

Stata also allows us to replace certain values in the dataset. This is done using the **replace** command. There are several uses of this command. For example, we already know that the highest gpa in our dataset is 96, and we know that this grade has been achieved by a single student. To verify this, we can execute the following:

```
list if gpa == 96
```

Once again, I would like to point out that we use two equal signs. Note the value of `_n` which is included in the output. It is important and we will use it in a while.

What if it comes to our attention that this student actually went to University 2, and not University 1. How can we modify this? Here, it is crucial that we remember that the variable *university* is actually numeric even though the values that we see are text. The text that we see is the label that is attached to this variable. To verify this we run the **codebook** command:

```
codebook university
```

The output tells us that the label University 1 is actually stored as a zero, while the label University 2 is stored as a one. This means that what we want to do is to change the zero to a one, but we only want to do this for this particular observation:

```
replace university = 1 if _n == 223
```

The **replace** command is how we change values. The command that we executed is telling Stata to change the value of the variable *university* to 1 only for the observation with an `_n` value of 223, which is the observation that we had previously listed. Stata tells us that a single change has been made to the dataset. Let us make sure that the change was made correctly:

```
list if gpa == 96
```

We see that the value of *university* has been correctly changed.

What if we wanted to change the value of a variable for more than one observation? This is extremely easy to do. All we need is to include the correct condition. For example, what if we were told that University 2 did not actually offer a course on business ethics, and that the students who answered yes had taken a course on the philosophy of ethics. The students thought that the question in the survey was referring to this course. In that case, we need to replace all values of the variable *course* for all students who attended University 2. We know that students who attended University 2 have a value of 1 stored in the variable *university* (we saw this when we executed the **codebook** command). We now need to know how the variable *course* was coded:

```
codebook course
```

We see that we need to set the value of this variable to 1 for all students who attend University 2. This is done using the following command:

```
replace course = 1 if university == 1
```

Stata tells us that 61 changes have been made. To verify that everything is in order, we can use the **tabulate** command:

```
tabulate university course
```

The output clearly shows that all students who went to University 2 are now recorded as not having taken this course.

It is also possible to change the value of a variable for all observations. This is actually something that I did for this dataset before sharing it with you. I did not want the dataset to contain the correct *gpa* values of the students who participated in this survey. The way I changed the values is slightly more complex than what I want to cover in this course, but just as an example, assume that I just went ahead and added five points to all *gpa* values:

```
replace gpa = gpa + 5
```

Stata tells us that 676 changes were made. Why only 676 and not 818? This is because Stata does not change the missing values. A missing value remains a missing value. If we get the summary statistics for this variable now we will notice something that is clearly wrong:

```
summarize gpa
```

We see that the maximum value of the variable is now 101. This is because the highest *gpa* in the dataset was 96, and we just added five to it. This is clearly wrong. We need to make sure that the highest *gpa* is 100. You might think that this should be accomplished by the following command:

```
replace gpa = 100 if gpa > 100
```

This, however, is very wrong. The reason is that it has been previously mentioned that missing values are larger than any number. Therefore, a missing value satisfies the condition “> 100” and hence all missing values of the variable *gpa* will be changed to 100. This is clearly not what we want. What we want is to replace the nonmissing values that are greater than 100. Therefore, the correct command is:

```
replace gpa = 100 if gpa > 100 & !missing(gpa)
```

This command tells Stata to replace the values of *gpa* that are greater than 100 AND that are not missing. Missing values satisfy the first condition but they do not satisfy the second condition and hence they are excluded from the operation.

Since I don’t want any of the changes to be saved, it would be saved to clear the data:

```
clear all
```