

Exercise 2: Appending and making new REC files EpiData Analysis

At the end of this exercise you should be able to:

- a. Differentiate between merging and appending data files.
- b. Append different data files into a single data file
- c. Drop some fields from the data set
- d. Creating value labels for new numeric fields

Sometimes, it is useful to *merge* several data files or to *append* some data files to an existing one. For the definitions of MERGE and APPEND and how to proceed you may refer to EpiData Entry. The following graphic outline summarizes the difference between the procedures.

It is common that different data files exist and a researcher wishes to combine them into a single set. If we take two data sets, set A and set B, we can imagine different possibilities.

Set A and set B have the same variables:

Set A			
ID	VAR1	VAR2	VAR3
A	33	A	2
B	21	B	3
C	24	X	7
D	44	Y	8
E	56	C	2
Set B			
F	74	T	1
G	67	A	3
H	34	B	2
I	2	J	1

If we want to combine the data set A and data set B, we will have to **append** set B to set A.

Set B may contain the same variables as set A plus some additional variables:

Set A (and Set B)				Set B only			
ID	VAR1	VAR2	VAR3	ID	VAR4	VAR5	VAR6
A	33	A	2				
B	21	B	3				

C	24	X	7				
D	44	Y	8				
E	56	C	2				
F	74	T	1	F	ER	DFG	6
G	67	A	3	G	YT	DFG	7
H	34	B	2	H	CX	ERT	8
I	2	J	1	I	EW	CVB	1

It is of course also possible that set B contains only variables that are not contained in set A:

Set A				Set B			
ID	VAR1	VAR2	VAR3	ID	VAR4	VAR5	VAR6
A	33	A	2				
B	21	B	3				
C	24	X	7				
D	44	Y	8				
E	56	C	2				
				F	ER	DFG	6
				G	YT	DFG	7
				H	CX	ERT	8
				I	EW	CVB	1

or any combination with some variables contained in both sets:

Set A (and Set B)				Set B (and set A)			
ID	VAR1	VAR2	VAR3	ID	VAR4	VAR5	VAR6
A	33	A	2				
B	21	B	3				
C	24	X	7				
D	44	Y	8				
E	56	C	2				
F			1	F	ER	DFG	6
G			3	G	YT	DFG	7
H			2	H	CX	ERT	8
I			1	I	EW	CVB	1

In all these cases, we will have to **merge** the two data sets.

In brief, we might summarize four rules on when to use APPEND and when to use MERGE:

- Append – add records (rows) when data structure is identical;
- Merge – add variables (columns) based on unique identifier;
- Adding records and variables – use merge;
- Merge function can do what append can do and more

Thus, APPENDING is the procedure where several files with exactly the same fields and field definitions are combined. You may have prepared a single QES file and created from that one questionnaire file several identical REC and CHK files for data entry in various locations.

With the previous exercise B_EX01 you obtained four supplementary EpiData REC files:

A.REC
B.REC
C.REC
D.REC

The data for these files were obtained from four different Tuberculosis laboratory registers and they contain the following fields:

Field name	Field label	Field type	Field length	Field values	Value labels	Comment
Id	Unique laboratory identifier	T	6	Any		Any unique identifier as a combination of LABCODE and SERNO
Serno	Laboratory serial number	I	4	1001,,1075 2001,,2075 3001,,3075 4001,,4075		
labcode	Laboratory code	T	1	A B C D	Laboratory A Laboratory B Laboratory C Laboratory D	
regdate	Registration date	D	10	01/01/1980, ...,31/12/200 2, 01/01/1900		Date of registration known Unknown date of registration
sex	Examinee's sex	I	1	1 2 9	Female Male Sex not recorded	
age	Examinee's age in years	I	2	0, ... ,97 98 99		Known age in years 98 years or older Unknown age
reason	Examination reason	I	1	0 8 9 1 2 3 4 5 6 7	Diagnostic examination Follow-up, month not stated No reason recorded Follow-up at 1 month Follow-up at 2 months Follow-up at 3 months Follow-up at 4 months Follow-up at 5 months Follow-up at 6 months Follow-up at 7 m or later	
res1		I	3	0.0 0.1,,0.9 1.0 2.0 3.0 4.0 8.0 9.0	No bacilli found 1 to 9 AFB per 100 fields 1+ positive 2+ positive 3+ positive Scanty, no quantification Positive, no quantification No result recorded	
res2		F	3	0.0 0.1,,0.9 1.0 2.0 3.0 4.0 8.0 9.0	No bacilli found 1 to 9 AFB per 100 fields 1+ positive 2+ positive 3+ positive Scanty, no quantification Positive, no quantification No result recorded	
res3		F	3	0.0 0.1,,0.9 1.0 2.0 3.0 4.0 8.0 9.0	No bacilli found 1 to 9 AFB per 100 fields 1+ positive 2+ positive 3+ positive Scanty, no quantification Positive, no quantification No result recorded	

Appending files and making a new data file

The beginning of the program looks familiar, but has added two new components shown in bold:

- * This B_EX02.PGM appends three files
- * to a first file and creates a new
- * REC file

```
cls
logclose
close

read "a.rec"
```

```
append /file="b.rec"  
append /file="c.rec"  
append /file="d.rec"  
savedata "abcd.rec"
```

```
close  
read "abcd.rec"
```

The command APPEND adds another file with the same structure to the file that was read in first (A.REC) and at the end of appending all three files (giving a total of four files), the command SAVEDATA followed by the name of the new REC file saves that data file.

This program will run fine if we run it only once. However, if we run it a second time, we will get an error message that the ABCD.REC file cannot be saved because it exists already. We must therefore ensure that the ABCD.REC file does not exist when the program starts to run. To this end, we add another command line before we even read the first file:

```
* This B_EX02.PGM appends three files  
* to a first file and creates a new  
* REC file
```

```
cls  
logclose  
close
```

```
read "a.rec"  
append /file="b.rec"  
append /file="c.rec"  
append /file="d.rec"  
savedata "abcd.rec" /replace
```

```
close  
read "abcd.rec"
```

This will work even if we run the program for the first time: the first time, there is simply no file to be erased and EpiData Analysis will report so and continue to execute the program.

Let's now say that we do not want to keep the identifiers (fields ID and SERNO). We can tell EpiData Analysis to drop these two fields:

```
* This B_EX02.PGM appends three files  
* to a first file and creates a new  
* REC file
```

```
cls  
logclose  
close
```

```
read "a.rec"  
append /file="b.rec"  
append /file="c.rec"  
append /file="d.rec"  
var drop id serno  
savedata "abcd.rec" /replace
```

* Note: instead of "var drop" you may use more simply just "drop"

```
close
read "abcd.rec"
```

If you create a new variable, this variable can also be made with numeric coding with a numeric field value and a text as value label, as you learned in EpiData Entry. For instance, you wish to have only two values for the first result (RES1), positive or negative, then you could define:

```
define result1 #
result1=0
if res1>0 and res1<9 then result1=1
labelvalue result1 /0="Negative" /1="Positive"
label result1 "Result of first examination"
```

Use this approach in solving the following task for the case definition.

Task:

- o Start with the program B_EX01.PGM, save it as B_EX02.PGM, and edit it as needed to make the new dataset ABCD.REC.*
- o In the same program, create a new dataset B_EX02.REC file that has a new variable CASE which can be either positive or negative and one for WHO age groups. Define as positive any examinee who has at least one bacillus in at least one result and make a tabulation by case (column) and age group (row), stratified by a new variable that groups reason into diagnosis, follow-up (irrespective of month) and unknown reason. Calculate row percentages.*