# 4.4 The Compute Procedure: Creating Variables

In the current data set, it would be quite normal for the teacher to instruct the computer to create new variables that calculate total points earned as the sum of the quizzes and final exam, and to determine the percent of total points for each student. The sequence of steps that follows will compute two new variables called **total** and **percent**. As noted earlier, the **grades.sav** file (available for download at *www.spss-step-by-step.net*) already contains the four new variables computed in this chapter: **total**, **percent**, **grade**, and **passfail**.

*Beginning with a screen that shows the menu of commands, perform the following step to access the **Compute variable** window.*

In Screen	Do This	Step 4c
Menu	✳ **Transform** ➔ ✳ **C̲ompute Variable…**	4.3

At this point, a new window opens (Screen 4.3, below) that allows you to compute new variables. The list of variables is listed in a box to the left of the screen. Above these is the T̲arget Variable box. In this box you will type the name of the new variable you wish to compute. To the right is the **Numeric Expression** box. In this box you will type or paste (usually a combination of both) the expression that will define the new variable. Three options are then provided to assist you in creating the expression defining the new variable.

Screen 4.3 The Compute Variable Window
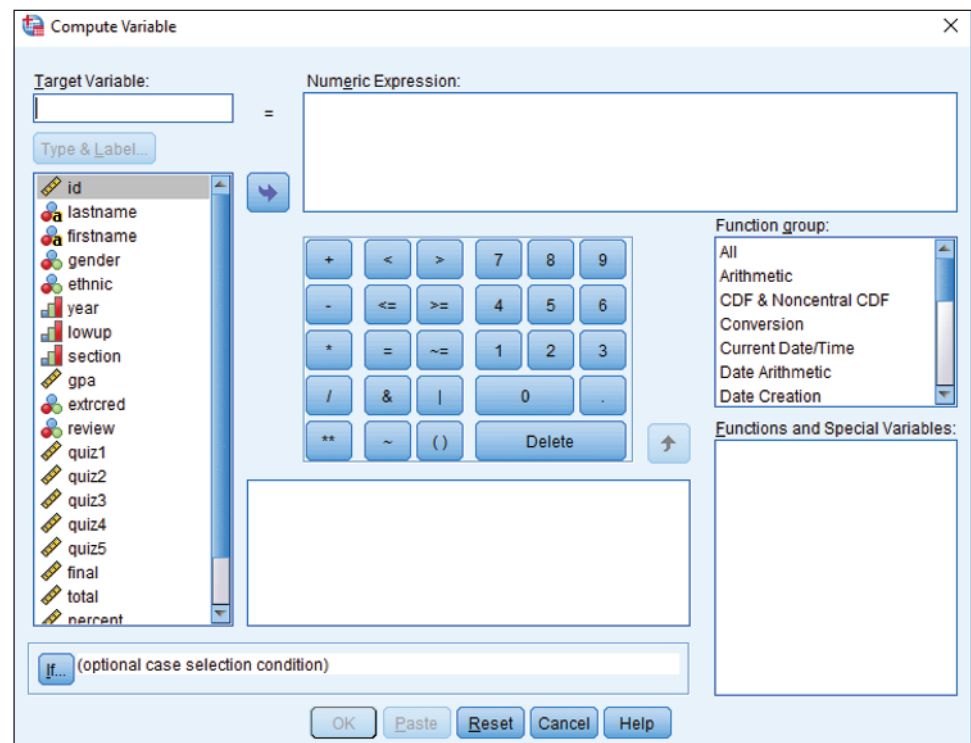
- *The calculator pad*: On this pad are all single-digit numbers and a decimal point. Each of these may be entered into the **Numeric Expression** by a mouse click on the screen button or by typing the same on the keyboard. In addition there are a number of *operation* buttons. If a similar button occurs on the keyboard (e.g., <, >, +, etc.) you may type that symbol rather than click the screen button. The operation keys and their meanings follow:

Arithmetic Operations		Rational Operations		Logical Operations	
+	add	<	less than	&	and: both relations must be true
–	subtract	>	greater than		
*	multiply	<=	less than or equal	|	or: either relation may be true
/	divide	>=	greater than or equal		
**	raise to power	=	equal	~	negation: true becomes false, false becomes true
( )	order of operations	~=	not equal		

- *The* U̲nderlined: *Functions boxes*: A terrifying array of over 180 different functions emerge. We present only nine of these functions; the nine we feel are most likely to be used. To help create order, SPSS has created three boxes in the screen (Screen 4.3) to assist:

  1. The box labeled **Function group** provides different categories of functions. For instance, in Screen 4.3 the Arithmetic function may be selected. This provides (in the window below) just 13 functions (rather than 188)—much easier to negotiate. The first seven functions in the chart on this page (below) are under **Arithmetic**, the final two are under **Random numbers**.

  2. The box labeled **Functions and Special Variables** lists the available options in each category. If you select **All** (in the box above), then all 188 functions will be listed alphabetically.

  3. The box to the left of the **Functions and Special Variables** box provides the term used in the chart on this page (below) and identifies what the function does. In this case **Abs** is highlighted (box to the right) and the definition of **ABS(numexpr)** is provided.

An example of a sample **Target Variable**, **Numeric Expression**, *and how it would compute in your file follows. Please refer back to Screen 4.3 for visual reference.*

Expression	Illustration
ABS(numexpr) *(absolute value)*	Target variable: **zpositiv** ⟶ Numeric expression: **ABS(zscore)**. Creates a variable named **zpositiv** that calculates the absolute value of a variable named **zscore** for each subject.
RND(numexpr) *(round to the nearest integer)*	Target variable: **simple** ⟶ Numeric expression: **RND(gpa).** Computes a variable named **simple** by rounding off each subject's **gpa** to the nearest integer. Note: use "**Rnd(1)**" and (next entry) "**Trunc(1).**"
TRUNC(numexpr) *(truncates decimal portion of a number)*	Target variable: **easy** ⟶ Numeric expression: **TRUNC(gpa)**. Computes a variable named **easy** that truncates each subject's **gpa** (truncate means "cut off"). This is like rounding, but it always rounds down.
SQRT(numexpr) *(square root)*	Target variable: **scorert** ⟶ Numeric expression: **SQRT(score)**. Creates a new variable, **scorert**, by square rooting each **score** for each subject.
EXP(numexpr) *(exponential (e) raised to a power)*	Target variable: **confuse** ⟶ Numeric expression: **EXP(gpa).** Computes a variable named **confuse** that calculates the value of e raised to each subject's **gpa's** power. e ≈ 2.721 (it is irrational). For a particular subject with a 3.49 **gpa**, **EXP**(3.49) = $(2.721\ldots)^{3.49} \approx 32.900506\ldots$
LG10(numexpr) *(base 10 logarithm)*	Target variable: **rhythm** ⟶ Numeric expression: **LOG10(total)**. Creates a new variable named **rhythm** that calculates the base 10 logarithm for the variable **total** for each subject.
LN(numexpr) *(natural logarithm)*	Target variable: **natural** ⟶ Numeric expression: **LN(total)**. Creates a new variable named **natural** that calculates the natural logarithm for the variable **total** for each subject.
RV.NORMAL(mean, stddev)	Computes random numbers based on a normal distribution with a user-specified mean (**mean**) and standard deviation (**stddev**). Target variable: **randnorm** ⟶ Numeric expression: **RV. NORMAL(5,3).** This function will generate random numbers based on a normal distribution of values with a mean of 5 and a standard deviation of 3. One of these numbers will be assigned randomly to each subject (or case) in your data file under the variable name **randnorm**.
RV.UNIFORM(min, max)	Computes random numbers based on equal probability of selection for a range of numbers between a user-specified minimum (**min**) and maximum (**max**). Target variable: **random** ⟶ Numeric expression: **UNIFORM(1,100).** Random numbers from 1 to 100 will be assigned to each subject (or case) in your data file.

• *The **If** pushbutton*: A click on this button opens a new screen so similar to Screen 4.3 that we will not reproduce it here. The only differences are an **Include all cases** option paired with an **Include if case satisfies condition**, and a **Continue** button at the bottom of the window.

In creating more complex computations, be sure to adhere strictly to the basic algebraic rules of order of operations. If you wish to do an operation on a complex expression, be sure to include it within parentheses.

Two examples illustrate computing variables: Computing z scores and T scores. In the first example, we will compute z scores for **total** class grades. If by chance you have not learned the equation for z scores yet, it is easy: For each person, you take her or his score and subtract from it the mean (average) of everyone's score. Then, you divide that number by the standard deviation of those scores. (You can read more about means and standard deviations in Chapter 7, page 114.) For total, the mean score is 100.57 and the standard deviation is 15.299, so:

$$z = (\textbf{total} - 100.57) \, / \, 15.299$$

z scores are sometimes called standardized scores, because the mean of every set of z scores is the same (0), and the standard deviation is the same (1).

*The starting point for this operation is Screen 4.3. To compute the new variable **totalz** perform the following sequence of steps:*

In Screen	Do This												Step 5c
4.3	**type** totalz ⟶ **press** () ⟶ ✳ **total** ⟶ ✳ ➜ [*or* ✳ ✳ **total**] ⟶
	type -100.57 ⟶ ✳ [*to the right of the equation so far*] ⟶ **press** / ⟶
	type 15.299 ⟶ ✳ **OK**

With the click of the **OK**, the new variable is computed and entered in the data file in the last column position. If you wish to move this new variable to a more convenient location, you may cut and paste it to another area in the data file. (Important confession: Because z scores are used so frequently, there is actually an easier way to create z scores. For that method, see Chapter 7, page 117.)

For the second example, we will be computing T scores from the z scores that were just calculated. T scores are also standardized, with a mean of 50 and a standard deviation of 10. They are commonly used in some areas of psychology, because it's nicer to tell someone "Your depression T score is 30" than "Your depression z score is -2." To calculate a T score, multiply the z score by 10 and add 50. T scores are usually rounded to the nearest whole number.

*The starting point for the next operation is also Screen 4.3. If a previous equation is already in the box, click on **Reset** first. To compute the new variable **totalT** and save in the data file the two variables we have created, perform the following sequence of steps:*

In Screen	Do This										Step 5c'
4.3	**type** totalT ⟶ ✳ **Arithmetic** *in the* **Function group** *box* ⟶ *scroll down in the* **Functions and**
	Special Variables *box and* ✳ **Rnd(1)** ⟶ ✳ ⬆ ⟶ **type** 10*totalz+50 ⟶ ✳ **OK**
Data	✳ **File** ⟶ ✳ _Save

Upon completion of a compute operation, SPSS returns the screen to the data file and you may use the scroll bar to check the accuracy of the new variable(s).