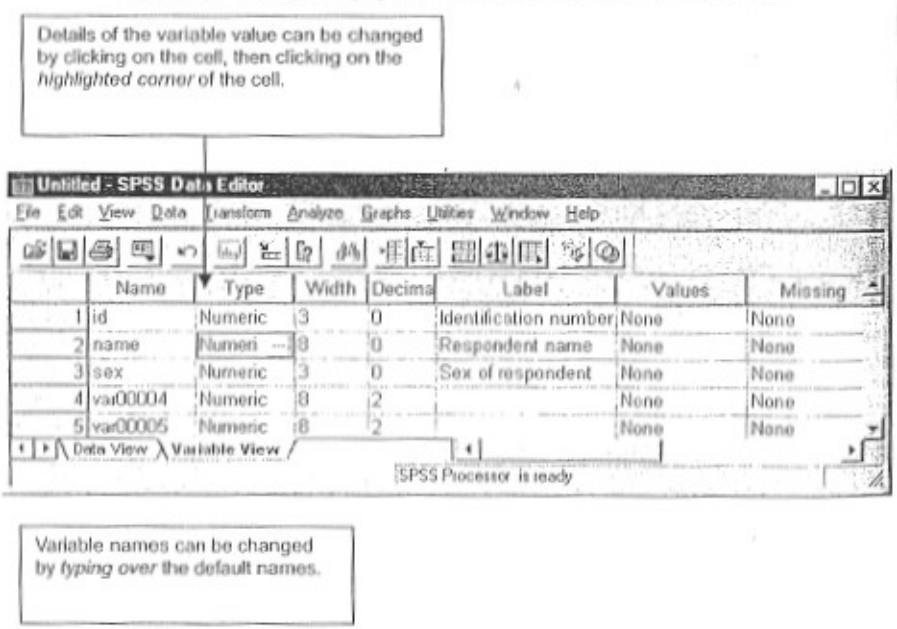
second column, etc. Each *row* provides specific information of each variable, including **Name**, **Type of variable**, **Width**, **Decimals**, **Labels**, **Values** and so on.

Figure 1.4b Variable View window of the Data Editor



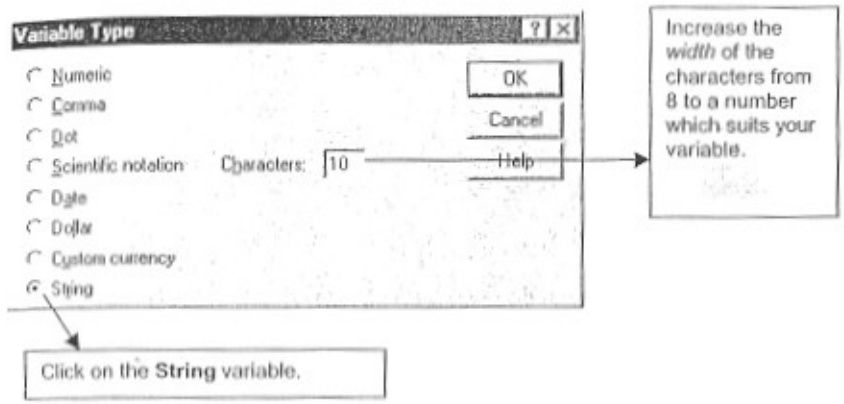
Until changed by the user, **Variable View** format presents the default aspects of the variable. For example, each variable will be **Type** '**Numeric**' with a **Width** of '8' characters, '2' decimals, no specified **Labels** or missing data, etc. Any of these characteristics can be changed in the Data Editor window when it is in **Variable View** format (Figure 1.4c).

Figure 1.4c Editing/changing the **Variable View** window of the Data Editor



Some of the default characteristics will be appropriate for each variable, but some may need to be changed. It is quite easy to change a characteristic by clicking on the cell containing the information. For example, the variable **id** is numeric, but the variable **name** is a *string* (person's name rather than number). To change the **Type**, move the cursor to the three dots in the shaded corner of the cell and click once. This will open the subwindow with eight variable types in Figure 1.4d. Click on 'string' variable, and change the width to 10 characters to accommodate longer names.

Figure 1.4d Changing **Variable Type**



SPSS gives us the facility in **Variable View** for including fuller details for each variable name by attaching a longer label to it. For example, we might want to give the variable name **wend** (for the question in the survey *Did you drink last weekend between 12 noon on Friday and last Saturday Night?*) a longer label to help us remember what **wend** is. To *label the variable name*, simply type in the full label in the cell of that variable. For example, to give more details to the variable name **wend,** move the cursor along the row containing **wend** until you reach the **Label** column. Click on this cell, and type in the full details, as shown in Figure 1.4e.

It is important to *label each of the number codes* for categorical variables. To do so, click on the cell of the variable you wish to define. Move the cursor to the three dots in the shaded corner of the cell, and click once. This will open a dialog box which allows you to label each of the values, such as that in Figure 1.4f. In the example here, for the variable fac (*Faculty*), 1 = 'Social Science', 2 = 'Arts', 3 = 'Sciences', 4 = 'Engineering' and 5 = 'Other'. In the dialog box for this variable, type in the first number code (which is 1) in the **Value** box, then click on the **Value label** box and type in the label for the code (which is **Social Science**). Click on the **Add** button, and the value and its label are entered into the workbox. Repeat this for each value, and then click on **OK** to save the labels.

The longer labels attached to the short variable name or to the number codes of variables have no effect whatever on any analysis that SPSS will carry out. What the labels do is make the output resulting from an analysis *easier to interpret*. Once the variables have been labelled, SPSS automatically will print the longer labels next to variable names and values wherever these appear on the outprint, making the output much easier for you to read and understand.