TWO

Setting up an SPSS data file

Learning objectives

At the end of this chapter you will be able to:

- enter raw data into an SPSS data file
- define the variables that make up an SPSS data file
- use SPSS to extract variable definitions
- save SPSS files
- use SPSS for data cleaning

This chapter will introduce the most widely used statistical package for analysing data. This is IBM SPSS Statistics (hereafter SPSS).

Obtaining a copy of SPSS

To conduct the procedures detailed in this book for yourself, using the data files available from www.uk.sagepub.com/argyrous3, you need a copy of SPSS. At the time of printing the latest version of SPSS was Version 19. For a brief period, versions 17 and 18 were known as PASW Statistics, and this name may occasionally appear in some of the images below. However, it is essentially the same program. SPSS is sold as a Base system for a licence fee plus annual renewal charge. This Base system can be extended through the purchase of add-on modules for an additional charge. This text will cover the functions that are available as part of the Base so that it is relevant to all users of this program, regardless of the configuration. Those who do have add-on modules should explore these, however, to see if they provide alternative and better options for obtaining the statistical results we describe in the rest of the book.

There are several options for obtaining a copy of SPSS:

1. *Purchase a commercial version of SPSS*. This can be done through a software retailer or on-line (www.spss.com/software/statistics/). The initial fee is substantial, although the annual upgrade is much cheaper. A demonstration copy can be downloaded for free from the SPSS website, but this has a limited period of use.
2. *Access a site licence copy*. If you belong to a large organization such as a university or public sector department, your organization may have negotiated a site licence with SPSS for installation and use of the program. You should check with the relevant people who manage software licences to see if you can obtain a copy of the program through such an arrangement and what the licensing conditions include.
3. *Purchase an SPSS GradPack*. If you are a university or college student you may be able to purchase a GradPack from your campus bookstore, which includes a manual and copy of the software at a much lower price than the commercial version. As with any software you purchase, however, you should check the licensing details before purchase.
4. *Purchase a Student Version*. A Student Version of the program is available at a relatively inexpensive price. This does not have the full functionality of the commercial version or GradPack; it is limited to 1500 cases and 50 variables. However, it is suitable for most of the needs of an introductory user for non-commercial purposes. Prentice Hall distributes SPSS Student Version through university and college bookstores around the world. Simply present your valid student identification. If your campus bookstore does not carry SPSS Student Version, order the software on-line at www.prenhall.com, or ask the bookstore manager to contact the local Prentice Hall distribution office.

Alternatives to SPSS

This text details SPSS procedures for statistical analysis because it is the most widely used statistics package (other than common spreadsheet programs such as Excel, which can be used for complex statistical analysis but are really designed for other purposes). This text does not use SPSS because it is the best; readers will note my frustration with this program and its peculiarities as they read the following chapters. It is only appropriate therefore to draw your attention to alternatives that are available and which you may choose rather than SPSS. To assist with this, the website for this book contains all the data files for the following examples in tab-delimited ASCII format so that they can be imported into a wide range of alternative software. There are three broad classes of alternatives to SPSS:

1. *Other comprehensive commercial programs*. There are many commercial alternatives to SPSS such as GB-Stat, InStat, JMP, Minitab, SAS, and Stata. A full list of such packages is available at www.statistics.com/resources/software/commercial/fulllist.php3.

2. *Free comprehensive programs*. An exciting development in recent years is the amount of free statistical analysis software available, usually produced under the open source licence. A listing of such software is available at statpages.org/javasta2.html#Freebies. Of these, four are particularly worth mentioning. Epi Info has been developed by the US Center for Disease Control (www.cdc.gov/epiinfo) mainly for the use of epidemiologists and other health scientists, although researchers from other disciplines will also find this program suitable to their needs. An open source version of Epi Info, called OpenEpi, which runs on all platforms and in a web browser, is available from www.openepi.com. Second, the program StatCrunch (www.statcrunch.com) allows users to load, analyse, and save results using nothing more than a standard web browser, but with much of the functionality of SPSS (and sometimes more). Third, an extremely powerful and comprehensive open source program called R is available for all platforms for free (www.r-project.org). At present it requires some knowledge of the R programming language, but graphical user interfaces are being developed so that users can select commands from a drop-down menu, a phase of development similar to that which SPSS experienced in the early 1990s. Last, a free replacement for SPSS, called PSPP, is now available. It is an open source development available at www.gnu.org/software/pspp/, and is designed to look and operate identically to SPSS. Installation is not very straightforward, but once installed it provides virtually all of the SPSS functionality at no monetary cost.
3. *Calculation pages for specific statistical pages*. These are web pages that provide tools for conducting specific analyses, including many that we will cover in later chapters. We will refer to some of these below, but a general listing of these pages is available at statpages.org.

Options for data entry in SPSS

When setting up an SPSS file to undertake statistical analysis we encounter in a very practical way many of the conceptual issues introduced in Chapter 1. Assume that in order to answer the research questions I posed at the start of Chapter 1 I survey 200 of my statistics students. In this hypothetical survey I am interested in three separate variables: age, sex, and health level. Sex is measured by classifying cases into male or female (nominal). The survey respondents also rate their respective health level as 'Very healthy', 'Healthy', or 'Unhealthy' (ordinal, but with a 'Don't know' option). Finally, I ask students their respective age in whole years on their last birthday (interval/ratio). This chapter will detail how we can record this information directly in SPSS. We will then use this data set as an example when we learn the techniques for statistical analysis in later chapters.

In this chapter we will learn to enter data directly into a 'blank' file. However, there are other means by which data can be imported into SPSS:

1. *Importing data from database, spreadsheet, or statistics programs*. SPSS recognizes files created by other popular data programs such as Excel, Systat, Lotus, dBase, and SYLK. The

programs and file extensions that SPSS recognizes are listed under the **File/Open/Data** command; click on the **Files of type:** (Windows) or **Enable** (Macintosh) option when the **Open File** box appears to see the full list. This is a convenient option for data entry projects that involve a large team of people who are not familiar with SPSS. Data can be entered into these other programs, which are widely available and well known, and then imported into SPSS. Thus only one copy of SPSS needs to be purchased and located on the computer where the analysis will be conducted, with data entry performed on these other programs and then imported into the copy of SPSS. The limitation is that data have to be entered into these other programs in a specific way so that problems and errors are not encountered when importing into SPSS. These problems usually involve the labels that normally appear as headings in the first row of the data table (such as unusual characters or the labels being split across two or more rows), or the use of unusual formats in the data block (such dates and text). These problems can sometimes be avoided by simply copying the block of data from the other program and then pasting the data into the SPSS **Data Editor** and defining the variables within SPSS using the procedures we detail below, rather than using the **Import** command.

2. *Importing text files* (*.txt). If you are not sure whether SPSS will read the 'native' version of the data file you create in another program, you may be able to save the file as a tab-delimited text (ASCII) file, which is given the extension .txt at the end of the filename. SPSS will import such a file through the **File/Read Text Data** command. An Import Wizard will then appear to assist you to bring the data across.
3. *Importing from data entry programs*. There are many programs available that require no special knowledge of SPSS (or any other data program) to facilitate data entry. These programs have many advantages: for example, they often restrict the numbers that can be entered to only those that are considered valid, thus avoiding errors. They also save the data set directly in SPSS format, or else in ASCII format that can be imported into SPSS. SPSS has its own product called SPSS Data Entry, but other commercial services exist such as Quest (www.dipolar.com). Similarly, some of the free programs listed above, such as Epi Info, also provide data entry facilities, as do on-line survey programs such as LimeSurvey (www.limesurvey.org).

The SPSS Data Editor

When you launch SPSS a window first appears asking **What would you like to do?** Select **Type in data** and then **OK**, which is the option for directly entering new data, rather than opening a file that already has data. You will then see the **SPSS Data Editor** window (Figure 2.1); make sure that the **Data View** tab at the bottom left of the window is selected.

Note the **Data Editor** menu at the top of the window. We define and analyse data by selecting commands from this menu. Usually, selecting commands from the menu will bring up on the screen a small rectangular area called a **dialog box**, from which more specialized options are available, depending on the procedure we want to undertake. By the end of this and later chapters this

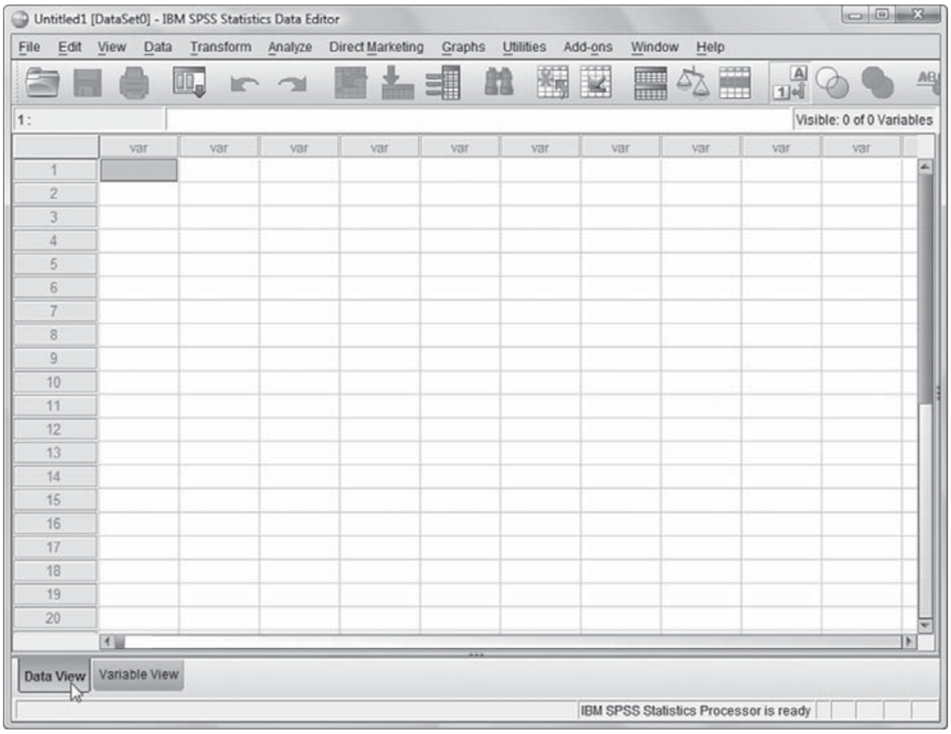


Figure 2.1 The SPSS **Data Editor** window open on **Data View**

way of hunting through the **Data Editor** menu for the appropriate commands will be very familiar. In fact, it is very similar to many other software applications that readers have encountered, such as word processing and spreadsheet software.

In Figure 2.1, below the **Data Editor** menu bar is a **button bar** that provides an *alternative* means for activating many of the commands contained within the menu. Generally we will concentrate on using the **Data Editor** menu to activate SPSS commands, even though sometimes clicking on the relevant button on the b**utton bar** may be quicker. We will concentrate just on the use of menu options simply to ensure that we learn one method consistently; after some level of proficiency readers can then decide whether selecting commands through the menu or by clicking on the buttons is preferable.

You should also observe that the cell at the top left of the page in Figure 2.1 is shaded, indicating it is the **active cell**. The active cell is the cell in which any information will be entered if I start typing and then hit the enter key on the keyboard. Any cell can be made active simply by pointing the cursor to it and clicking the mouse. You will then notice a heavy border around the cell on which you have just clicked, indicating that it is the active cell.

The **Data Editor** window consists of two pages, indicated by the page tabs at the bottom of the window. The first is the **Data View** page on which we enter the data for each variable. The **Data View** is the 'data page' on which all the information will be entered. Think of it as a blank table without any information typed into it. The **Data View** page is made up of a series of columns and rows, which form little rectangles called **cells**. Each column will contain the information for each one of the variables, and each row will contain the information for each case. The first row of cells at the top of the columns is shaded and contains a faint **var**. This row of shaded cells will contain the names of the variables whose information is stored in each column. Similarly, the first column is shaded and contains faint **row numbers** 1, 2, 3, etc.

The second page is the **Variable View** page on which we define the variables to be analysed. A *column* in the **Data View** page stores data for a single variable, whereas each *row* in the **Variable View** contains the definition for a single variable. The sequence of variables in the columns of the **Data View** page corresponds with the sequence of rows in the **Variable View** page: the variable that is defined in the fourth row of the **Variable View** will be the fourth column in the **Data View**.

We can switch from the **Data View** page to the **Variable View** page in one of two ways:

(i) click on the **Variable View** tab at the bottom of the window, or

(ii) point the cursor at the shaded cell at the top of the relevant column (**Data View**) or left-hand edge of the relevant row (**Variable View**) and double-click the mouse button (Figure 2.2).

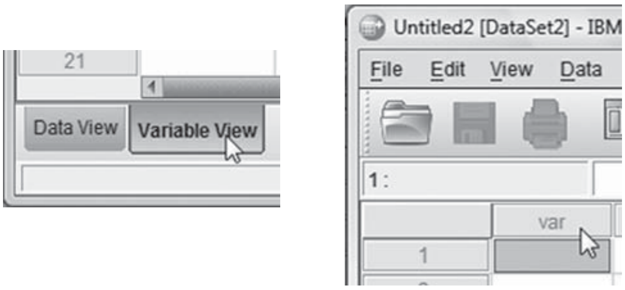
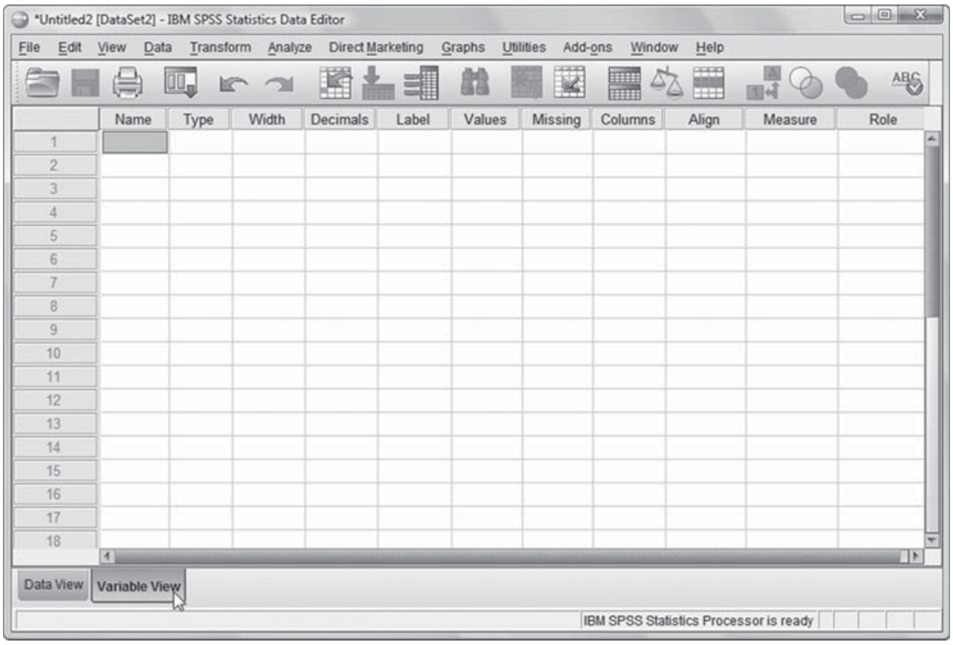


Figure 2.2 Switch from **Data View** to **Variable View** by tabbing or double-clicking column head

Try both methods to see that the result will be the same: the **Variable View** page moves to the front of the **Data Editor** window (Figure 2.3).

Each row on this page defines a single variable. There are various attributes to a variable's definition that are contained in the columns of the **Variable View**, and we define each of these attributes for a variable by moving across these columns (Table 2.1).



IBM SPSS Statistics Processor is ready

Figure 2.3 The **Variable View** page

Table 2.1 Summary of variable definition attributes

Attribute	Summary
Name	A short summary label that will appear on the column heading for the variable's data in the **Data View**
Type	Indicates to SPSS the type of information that will be entered for that variable
Width	Determines the maximum number of characters that can be entered as a datum for that variable
Decimals	Determines the number of decimal places that appear in the **Data View** when numeric data are entered for the variable. Default setting is two decimal places
Label	A variable label can have up to 128 characters and has more formatting options than the variable name. Spaces are permitted. Replaces the variable name in all output such as graphs and tables
Values	Allows us to assign numerals to replace the word labels that make up categorical scales
Missing	Indicates values for a variable in the data set that are to be ignored in data analysis
Columns	Determines the width of the columns in the **Data View**
Align	Determines whether the data entered into the **Data View** column align to the left, centre, or right of the column
Measure	Indicates the level of measurement for the scale measuring the variable
Role	Indicates the role each variable is anticipated to play in statistical models

Assigning a variable name

The first task is to give the variable a name. If we make the cell below **Name** in the first row active by clicking on it, we can type in a variable name, which in this instance is **sex**.

There are some limitations imposed by SPSS on the names we can assign to our variables:

- A variable name can have a maximum of 64 characters made up of letters and/or numbers.
- A variable name must begin with a letter.
- A variable name cannot end with a period.
- A variable name cannot contain blanks or special characters such as &, ?, !, ‘, *, or commas.
- A variable name must be unique. No other variable in a data file can have the same name.

Given these specific limitations, there are two schemes for naming variables in SPSS. One scheme uses **sequential names** indicating where on the research instrument (the questionnaire, interview schedule, record sheet, etc.) the variable appears. An example of this might be to name variables q1, q2, q3a, q3b, and so on, to indicate which question number on a questionnaire generated the data for a given variable. This provides a quick and easy way of assigning variable names and allows you to link a name directly to the research instrument on which the data are recorded. Its disadvantage is that the individual variable names do not give an impression of the contents of the variable.

The other variable naming scheme that is commonly adopted, and which we are using here, is **descriptive names**. This is a more time-consuming method, but the individual variable name, such as **sex**, gives a direct impression as to what the data in a given column are about.

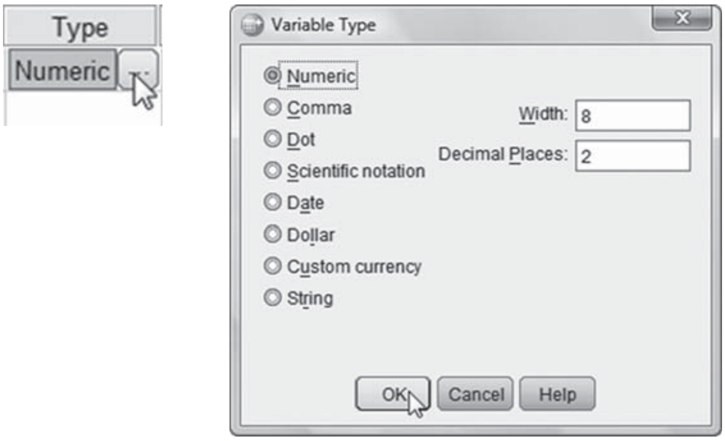
It is also possible to use a combination of these two naming schemes. For example, we might use sequential names for the bulk of responses to a questionnaire, but also use descriptive names for key demographic variables such as sex and age.

Whatever we choose to name a variable, this name will appear at the top of the column for that variable in the **Data View**.

Setting the data type

You should notice that as soon as you enter the variable name and strike the return key, information is also automatically entered in the subsequent cells in the first row. These are termed **default** settings – things about the variable's definition that are pre-set unless we choose to change them.

For example, the word **Numeric** appears in the second column headed **Type**. This is the most common form of data type, whereby numbers will be entered to indicate the category that each case falls into. In this instance we plan to enter 1 for female and 2 for male. Since most data are of a numeric type, SPSS sets this as the default so we don't need to change it. If we did want to change the data type we click on the small shaded square next to **Numeric**, which appears when this



● Numeric
○ Comma
○ Dot
○ Scientific notation
○ Date
○ Dollar
○ Custom currency
○ String

Width: 8
Decimal Places: 2

Figure 2.4 Setting the data **Type**

cell is active. This brings up the **Type** dialog box in which we can select other data types (Figure 2.4).

There are a number of other choices available for data type listed below **Numeric**. The following is a brief description of some of these items (a useful feature of SPSS is the contextual help available; if you right-click the mouse button on an item in any dialog box for which you require more information, such as the list of data types in the **Variable Type** box, a contextual help option appears which if selected will give details about that item):

- *Comma*. This defines a numeric variable whose values are displayed with a comma for every three places and with a period as the decimal delimiter. Choosing **Comma** rather than **Numeric** can be useful if you wish the values generated in tables, graphs, and statistics such as the mean, to appear formatted with a comma.
- *Scientific notation*. A numeric variable whose values are displayed with an embedded E and a signed power-of-ten exponent. The **Data Editor** accepts numeric values for such variables with or without an exponent. The exponent can be preceded either by E or D with an optional sign, or by the sign alone.
- *Date*. A numeric variable whose values are displayed in one of several calendar-date or clock-time formats. You can enter dates with slashes, hyphens, periods, commas, or blank spaces as delimiters. This data type can be useful, for example, where a person's birth date needs to be recorded, or the date on which a survey was completed needs to be included with the data set.
- *Custom currency*. A numeric variable whose values are displayed in one of the custom currency formats that are defined in the **Currency** tab of the **Options** dialog box. Defined custom currency characters cannot be used in data entry but are displayed in this format on the **Data View** page.

- *String* (also known as *alphanumeric* variable). Values of a string variable are not numeric, and hence not used in calculations. They can contain any characters up to the defined length. Upper and lower case letters are considered distinct. An example is 'm' and 'f' for male and female, respectively. This data type is often used for typing responses to open-ended questions that may be different for each case, and therefore cannot be precoded.

Setting the data width and decimal places

The **Width** of the data is the maximum number of characters that can be entered as a datum for each case. The default setting is 8, so if we had values for a variable with more than eight digits we would need to change this. For example, if we were entering the populations of various countries, we would not be able to include data for countries such as the USA or China, which have populations greater than 99,999,999. We would need to change the default data **Width** from 8 to a higher number such as 10. Similarly, if we chose **String** as the data type so that we can enter long strings of text (such as open-ended responses in a questionnaire) we would need to increase the **Width**.

The number of **Decimal Places** is a 'cosmetic' function, in that it alters the way data are displayed once they are entered but does not affect what we can do. If we do not change the default setting of 2 decimal places, 1 will show up as 1.00 on the **Data View** page.

There are two ways in which the data **Width** and **Decimal Places** settings can be changed from the default settings:

1. In the **Variable Type** dialog box (Figure 2.4) that we brought up to enter the data **Type** we also have the option to change the variable width and the number of decimal places.
2. Another way to change these aspects of the variable definition is in the columns headed **Width** and **Decimals** on the **Variable View** page. Clicking on either of these cells produces up and down arrows on the right edge of the cell, which can be used to change values (Figure 2.5). Alternatively, you can highlight the number 8 and type in the desired value.

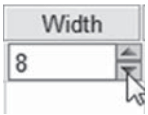


Figure 2.5 Setting the data **Width**

Defining variable labels

The next column is headed **Label**. A v**ariable label** is a longer description of the variable (up to 120 characters) than can be included in the **Name** column. It also permits formatting that is more suitable for presentation purposes, such as the use of capital letters and spaces between words. Although the short variable name **sex** is fairly self-explanatory, to get into the habit of providing variable labels we will type **Sex of student** in the **Label** column. If we do not provide a label, any tables we generate for this variable, for example, will be headed by 'sex'; by providing a longer and better-formatted label, the table will instead be headed by 'Sex of student', which is a much better way of presenting results. Getting the labelling formatted correctly at the start of data analysis can avoid having to re-edit numerous tables and graphs later.

There are a couple of tips for providing **variable labels**:

- With interval/ratio data it is very useful to include the unit of measurement in the label. Thus, even though the variable 'age' does not seem to require a **variable label** to explain its meaning, it is useful to type 'Age in years' as the label for that variable.
- It is often helpful to use the exact wording of a questionnaire/interview question as a **variable label** so that it will be presented in any output.

Defining value labels

The **Value Labels** function allows us to specify our coding scheme – the way in which responses will be transformed into 'shorthand' codes (numbers) that allow us to perform statistical analysis, and especially to make data entry quicker. In SPSS the numerical codes are called **values** and the actual responses are called **value labels**. Thus sex has two value labels, female and male, and we link each label to a specific code number or **value**:

1 = female
2 = male

Instead of typing in male or female as our data, we type in the *codes* assigned to these labels, and this is a much faster procedure.

With a nominal scale such as 'sex' the actual numerical code given to each value label is arbitrary: we can just as easily reverse the order and assign 1 to male and 2 to female. In fact, we could assign 3 to female and 7 to male, or any other combination of values. But, generally, the simpler the coding scheme, the better. The procedure for defining the value labels for sex is provided in Table 2.2 and Figure 2.6.

If you receive an error message when you click on **Continue** stating '**Any pending Add or Change operations will be lost**', it is because you did not click the **Add** button after typing in a value and value label. If this happens click on **OK**, which will return you to the **Value Labels** dialog box, and then click on **Add**.

Table 2.2 Assigning value labels in SPSS

SPSS command/action	Comments
1 In the column headed **Values** click on the small shaded square next to **None**	This brings up the **Value Labels** dialog box so that we can assign labels
2 In the box next to **Value:** type **1**	
3 In the box next to **Label:** type **female**	You will notice that as soon as you start typing **Add** suddenly darkens, whereas it was previously faint
4 Click on **Add**	This pastes the information into the adjacent area so that **1**=**female**. The cursor will automatically jump to the box next to **Value:**
5 Type **2**	
6 In the box next to **Label:** type **male**	
7 Click on **Add**	**2**=**male** will be added to the list
8 Type **3**	
6 In the box next to **Value Label:** type **Did not answer**	
7 Click on **Add**	**3**=**Did not answer** will be added to the list
8 Click on **OK**	

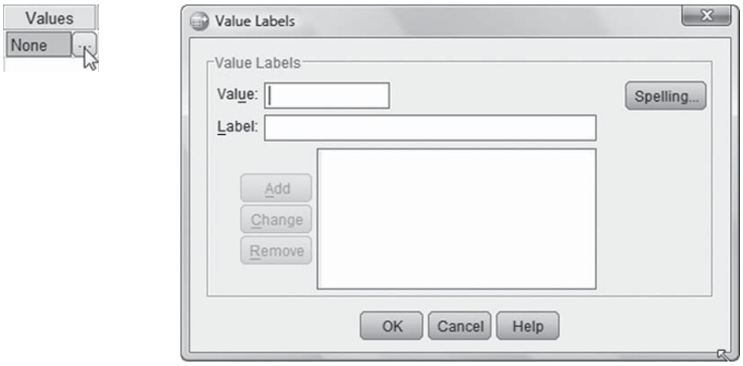


Figure 2.6 Assigning value labels in SPSS

This variable is an easy one for providing value labels, but for other instances there are some simple rules to follow in determining value labels that can prevent problems during analysis:

1. *Give separate codes for all possible types of 'non-valid' responses*. In my hypothetical survey of students, I asked students to rate their own health. The scale had three options, plus a 'Don't know' option at the end of the scale for students who did not feel competent to answer. For this variable I would like to separate students who answered 'Don't know' and exclude them from analysis. In other words, I am treating 'Don't know' as a non-valid response. Some students, however, may simply not answer this question, and such cases also represent non-valid responses. In other words, there are at least two reasons why a student may not give 'Unhealthy', 'Healthy', or 'Very healthy' as the response: either they responded 'Don't know' or they simply did not answer. In order to

allow us to later distinguish between these types of non-valid responses, they are each assigned a unique code as follows:

1 = Unhealthy
2 = Healthy
3 = Very healthy
4 = Don't know
5 = Did not answer

2. Give 'null' values a code of zero. We often encounter survey questions that have a No/Yes response set. In such instances we would code '0 = No'. The value of 0 is assigned to No for a deliberate reason: it is a type of response that indicates a null response, a case where no quantity of the variable is present. Other examples of null responses are responses of 'Never' and 'Not at all'. Giving these responses a value of zero is very useful in later analysis, such as scale construction where values for a set of variables are added together.
3. *For ordinal variables, make sure that the numeric scale reflects the measurement scale*. Variables measured at the ordinal level have categories that indicate the relative strength in which the variable is present. Our scale for measuring the health of students is an example. We can say that a student who answers 'Very healthy' *possesses more* of the variable 'health' than does a person who responds 'Unhealthy'. This quantitative increase should be reflected in the numerical codes assigned to the categories: 'Very healthy' should get the biggest number, and 'Unhealthy' the smallest. Our coding scheme might therefore be:

   1 = Unhealthy
   2 = Healthy
   3 = Very healthy

   This is particularly helpful in later data analysis, where correlations might be calculated with this variable.
4. Do not give interval/ratio scales complete value labels. With interval/ratio scales we do not need to code the responses since the values 'speak for themselves', especially if we have included the units of analysis in the **variable label**. Thus we don't need to indicate that case number '1 = 1', case number '2 = 2', and so on. Nor do we have to state for age that '20 = 20 years of age', '21 = 21 years of age', and so on. The only situation in which we might use the **Value Labels** function for interval/ratio data is where there are specific categories of missing data that we want to identify (as is the case with the age variable in our example of student survey responses).

Setting missing values

The next option in defining a variable is the **Missing** values option. A missing value is a number that indicates to SPSS that the response is not valid (as we discussed in the previous section) and should not be included in the analysis. Missing values can arise for many reasons. For any survey question there is always the possibility that someone simply did not answer the question, or else wrote with illegible handwriting. In another instance, a survey question, such as the health rating question in our student survey, may allow for 'Don't know' at the end of the scale and as such should not be used in analysis. A third type of missing data occurs where we have skip or filter questions that result in some questions being not applicable to all respondents. For whatever reason, when we do not have a useful datum for an individual case for a specific variable we need to enter a missing value into the relevant cell, indicating that a valid response was not provided in that instance and therefore should not be included in any analysis of that variable.

SPSS has a default setting called the **system missing value** that appears as a period in a cell where no valid response is entered. We can also provide **user-defined missing values**, whereby we specify a particular number to indicate invalid responses. We need to be careful to select a number for the missing value that the variable cannot possibly take. In this example, we can choose 3 to be the missing value, since it is impossible for the variable 'sex' to take on this value. Obviously if we were measuring the age of pre-school children, 3 would not be an appropriate choice for the missing value; 99 might be better because such a score could not actually represent a real case.

There are a number of categories of missing values that are commonly encountered in research. We can enter unique non-valid values into the data set for each of these where they arise and code each value as missing:

- *No response*. People do not always complete every question in a survey, or provide a measurement for every variable in a study.
- *Not applicable*. There are often questions in a survey, for example, that only apply to a subset of the sample. For example, a question about religious affiliation may be preceded by a question on whether a respondent is religious. Only those who answer Yes to the first question are then asked the question about religious affiliation. Those who stated they have no affiliation give no answer to the second question, but this is different than a non-response from someone who was meant to answer that question. We therefore want to keep separate those respondents for whom the second question about religious affiliation does not apply and those for whom the question does apply but did not answer this second question.
- *Don't know*. Measurement scales, such as those that ask for opinion or satisfaction, often include a 'Don't know' option that is 'off' the scale, and therefore needs to be coded but treated as missing.

SPSS allows us to enter either **discrete** values or a **range** of values as missing:

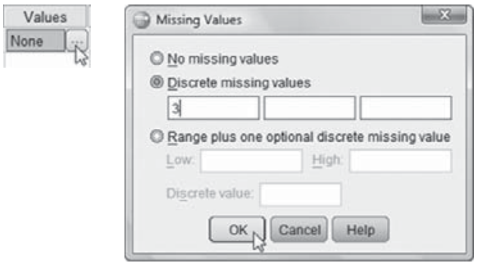
- Discrete missing values are specific numbers to be excluded. It is common practice to use 999, for example, as a discrete missing value, as this does not represent a valid response for most of the variables we could analyse (although it is not an appropriate missing value for variables such as weekly income, where $999 is a potentially valid weekly income).

- A range takes in a contiguous set of numbers that are to be treated as missing. These can be numbers above the highest possible valid response on the scale, or numbers below the lowest possible valid response. Thus if you know that no one in your survey can be above the age of 18, because you have specified this age as a sampling criterion, you can assign 19 through to 9999 as a range of missing values. Similarly, it allows the use of negative numbers to be employed as missing values in data entry, with –9999 to –1 specified as the missing value range. This can be useful for any variables that can only take on positive values. SPSS also allows us to include one discrete missing value as missing along with the range of values.

The procedures for assigning user-defined missing values to the 'Sex of student' variable in my survey are presented in Table 2.3 and Figure 2.7. In this instance we specify 3 as the missing values, which then requires us to enter 3 during the data entry process if we encounter a respondent for whom their sex cannot be determined. Note that we could have used any number other than 1 or 2 as missing (the codes for female and male, respectively), but the number specified as the missing value needs to be the same number entered in the **Data View** for that type of invalid response.

Table 2.3 Assigning user-defined missing values in SPSS

SPSS command/action	Comments
1 In the column headed **Missing** click on the small shaded square next to **None**	This brings up the **Missing Values** box. The radio button next to **No missing values** is selected indicating that no user-defined missing values is the default setting
2 Click on the radio button next to **Discrete missing values**	The radio button to the left of **Discrete missing values** is selected, and the cursor will be flashing in the adjacent rectangle
3 Type **3**	
4 Click on **OK**	



Values
None

Missing Values

○ No missing values
● Discrete missing values
3

○ Range plus one optional discrete missing value
Low: High:
Discrete value:

Figure 2.7 Assigning missing values in SPSS

Setting the column format and alignment

The next two columns on the **Variable View** page are headed **Column** and **Align**. These options are basically cosmetic, in so far as they only affect the appearance of the **Data View** without affecting the type of analysis we can do. These columns affect two particular settings. The first is the column width as it appears in the **Data Editor**, and the other is the alignment of the data contained in each column. The default settings are a column width of 8, and right alignment, which are both adequate for our purposes, so we will not change these settings.

Specifying the level of measurement

The last step in defining a variable for data entry is to specify the level of measurement for the variable. The default setting is **Scale**, which is SPSS's unfortunate name for interval/ratio data. In this example, sex is a nominal variable, so to change the setting to nominal we click on the down arrow next to **Scale** and select **Nominal** (Figure 2.8).

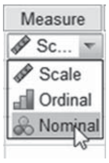


Figure 2.8 Assigning the level of measurement

Specifying the role of each variable

With SPSS 18 a new attribute of variable definition was added. This is the specification of the role that each variable might play in subsequent data analysis. In Chapter 1 we discussed the idea of statistical models. SPSS allows us to build these conceptualizations into the data set, and some commands support pre-defined roles that can be used to preselect variables for analysis. When you run one of these commands, variables that meet the role requirements will be automatically displayed in the destination list(s). Available roles are:

- *Input*. The variable will be used as a predictor or independent variable.
- *Target*. The variable will be used as an output or dependent variable.
- *Both*. The variable will be used as both input and output.
- *None*. The variable has no role assignment.
- *Partition*. The variable will be used to partition the data into separate samples for training, testing, and validation.
- *Split*. Included for round-trip compatibility with the SPSS add-on SPSS Modeler. Variables with this role are not used as split-file variables in PASW Statistics.

By default, all variables are assigned the Input role. With our data we can expect that age and sex, being 'pre-set' demographic variables, will be independent variables in our analysis. We therefore leave them with the default Input role. Health rating, however, is likely to be a variable we want to explain. We therefore change its role to Target. The id variable has no role to play in analysis, as it is there for administrative purposes. We can change its assignment from Input to None.

Controlling the appearance of the Variable View

We can now go through the same variable definition procedure for our remaining two variables. I leave it to you to go through the steps we have just followed, but adapting them to record the relevant information for Health rating and Age. To help you along you should follow the coding scheme in Table 2.4. In fact, before undertaking data entry it is very helpful to write out a coding scheme such as this to clarify the definitions that you will follow. Constructing a coding scheme is especially helpful if more than one person is involved in the data entry process, so that everyone follows the same scheme.

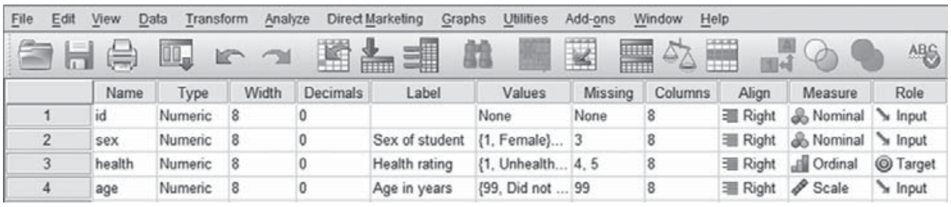
Table 2.4 Coding scheme

Variable name	Variable label (optional)	Values = Value labels	Missing values	Level of measurement	Role
sex	Sex of respondent	1 = Female 2 = Male 3 = Did not answer	3	Nominal	Input
health	Health rating	1 = Unhealthy 2 = Healthy 3 = Very healthy 4 = Don't know 5 = Did not answer	4, 5	Ordinal	Target
age	Age in years	99 = Did not answer	99	Interval/ratio	Input

Another consideration that we often make when setting up a data file is to first insert a column for the ID number of each case. We do not need to provide any definitions for this column, since it is simply used to keep track of where the data came from. It ensures that if the data file is sorted so that the cases appear in a different order, or new cases are inserted, each case retains its unique identification number.

If you follow these procedures correctly the **Variable View** page will look like Figure 2.9. Notice that the order of the variables across the columns in the **Data View** page is the same as the order of the variables down the rows in the **Variable View** page. We can easily change this order so that we have particular groups of variables sitting side-by-side in the **Variable View** page.

For example, rather than have age appear as the last variable, we might want to have age appear in the second column between id and sex. To move this variable, on the **Variable View** page, we follow these steps:


File Edit View Data Transform Analyze Direct Marketing Graphs Utilities Add-ons Window Help

	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure	Role
1	id	Numeric	8	0		None	None	8	Right	Nominal	Input
2	sex	Numeric	8	0	Sex of student	{1, Female}...	3	8	Right	Nominal	Input
3	health	Numeric	8	0	Health rating	{1, Unhealth...	4, 5	8	Right	Ordinal	Target
4	age	Numeric	8	0	Age in years	{99, Did not ...	99	8	Right	Scale	Input

Figure 2.9 **Variable View** with complete data definitions

1. Click on the shaded, numbered cell that is the left edge of the row containing the variable definition for age. The whole row should then be highlighted.
2. Click and hold on the same shaded, numbered cell.
3. Move the cursor up so that the red line that appears sits between the two variables where you want age to appear.
4. Release the mouse button.

We can also change the order or hide some of the columns in the **Variable View**. From the **Variable** menu we choose **View/Customize Variable View**, which brings up the **Customize Variable View** dialog box (Figure 2.10). In the dialog box we can:

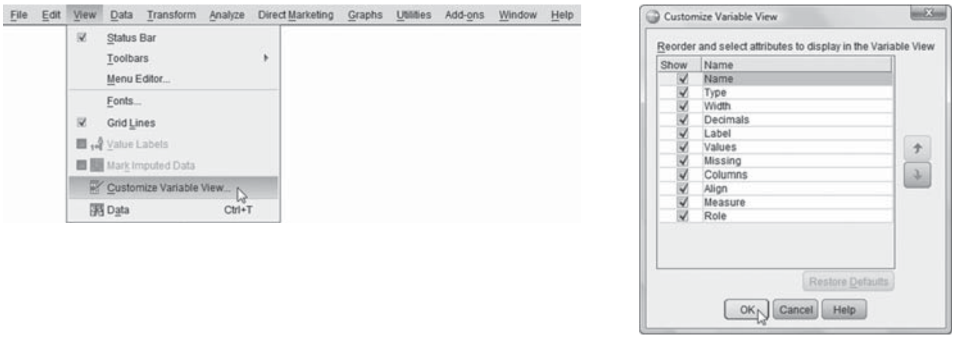


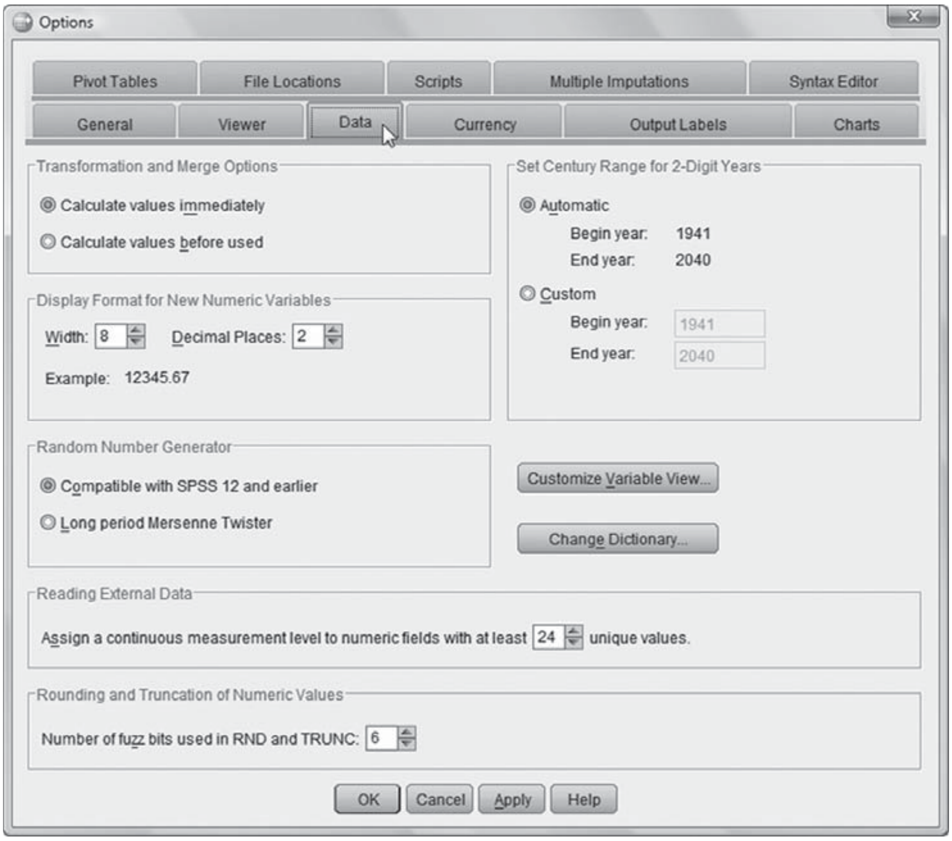
Figure 2.10 The **Customize Variable View** command and dialog box

- hide some columns that we do not want to see in the **Variable View** page, by clicking on the tick-box next to the name of the variable property we want to hide. This removes the tick mark, indicating that the attribute will be hidden.
- reorder the sequence in which the columns appearing in the **Variable View** will be displayed. We select the variable attribute in the list and use the up or down arrows to change where it appears in the sequence. For example, I prefer to have the variable **Label** column appear immediately after the variable **Name** column, and to do this I select **Label** from the list and move it up three spots.

Shortcuts for defining variables

This step-by-step process of defining each variable that will make up a data set can become very tedious, especially where we are setting up a file that will contain information for many variables. There are some other tools that SPSS provides to speed up the process and remove much of the repetition involved in variable definition:

1. *Change the default settings for the program*. Many of the default settings for entering, presenting, and analysing data in SPSS are less than ideal. For example, displaying numeric data to 2 decimal places by default is annoying as most data are entered in whole numbers. Two decimal places are usually unnecessary and make the **Data View** page cluttered. The **Decimals** setting on the **Variable View** page can be changed for each variable as we described above. An alternative is to use the **Edit/Options** (Windows) or the **SPSS Statistics/Preferences** (Macintosh) command to change the default setting, so that when data are entered they appear with 0 decimal places (i.e. whole numbers). To change the default settings for data definitions and appearance, click on the **Data** tab at the top of the **Options** window (Figure 2.11).



Transformation and Merge Options
● Calculate values immediately
○ Calculate values before used

Display Format for New Numeric Variables
Width: 8 Decimal Places: 2
Example: 12345.67

Random Number Generator
● Compatible with SPSS 12 and earlier
○ Long period Mersenne Twister

Set Century Range for 2-Digit Years
● Automatic
Begin year: 1941
End year: 2040
○ Custom
Begin year: 1941
End year: 2040

Reading External Data
Assign a continuous measurement level to numeric fields with at least 24 unique values.

Rounding and Truncation of Numeric Values
Number of fuzz bits used in RND and TRUNC: 6

Figure 2.11 The SPSS **Options/Data** tab

While we are discussing the use of **Edit/Options** or **SPSS Statistics/Preferences**, we note that there are many other default settings that can be changed under this command. These cover many aspects of the functioning of the program and are far too numerous to even list here; you should explore these at your leisure, but with one word of warning: do not alter the default setting on someone else's version of SPSS without their permission.

2. *Copy and paste variable definitions*. One of the big advantages of the **Variable View** page is the ease with which we can copy aspects of one variable's definitions to another. For example, if we had two or more variables with exactly the same value labels, we could specify these labels for each variable individually, or we can simply define one variable's value labels, click on this cell with the definitions, select **Edit/Copy** from the menu, click on the relevant cell(s) for subsequent variables into which the same value labels will be entered, and select **Edit/Paste**. You might do this here to copy 0 decimal points specified for sex down to the other two variables.
3. *Create custom variable attributes*. We can add our own attributes to the default list that appear in the **Variable View** and which we listed in Table 2.1. A new attribute can apply to one or more of the variables in the data set, and a default value specified. For example, we might want an attribute that indicates the broad class of variables to which each variable belongs, such as demographic, behavioural, attitudinal, and so on. To create a new attribute we select **Data/New Custom Attribute** from the menu, and give the attribute a name and a set of values that it can take.

Generating variable definitions in SPSS

There are three ways in which we can obtain variable definitions during the course of the analysis (without, that is, actually going back into **Variable View**). One method by which we can ask SPSS to provide the coding scheme we have used in the variable definitions is by selecting **File/Display Data File Information/Working File** (Figure 2.12). This command will generate information in the **Viewer** window (Figure 2.13)

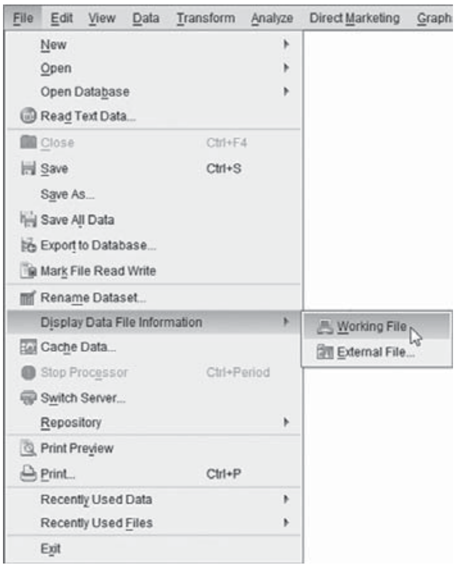
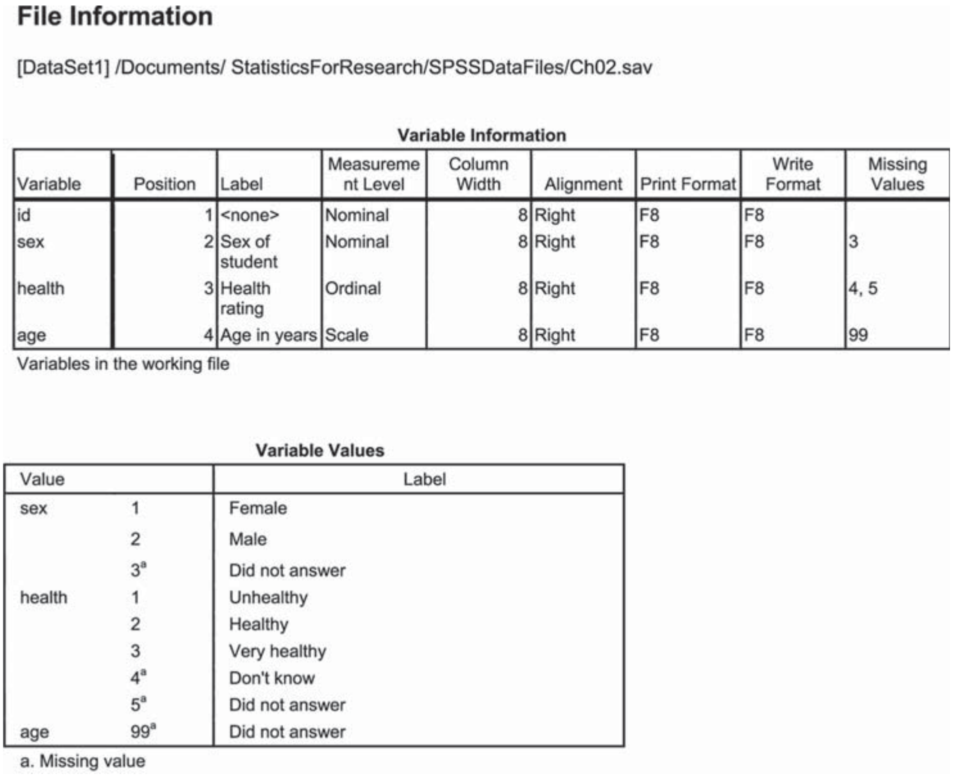


Figure 2.12 Generating file information

that can be printed for reference during the analysis. The numbers for **Position** next to each variable name indicate the column in which the variable appears. Printing out this file information and having it close at hand can be very useful when actually analysing data. It also provides a record for anyone else who may need to work with the data set and therefore will need to know the coding scheme used for data entry.

DISPLAY DICTIONARY.



File Information

[DataSet1] /Documents/ StatisticsForResearch/SPSSDataFiles/Ch02.sav

Variable Information

Variable	Position	Label	Measureme nt Level	Column Width	Alignment	Print Format	Write Format	Missing Values
id	1	<none>	Nominal	8	Right	F8	F8	
sex	2	Sex of student	Nominal	8	Right	F8	F8	3
health	3	Health rating	Ordinal	8	Right	F8	F8	4, 5
age	4	Age in years	Scale	8	Right	F8	F8	99

Variables in the working file

Variable Values

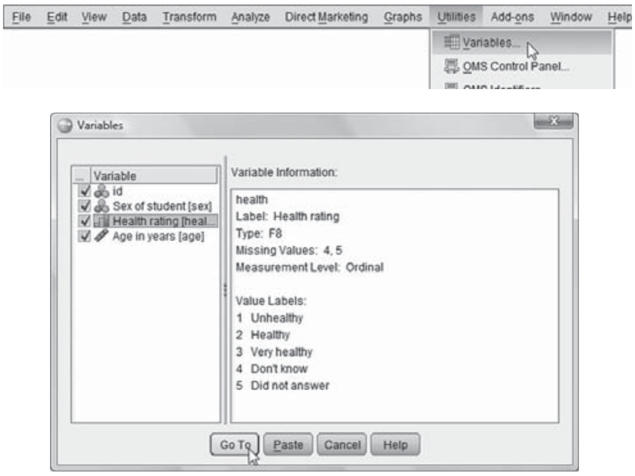
Value		Label
sex	1	Female
	2	Male
	3[a]	Did not answer
health	1	Unhealthy
	2	Healthy
	3	Very healthy
	4[a]	Don't know
	5[a]	Did not answer
age	99[a]	Did not answer

a. Missing value

Figure 2.13 SPSS **File Information** output

The **File Information** command allows you to obtain variable definitions for *all* your variables at once. You may, however, occasionally want to quickly obtain the definition for only one variable. This can be done through the **Utilities/Variables** command. This command will bring up the **Variables** dialog box, from which you can select the variable for which definitions are required (Figure 2.14).

The remaining way in which variable information can be obtained is in the various dialog boxes that appear when we are conducting particular types of analysis. For example, in Chapter 4 we will work with the **Analyze/Descriptive Statistics/ Frequencies** command. As with other dialog boxes, we are given a source list of variables from which we can choose variables to analyse. If we want to quickly find



health
Label: Health rating
Type: F8
Missing Values: 4, 5
Measurement Level: Ordinal

Value Labels:
1 Unhealthy
2 Healthy
3 Very healthy
4 Don't know
5 Did not answer

Figure 2.14 SPSS **Utilities/Variables** command and dialog box

the definitions for a variable in the list, rather than cancelling and going back to the **Data View**, SPSS provides a contextual menu that we can bring up on screen. To bring up this contextual menu right-click on the desired variable. A small menu will appear from which you select **Variable Information**. A window appears providing the basic definitions for the chosen variable (Figure 2.15).

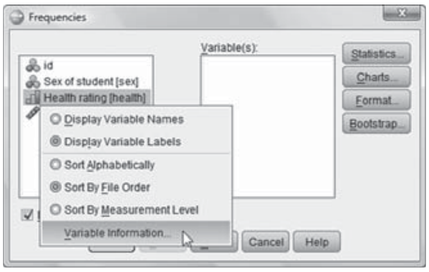
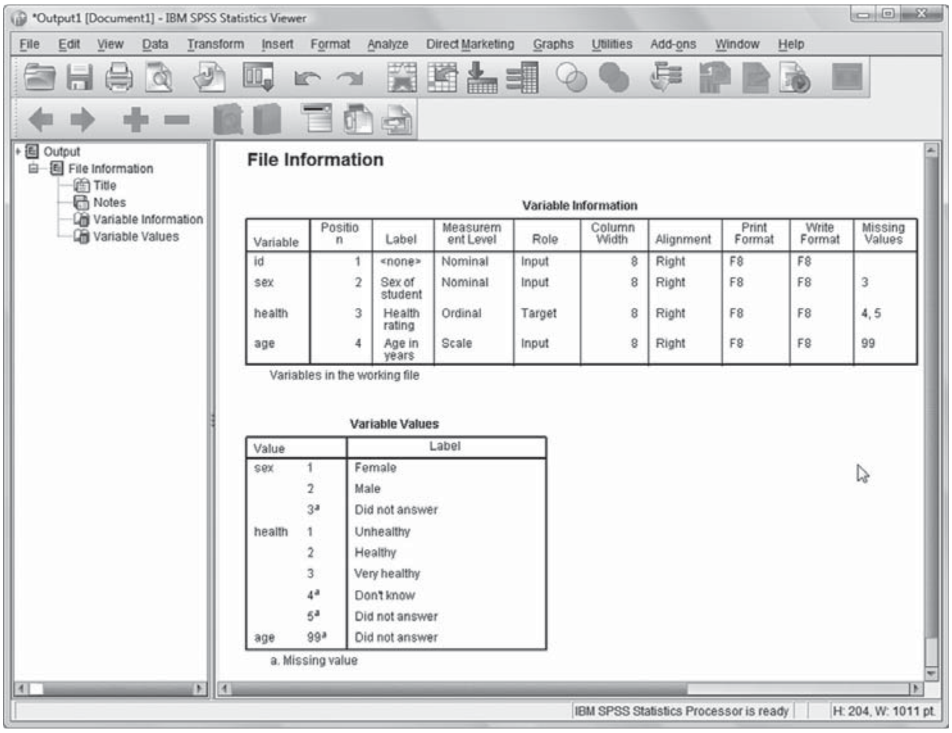


Figure 2.15 **Obtaining Variable Information** in dialog boxes

The SPSS Viewer window

You will notice that SPSS displayed the file information we requested in a separate **Viewer** window (Figure 2.16). You will become familiar with this process of requesting output from the menu, based on the data in the **Data Editor**, and viewing the output in the **Viewer** window. To switch back and forth we simply select the relevant 'page' from the **Window** option on the menu bar, or select either window from the task bar at the bottom of the screen (Windows), or right-click on the SPSS icon on the dock (Macintosh).


*Output1 [Document1] - IBM SPSS Statistics Viewer
File Edit View Data Transform Insert Format Analyze Direct Marketing Graphs Utilities Add-ons Window Help
Output
File Information
Title
Notes
Variable Information
Variable Values
IBM SPSS Statistics Processor is ready
H: 204, W: 1011 pt

File Information

Variable Information

Variable	Position	Label	Measurement Level	Role	Column Width	Alignment	Print Format	Write Format	Missing Values
id	1	<none>	Nominal	Input	8	Right	F8	F8	
sex	2	Sex of student	Nominal	Input	8	Right	F8	F8	3
health	3	Health rating	Ordinal	Target	8	Right	F8	F8	4, 5
age	4	Age in years	Scale	Input	8	Right	F8	F8	99

Variables in the working file

Variable Values

Value		Label
sex	1	Female
	2	Male
	3[a]	Did not answer
health	1	Unhealthy
	2	Healthy
	3	Very healthy
	4[a]	Don't know
	5[a]	Did not answer
age	99[a]	Did not answer

a. Missing value

Figure 2.16 The SPSS **Viewer** window

The major part of the **Viewer** window is a **Display** frame that contains the information we have requested. Two points are worth noting about the **Display**:

- SPSS does not print over existing output whenever we request new information. Instead it adds new output to the bottom of the existing output. When undertaking a lot of analysis this can create a lengthy **Display**.
- 'Old' output is not automatically updated whenever we change information on the **Data Editor** window. For example, if we change the value labels for variables we are working with, the old variable information on the **Viewer** window we have just generated will no longer apply, but will remain unchanged. We will need to again run the **File Info** command to generate new variable information for the updated variable labels.

It is often very helpful to add text to explain the output listed in the **Display**. For example, the file information we generated just states **File Information** as a title. You can double-click on this title (or any other text in the **Display**) and add/change the text, or also change the formatting. Thus you may add text to the title so that it reads **File Information for Ch.02.sav** so that when you print this output you know to which data file it applies. Alternatively, you can use the **Insert** command to insert particular types of preformatted text into the output. The options available for adding new text (rather than editing existing text) are:

- *New Heading*. This appears in the navigator window and acts like a new heading in a table of contents.
- *New Title*. This is a first-level heading that normally appears in large, bold font.
- *New Page Title*. A page-break is automatically inserted before a page title, and the typed text appears at the top centre of the page.
- *New Text*. This is small, plain text.

To make it easier to navigate through the output in the **Display** frame, on the left-hand side is a narrower **Outline** frame that provides a 'table of contents'. This is a list of the output we have generated during the course of an SPSS session. We can alter the relative size of the two frames that make up the **Viewer** by clicking on and dragging the line that separates them. The **Outline** frame provides a number of handy features that can make data analysis much easier, especially when working with a lot of results:

- *Move around the output page*. Sometimes we generate a large amount of output that spreads down several (virtual) pages of the **Display** frame. We may find ourselves constantly referring back and forth to different parts of the output. We can do this by using the vertical scroll-bar on the right edge of the window. A more direct way is to click on the desired bit of output in the **Outline** list.
- *Hide or show different parts of the output*. We can simplify the **Display** by hiding parts of the output that we don't think we will be referring to for a while, but still want to keep for possible later use or printing. To hide an item we double-click on its label in the **Outline** frame. This will also hide all other items below this one in the hierarchy of output. A hidden item is indicated by the closed-book icon next to its name in the **Outline** frame (by default the Notes that are generated with any output are hidden). To reveal an item we again double-click on it; the icon appearing next to it will now be an open book.
- *Move or delete selected parts of the output*. You can reorder the way output appears in the **Outline** by first selecting its title in the list, and then dragging and dropping it into the point at which you want it to appear. Similarly, by clicking on an item in the **Outline** and striking the delete key you can permanently remove it from the output. You can also select multiple items in **Outline** by holding down the shift key for consecutive items in the list, or by holding down the control key (Windows) or apple key (Macintosh) for non-consecutive items in the list.
- *Select items for printing*. When it comes to printing, SPSS is an extremely wasteful program. The output is not very compact, and the **Display** frame can accumulate a great deal of output during the course of analysis. To avoid waste when printing, it is possible to select a subset of output to print. By using the control key (Windows) or apple key (Macintosh) you can select individual items from the **Outline** frame before printing. The **File/Print** command will then default to printing only the selected items rather than the entire contents of the **Display** frame.
- *Select items for exporting to another format*. SPSS allows output to be exported into a variety of formats. For output that includes text, these formats are Excel (*.xls), Word (*.doc), webpage (*.htm), portable document format (*.pdf), PowerPoint (*.ppt), and plain text (*.txt). For graphics output the formats are bitmap (*.bmp), jpeg (*.jpg), encapsulated postscript (*.eps), portable network graphics (*.png), and tagged image file (*.tif). To export output into any of these formats select **File/Export** or right-click on the output and

select **Export** from the contextual menu that appears. As with printing, you can choose to export all the visible output or only preselected items.

Saving a data file

The remaining action in setting up the data file is to save the file (we need to switch from the **Viewer** window to the **Data Editor** containing the data before we save them). This is an identical process to most other computer programs you are familiar with: select the **File/Save As** from the menu and browse to a location on your computer where you want to store the file. Wherever you choose to store data, it is very important to *make a regular backup*. No storage medium is immune to errors. Get into the habit of making a copy of your data files to guard against any unforeseen problems.

Once this has been done, the name of the active data file will appear in the bar at the top of the **Data Editor** (Figure 2.17).

*Ch02.sav [DataSet3] - IBM SPSS Statistics Data Editor

Figure 2.17 SPSS filename display

After using the **File/Save As** command, a file can be quickly resaved in the same location and with the same name with the **File/Save** command instead of **File/Save As**. In fact, you should not wait until the end of your data entry session to save the file. Mishaps can happen, often at the worst time. Losing data, after spending a considerable amount of time entering them, can be very demoralizing. You should get into the habit of saving work every 15 minutes or so (and also making a backup).

Data entry

Once the variables have been defined, the data can be entered. If we switch to the **Data Editor** window and select the **Data View** page we will see that the first four columns are headed sequentially with the variable names we have just defined (Figure 2.18). The first column in the **Data View** corresponds with the first row in the **Variable View**, the second column in the **Data View** corresponds with the second row in the **Variable View**, etc.

We can now enter the data into each of these columns. To see how this is done, we will enter the data for the student with ID number 1, who is male, very healthy, and 17 years old. We click on the top left cell below **id** so that it is active and type **1**, which is this student's ID number, and then hit the tab key. The cursor will jump across to the cell below **sex** where we type **2,** which is the code value for male. We tab across to the next column and type **3** for 'Very healthy', and then tab across to enter **17** under **age**.

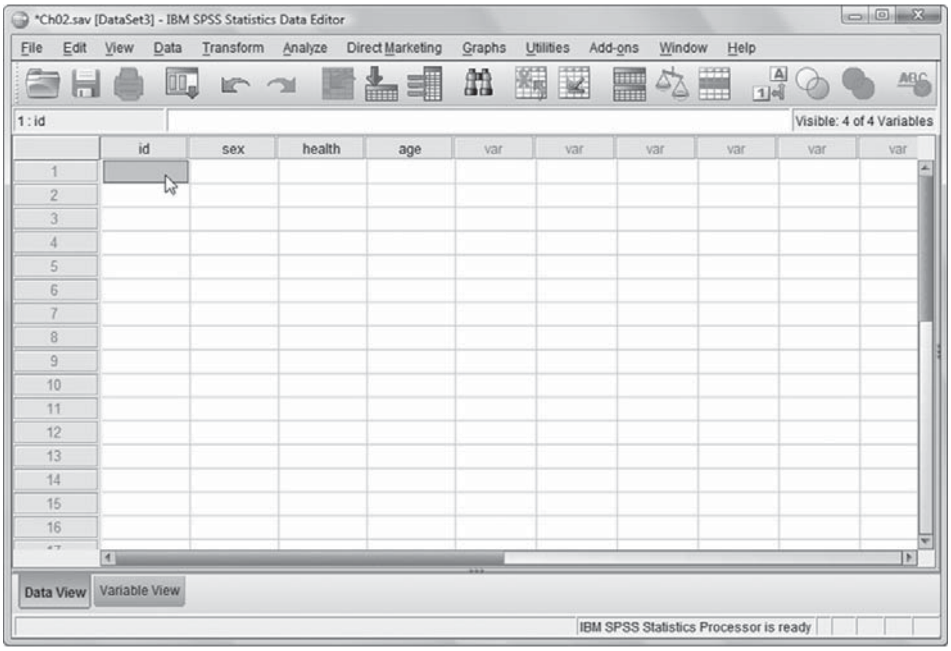


Figure 2.18 The **Data View** page with defined variables

SPSS will display either the value labels or the values on the data page, and we can switch between these options by selecting **View** from the menu and then **Value Labels** (Figure 2.19). The advantage of viewing data as labels rather than numeric codes is that when we enter data that have been assigned value labels into a cell, we are presented with a contextual menu from which we can select the appropriate response. For example, to enter the fact that case 1 is male, we can either type 2, or click on the down arrow in the cell and scroll to Male in the list.

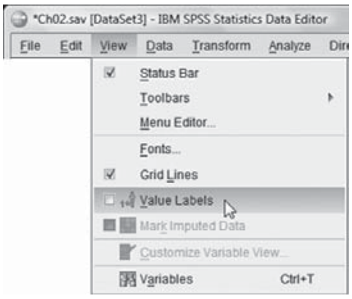
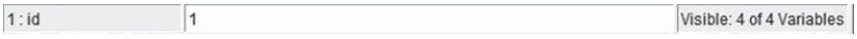


Figure 2.19 The SPSS **Value Labels** command

You may notice that, as you type, the data initially appear on the bar just above the data fields, called the **Cell Editor** (Figure 2.20). This is where information is entered until you hit the tab key, which then enters the data into the active cell. The 'address' of the active cell is indicated on the left of the **Cell Editor**. This

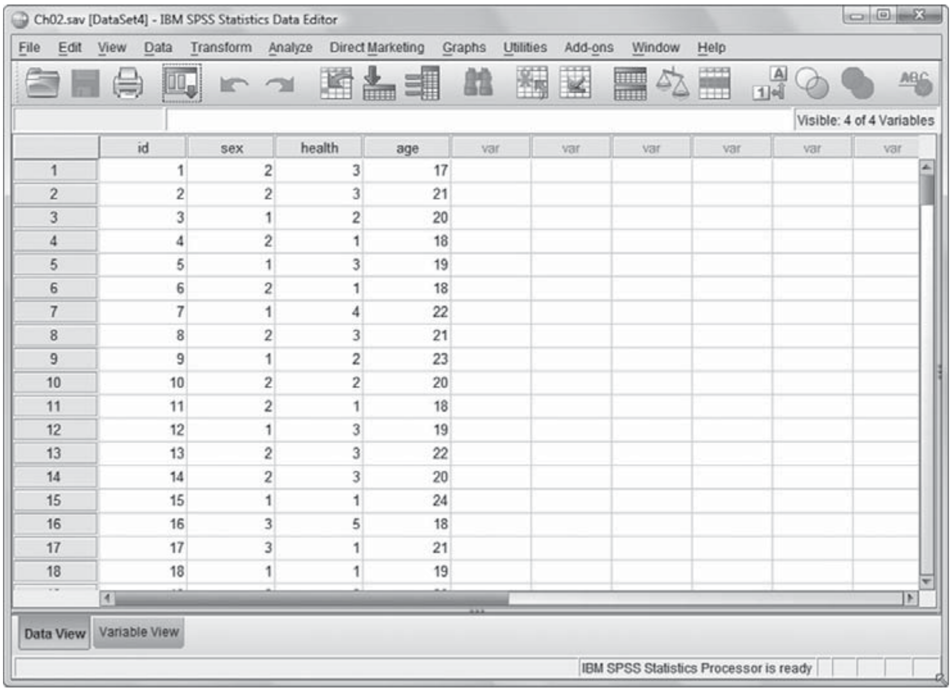


1 : id | 1 | Visible: 4 of 4 Variables

Figure 2.20 The SPSS **Cell Editor**

address is defined by the combination of the row number and column name that intersect at the active cell. If at any point we make a mistake, or we need to change the information in any particular cell, we simply make that cell active and type in the new information. On hitting the return key the new value will replace the old.

For your convenience, the data for the hypothetical survey of 200 students have already been entered into SPSS and saved in the file **Ch02.sav** on the website for this book (www.uk.sagepub.com/argyrous3). If you download this file, you can open it by selecting **File/Open** and then browse to the place on your computer to which the file has been downloaded (all the files that are used in this and later chapters can be obtained this way). If you open the **Ch.02.sav** file the **Data View** page will look like Figure 2.21.



Ch02.sav [DataSet4] - IBM SPSS Statistics Data Editor
File
Edit
View
Data
Transform
Analyze
Direct Marketing
Graphs
Utilities
Add-ons
Window
Help
Visible: 4 of 4 Variables
Data View
Variable View
IBM SPSS Statistics Processor is ready

	id	sex	health	age	var	var	var	var	var	var
1	1	2	3	17						
2	2	2	3	21						
3	3	1	2	20						
4	4	2	1	18						
5	5	1	3	19						
6	6	2	1	18						
7	7	1	4	22						
8	8	2	3	21						
9	9	1	2	23						
10	10	2	2	20						
11	11	2	1	18						
12	12	1	3	19						
13	13	2	3	22						
14	14	2	3	20						
15	15	1	1	24						
16	16	3	5	18						
17	17	3	1	21						
18	18	1	1	19						

Figure 2.21 The SPSS **Data Editor** after data entry

Checking for incorrect values: Data cleaning

Whether we enter data into SPSS ourselves or receive data from other people, before we actually analyse the data, we need to check that the data are 'clean'. Clean data are data that do not contain any invalid or nonsensical values; for example, we don't have someone with an age of 234 years, or someone coded 3 for sex, when we have coded 'females = 1' and 'males = 2'.

We check data to see if they need cleaning by generating frequency tables on all the variables (see Chapter 4). Frequency tables allow us to check to see that there are no values that are outside the permissible range. We can also use more elaborate analysis such as crosstabulations to assess whether particular groups within the data set only have values that are valid for them (see S. Pryjmachuk and D.A. Richards, 2007, Look before you leap and don't put all your eggs in one basket: The need for caution and prudence in quantitative data analysis, *Journal of Research in Nursing*, 12:1, pp. 43–54, for an excellent discussion of data cleaning procedures and the importance of doing them).

If we discover unusual responses during the data cleaning process, we make either of the following changes to the data set:

- *if it is a data entry mistake*, go back to the original data gathering instrument for that case, by cross-referencing the id number, and find the appropriate value to enter; or
- *if it is an invalid response*, type in an assigned missing value.

This process of tracking down possible missing values is made simple by a number of SPSS commands that are especially useful when working with large data sets.

Working with a large data set

The data set we have created is very small. There are only four variables and 200 cases. Moving around the data set is therefore relatively easy using the scroll-bars to the right and at the bottom of the window. With a large data set containing lots of variable and cases, scrolling can easily cause the numbers in the data set to blend into each other and make it difficult to find the particular entry we are trying to locate.

SPSS provides a number of commands that make a large data set manageable, and help track down specific values in a data set. These commands are summarized in Table 2.5.

Table 2.5 SPSS commands for working with large data sets

Command	Comments
Utilities/Variables/Go To	The active cell will be that determined by the intersection of the selected variable in the **Go To** command and the row of data in which the active cell was located before the command was invoked
Edit/Go to Variable	**Go to Variable** selects the entire column for the selected variable
Edit/Go to Case	**Go to Case** moves the active cell to that which intersects the selected row number and the column on which the active cell was located before the command was invoked
Edit/Find	Moves the active cell to that which meets the search criterion for the column on which the active cell was located before the command was invoked. Very useful during data cleaning when tracking down possible invalid values

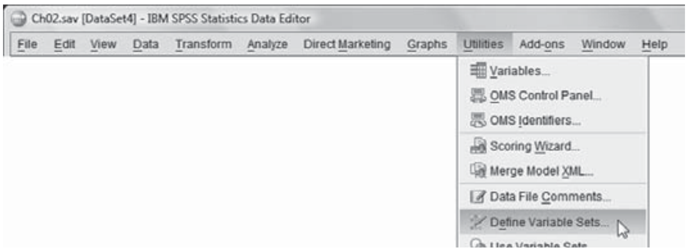


Figure 2.22 The SPSS **Define Variable Sets** command

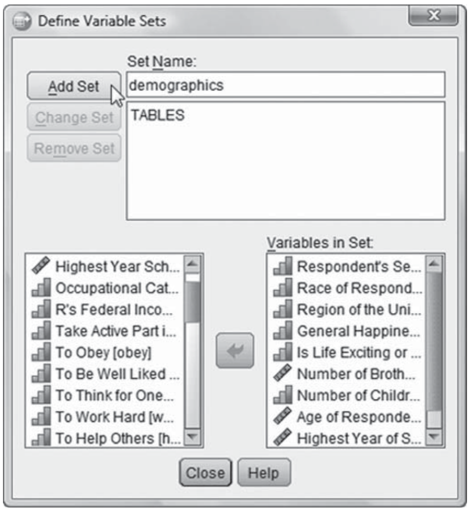
The other commands that help working with large data sets are the **Utilities/ Define Variable Sets** (Figure 2.22) and **Utilities/Use Variable Sets**. These commands can reduce the number of columns that appear in the **Data Editor** window, and also the number of variables that appear in a dialog box. To illustrate the use of these complementary commands, we will use a sample data set that comes with SPSS and which is also available from the book's website. This is the **1991 U.S. General Social Survey.sav** file.

There are 43 variables in this data set. If we look at these variables we can see that they form distinct groups:

1. A set of demographic variables such as **sex**, **race**, and **religion**.
2. A set of five variables that rate five behaviours from the most important to the least important.
3. A set of nine variables named **hlth1**, **hlth2**, **hlth3**, ..., **hlth9**. These are responses to questions asking whether a respondent experienced a particular health-related problem in the previous year, such as whether they were ill enough to go to a doctor.
4. A set of nine variables named **work1**, **work2**, **work3**, ..., **work9**. These are responses about work-related problems experienced in the previous year.
5. A set of four variables named **prob1**, **prob2**, **prob3**, and **prob4**. These indicate which of a listed range of problems were among the four most important problems faced by respondents in the previous 12 months.

Imagine that we are interested in doing some analysis of the relationship between the demographic characteristics of respondents and their responses to the nine questions about work-related problems. We might be interested in using the variables for other analysis, but for the time being we want to focus just on these two sets of variables. To simplify the **Data Editor** and reduce the number of variables we have to scroll through in dialog boxes, we first define five **variable sets** corresponding to the list above. For example, to create a set called **demographics** that includes all the demographic variables, we do the following (Figure 2.23):

1. Select **Utilities/Define Variable Sets**.
2. Move the demographic variables from the source list to the **Variables in Set:** list.
3. Enter a **Set Name:** such as **demographics**.
4. Click on **Add Set**.



Set Name: demographics

TABLES

Highest Year Sch...
Occupational Cat...
R's Federal Inco...
Take Active Part i...
To Obey [obey]
To Be Well Liked ...
To Think for One...
To Work Hard [w...
To Help Others [h...

Variables in Set:

Respondent's Se...
Race of Respond...
Region of the Uni...
General Happine...
Is Life Exciting or ...
Number of Broth...
Number of Childr...
Age of Responde...
Highest Year of S...

Figure 2.23 The SPSS **Define Variable Sets** window

We follow the same steps to create other sets called **MostImportant**, **health**, **work**, and **problems**. Note that the actual names we choose can be anything; I have used these names purely for the purpose of illustration.

Once we have created these sets, we then use the **Utilities/Use Variable Sets** command to select the cluster of variables that we want to appear. This is done by checking or unchecking the box next to the name of the respective variable set.

Summary

We have worked through the process of setting up an SPSS data file. Needless to say, we have only skimmed the surface with respect to the full range of options available. Chapter 28 details some of the more advanced functions available in SPSS: the **Recode** command, the **Multiple Response** command, the **Compute** command, and the **Select Cases** command. I also leave it to the reader to play around with SPSS and learn the full range of features that it provides. The program comes with its own tutorial and sample data files that will guide you through the various features. The program also comes with a very useful help facility, which is available from the menu bar.

EXERCISES

2.1 A survey gathers the data for the weekly income (in dollars) of 20 people, and obtains the following results:

0, 0, 250, 300, 360, 375, 400, 400, 400, 420, 425, 450, 462, 470, 475, 502, 520, 560, 700, 1020

Create an SPSS data file and enter these data, entering all the necessary labels. Save the file with an appropriate filename.

2.2 The following data represent time, in minutes, taken for subjects in a fitness trial to complete a certain exercise task:

31	39	45	26	23	56	45	80	35	37
25	42	32	58	80	71	19	16	56	21
34	36	10	38	12	48	38	37	39	42
27	39	17	31	56	28	40	82	27	37

The heart rate of each subject is also recorded in the same sequence as their time score:

63	89	75	80	74	65	90	85	92	84
74	79	98	91	87	76	82	90	93	77
74	89	85	91	102	69	87	96	83	72
92	88	85	68	78	73	86	85	92	90

The first 20 of these scores (reading from left to right) are taken from males and the second 20 from females. Create an SPSS data file and enter these data, entering all the necessary labels. Save the file with an appropriate filename.

2.3 A research project has collected data from 10 people on the following variables:

Television watched per night (in minutes)	Main channel watched	Satisfaction with quality of programmes
170	Commercial	Very satisfied
140	Public/government	Satisfied
280	Public/government	Satisfied
65	Commercial	Very satisfied
180	Commercial	Not satisfied
60	Commercial	Not satisfied
150	Public/government	Satisfied
160	Commercial	Not satisfied
200	Public/government	Satisfied
120	Commercial	Not satisfied

Prepare an SPSS data file for these data, creating variables and variable labels, values and value labels.

2.4 From the website for this text (www.uk.sagepub.com/argyrous3) download the following articles:

N. Dibben and V.J. Williamson, 2007, An exploratory survey of in-vehicle music listening, *Psychology of Music*, 35:4, pp. 571–89.
S.A. Creedon, 2006, Health care workers' hand decontamination practices, *Clinical Nursing Research*, 15:1, pp. 6–26.

Identify which article indicates the use of SPSS to conduct its statistical analysis and how this is reported.