

**Designing a CAPI DATA  
ENTRY Application  
using CPro :  
Module 2**



## Summeries

|      |   |    |
|------|---|----|
| I.   | DATA ENTRY APPLICATION FILE .....   | 5  |
| A.   | How to move from a traditional CSPro Data Entry to a CAPI application ..... | 5  |
| B.   | Design the Field Capture Type.....  | 8  |
| II.  | Adding CAPI questions and texts.....  | 11 |
| A.   | Add CAPI questions .....  | 12 |
| 1.   | Customize a CAPI Question .....   | 13 |
| 2.   | Add condition for CAPI Question.....  | 14 |
| B.   | Create Helps for Fields.....  | 16 |
| C.   | Add Pictures .....  | 16 |
| III. | Add multi language capi question .....                                      | 16 |
| IV.  | Read, Write In an External File.....  | 19 |
| A.   | Using lookup files .....  | 20 |
| B.   | Write in an external file.....  | 22 |
| C.   | Write in an external form .....   | 23 |

## **Introduction**

I'd like to thank whole team of FAO whose ceaseless efforts have allowed a perfect organization of my mission. I also thank the staff of the CSA and particularly the Deputy Director General, Operation, Methodology and Data processing and the team of programmers for not only the logistical arrangements they made for the working sessions to be held in a friendly atmosphere but also for their unconditional availability

This guide presents step by step the procedures for programming CSPro CAPI data entry. Notes you have written during on-the job training, your curiosity and your own work will deepen your knowledge about the subject. This guide is based on the CSPro's HELP, our personal experiences, the experience of CSA programmers (whose name are in appendix) and many other documents in the references.

This is the module 2 of the manual. It is assumed that the user has basic knowledge of CSPro. This also assumes that you have mastery of the three steps below:


- Create a dictionary
- Generate and design a Data entry application (Form)
- Write logic

If all three aspects above are not mastered, please refer to the Module 1

## **I. DATA ENTRY APPLICATION FILE**

“The Data Entry Application file is the master file for a data entry application. This file specifies all other files contained in the application, along with other information. CSPro allows you to open, close and save data entry application files. When you do so, all other files associated with the application are also opened, closed or saved”. (CSPro User’s Guide)

To create a new data entry application file do as describe below

- click  on the toolbar,
- or from the File menu, select New.

And follow the instruction as describe in the module 1 of the manual.

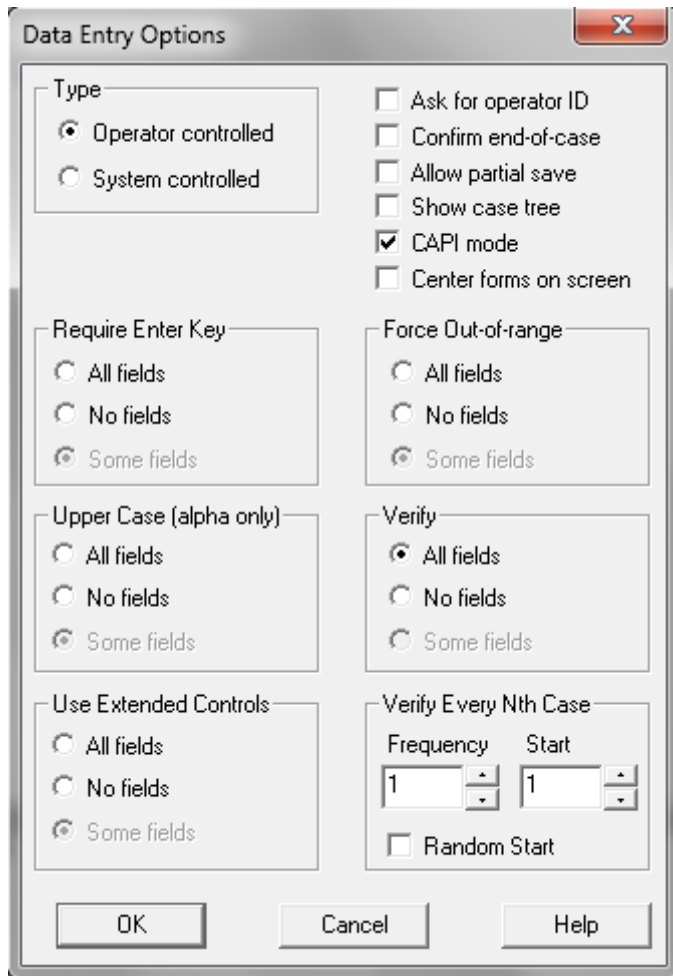
### ***A. How to move from a traditional CSPro Data Entry to a CAPI application***

CAPI means Computer-Assisted Personal Interviewing.

To move from a traditional CSPro Data Entry to a CAPI application, Data entry option has to be change.

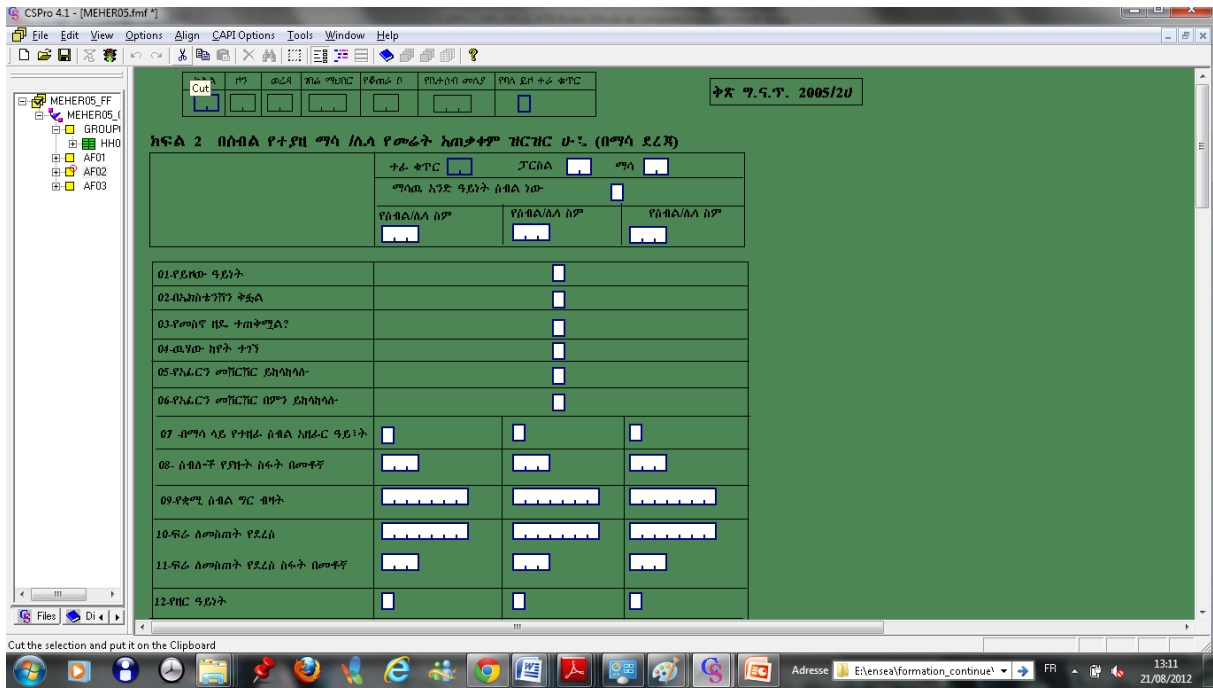
Select "Data Entry" from the Option menu and check “CAPI mode”. Indeed, If this box is checked, CSEntry will display the Computer-Assisted Personal Interviewing (CAPI) window.

The top part is for question text (to be read during the interview), the bottom is for the normal form contents.

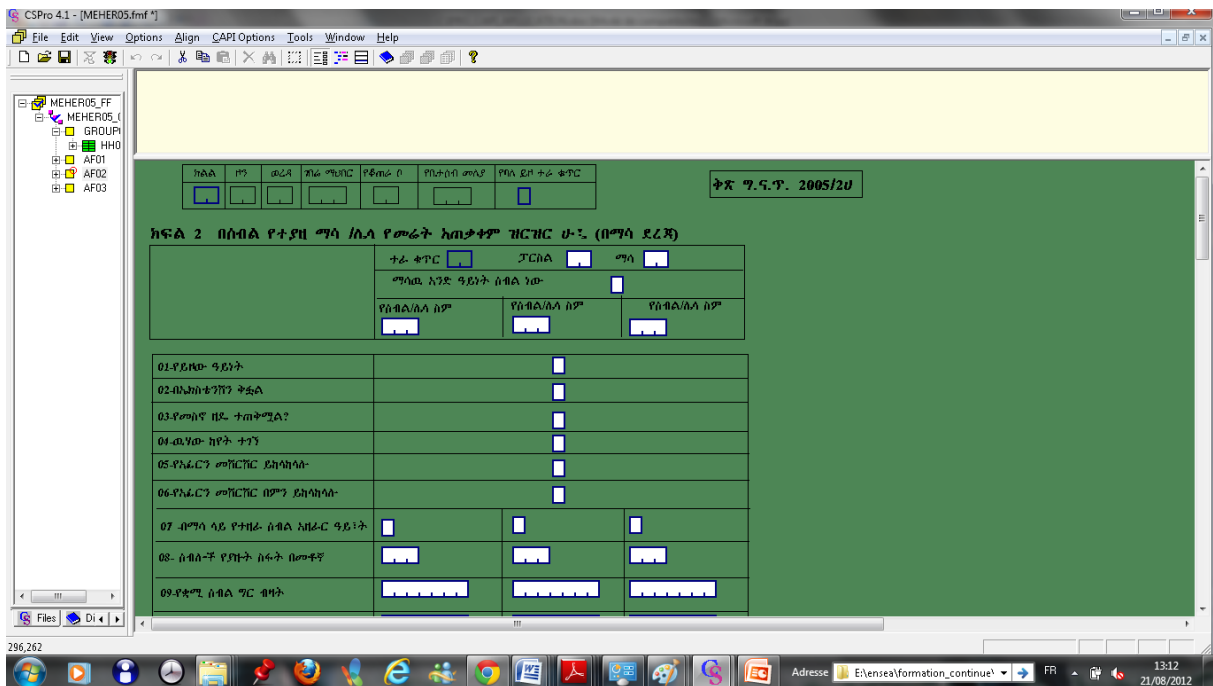


You can also check “*Use Extended Controls*”. CSPro 4.1 introduces the concept of extended controls. These are dialog boxes that appear during data entry that allow users to choose values from a list rather than key them. These controls are useful for applications run on tablet computers.

DE before checking “CAPI MODE”



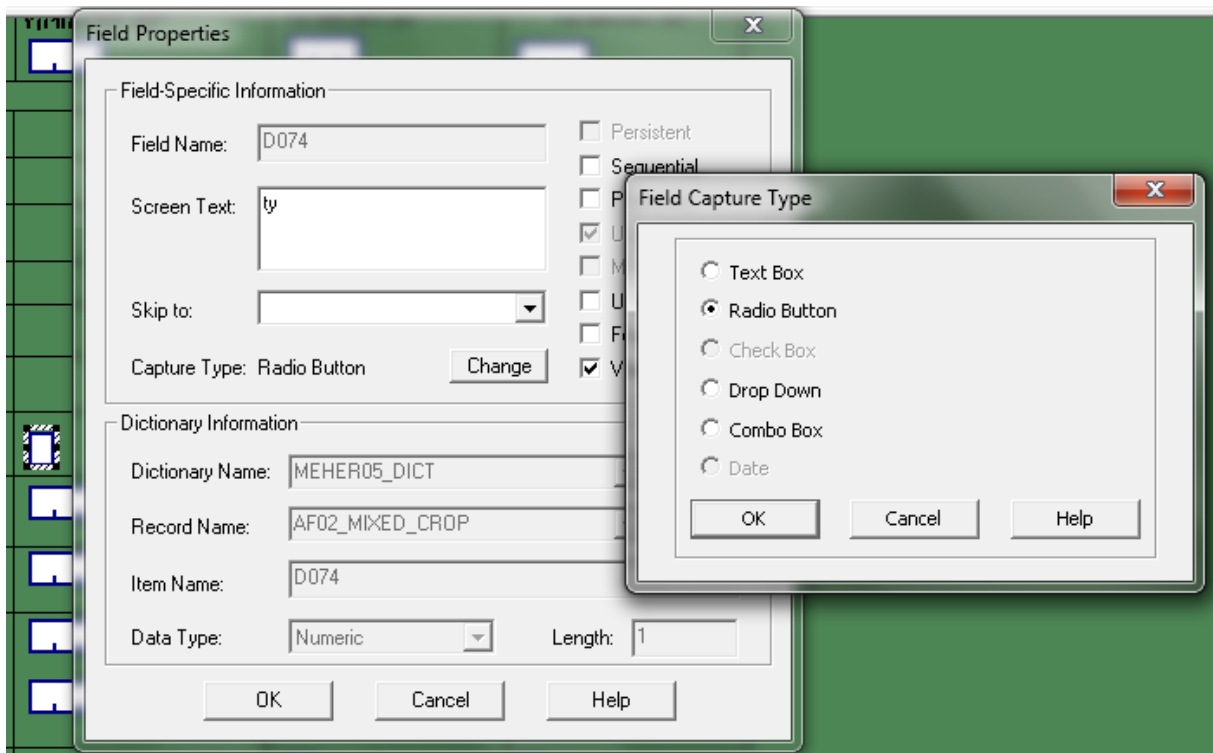
## DE after checking “CAPI MODE”



## B. Design the Field Capture Type

The choice of the field type is useful for applications CAPI application. Indeed, it may be useful to display to an enumerator or keyer the possible values for a given field in a data entry application. With CAPI application, it may be easier for an enumerator to use a finger or stylus to select a value rather than to key the value using the screen's touch keyboard.

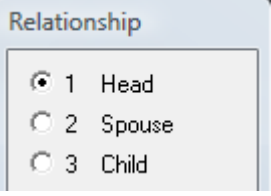
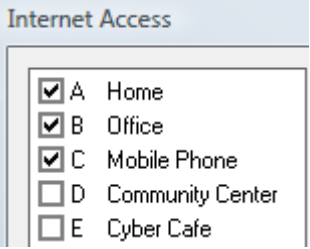
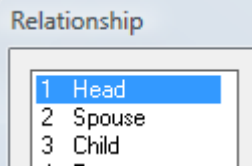
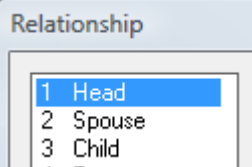
To manually change the capture type for a field, view the field's properties and select “Capture Type -> Change”.



“CSPro only allows you to select a capture type that is suitable for the item's first value set. For more programmatic control, see the setcapturetype function. The CSPro Designer shows the fields that use extended controls by coloring the border of the field in blue. The blue border does not exist when the data entry application is run in CSEntry” (CSPro User’s Guide)

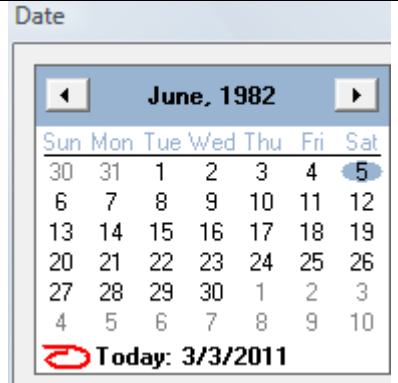
| Field | Condition of using | example |
|-------|--------------------|---------|
|-------|--------------------|---------|



|                     |  |   |
|---------------------|--|---|
| <b>capture type</b> |  |   |
| <b>Radio Button</b> | <p>The field must have discrete values defined in the first value set in the data dictionary. Each value will be displayed as a separate radio button on the Pocket PC.</p>  |    |
| <b>Check Box</b>    | <p>Check Boxes are used to capture a "multiple response" variable. To do this, you must first set up the item and its value set in the data dictionary in the following way. The item must be of type "Alpha", and must have a length equal to the number of possible responses. The value set must have that same number of discrete values. Each value corresponds to one of the responses and will be displayed as a separate check box on the Pocket PC. When the boxes are checked, the corresponding values will be put into the alpha field, from left to right.</p> <p>For example, consider a multiple response question, "Which of the following do you have in the household". The possible answers are "Radio", "Television", "Computer".</p> <p>In this example you would define an "alpha" item in the data dictionary with length three and make a value set with three values:</p> |    |
| <b>Drop Down</b>    | <p>The field must have a value set in the data dictionary. Each value in the first value set will be displayed as a separate line in the response area. The user can tap on the appropriate response, or enter the numerical value.</p>  |  |
| <b>Combo Box</b>    | <p>The field must have a value set in the data dictionary. Each value in the first value set will be displayed as a separate line in the response area. The user can tap on the appropriate response, or enter the numerical value. This is very similar to the Drop Down, except that the user can tap on the small arrow on the right to open and close the list.</p>  |  |
| <b>Text Box</b>     | <p>The is the capture type when the field has no values defined in the data dictionary. The user must enter a response via the virtual keyboard.</p>   |   |

**Date**

The field must have a length of 4, 6, or 8. The choice of "Date format" will determine how the data will be stored in the data file. This capture type gives the user a nice visual interface from which to choose the date.

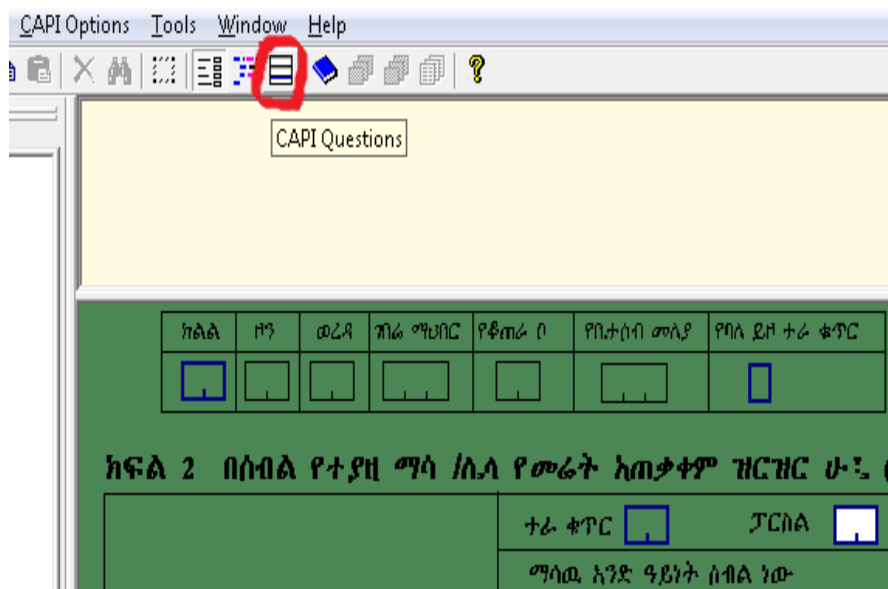


Source: CSPro User's Guide.

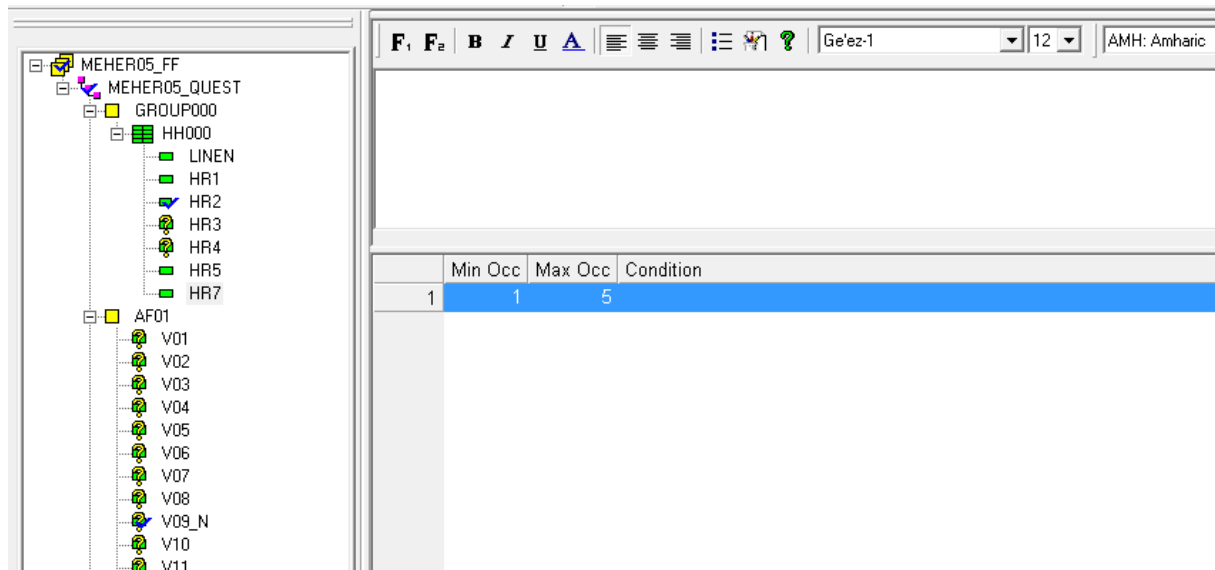
## II. Adding CAPI questions and texts

The peculiarity of the using of the CAPI-DE application is for not only to redefine the questions for each variable but also for personalizing the screen text in order to make them more suitable for CAPI.

To add CAPI question to an item: Make sure the Form tree is shown in the left pane (click the Forms tab on the bottom of the window), and highlight the item you want to make the question for. Then click the CAPI questions button on the toolbar (shown below):



After clicking on the capi Question button, the following window will appear (with the capi question editor in the right pane)

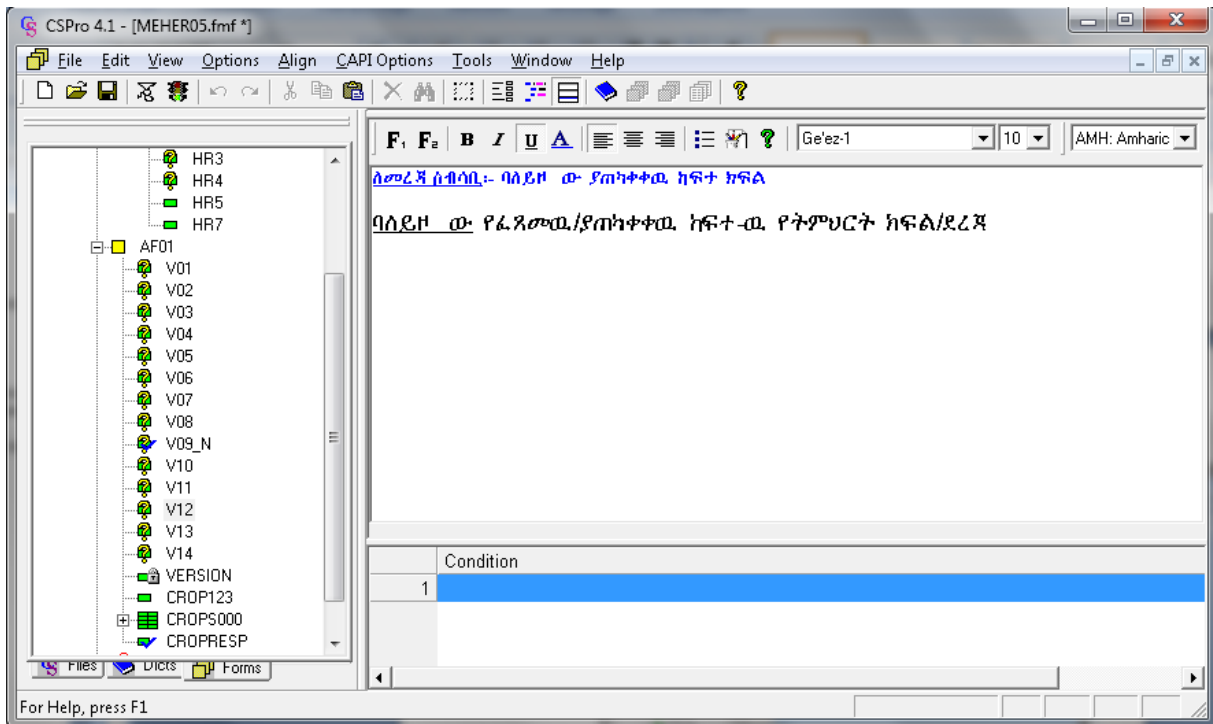


The “Capi Question” editor can be used for three purposes:

- Add CAPI question
- Add (write) help for question (to explain how to fill a variable field or Under what conditions the variable must be filled)
- Add pictures

#### *A. Add CAPI questions*

The CAPI question editor has two default fonts. The first one can be used for the questions, and the other one to use for additional comments or information or instructions to the interviewer or otherwise.



In the example above, the blue text is instruction for enumerators and the black one is the question.

The use of the fonts is possible by clicking on “F1” or “F2”. Write the text in the “CAPI QUESTION” editor, select it and according to your needs click on “F1” or “F2”.

## 1. Customize a CAPI Question

“Question text can be customized using fills, or text substitutions. These fills reference the contents of variables or dictionary items that can be embedded into the question text. They are identified by name of the variable or item surrounding by percent characters (%). During the interview, the value of the variable or data item is substituted into the question text. For example, "How old was %FIRST\_NAME% on January 1?" might be transformed into "How old was Edward on January 1?" Fills help customize the question text and frame it for specific respondents.” (CSPro User’s Guide)

**Note:** the Capi Question editor cannot recognize cspro functions (getlabel, ...). So that you cannot use directly the label or valueset associated to a code. You have to proceed as describe below:

Step1: define a variable will contain the valuetset label associated with the corresponding valueset code (ex: if \$ in 1:10 then CROPNAME=getlabel(\$,\$); endif;).

Step2: insert the variable in the question (ex: Which kind of sowing do you use for %CROPNAME%)

## 2. Add condition for CAPI Question

Your CAPI application can also present different sets of question text based on conditions. For example, the text for certain questions might differ based on the number of persons in the household or the row of the information. Conditions can also let your application support multiple kinds of questions, depending on criteria you decide.

### a) Multi-occurrence record type.

Consider the table below. We assume that it is a part of “industry survey questionnaire”.

|   |           |
|---|-----------|
| How many Permanents employees do you have last year | /_/_/_/   |
| How many temporary employees do you have last year  | /_/_/_/   |
| Total of employees                                  | /_/_/_/_/ |

In the dictionary, theses question can be define as a multiple occurrences

| <u>Item Label</u>   | <u>Item Name</u> | <u>Len</u> | <u>Data Type</u> | <u>Item Type</u> | <u>Occ</u> | ... |
|---------------------|------------------|------------|------------------|------------------|------------|-----|
| Number of employees | EMPNUMBER        | 3          | Num              | Item             | 3          | ... |

The screenshot shows a software interface with a text input field containing the question "how many Permanents employees did you have last year?". Below the input field is a table with the following structure:

|   | Min Occ | Max Occ | Condition |
|---|---------|---------|-----------|
| 1 | 1       | 1       |           |

how many temporary employees did you have last year?

|   | Min Occ | Max Occ | Condition |
|---|---------|---------|-----------|
| 1 | 1       | 1       |           |
| 2 | 2       | 2       |           |

|   |
|---|
| Add Condition      Ctrl+A<br>Modify Condition    Ctrl+M<br>Delete Condition    Ctrl+Del |
|---|

total of employees

|   | Min Occ | Max Occ | Condition |
|---|---------|---------|-----------|
| 1 | 1       | 1       |           |
| 2 | 2       | 2       |           |
| 3 | 3       | 3       |           |

**b) Non-repeated Item (occ=1)**

For non-repeated items, just set question under the corresponding criteria. If the holder is a pure livestock holder, then the question “what is the size of your livestock”.

what was the size of your livestock last year?

|   | Condition |
|---|-----------|
| 1 | THOLD=2   |


Otherwise (Crop only holder) then the question will be “how much Kg of %CROPN% you have produced last year”.

how much Kg of %CROPN% had you produced last year?

|   | Condition |
|---|-----------|
| 1 | THOLD=2   |
| 2 |           |

## ***B. Create Helps for Fields***

In addition to question text, a CAPI application can have help information for each question. During entry, the F2 key is used to display a question's help text.

1. From the View menu, press CAPI Questions (or press Ctrl+Q). The question editor will be displayed.
2. Select an item from the tree on the left. Any text already defined for this item will be displayed in the editing window on the right.
3. Switch from editing questions to editing helps by pressing the  button, or by choosing Help Text from the CAPI Options menu.
4. Enter help text into the editing window. You can format the text using the toolbar, or use the formatting choices shown in the CAPI Options menu.

## ***C. Add Pictures***

“Pictures and other graphics can be added to your question text. The picture or graphic must in a bitmap (\*.bmp) file. There are two ways to do this:

- Copy and paste them directly into the question window, or
- Select Insert Bitmap from the CAPI Options menu.

Pictures are limited to 512K bytes in size. You can determine a picture's size by right-clicking on it in the Windows Explorer and selecting Properties. The size, in bytes, will be displayed in the properties window. Bitmaps with 24 bit format take up the most space. Bitmaps with only 16 colors take up the least space.

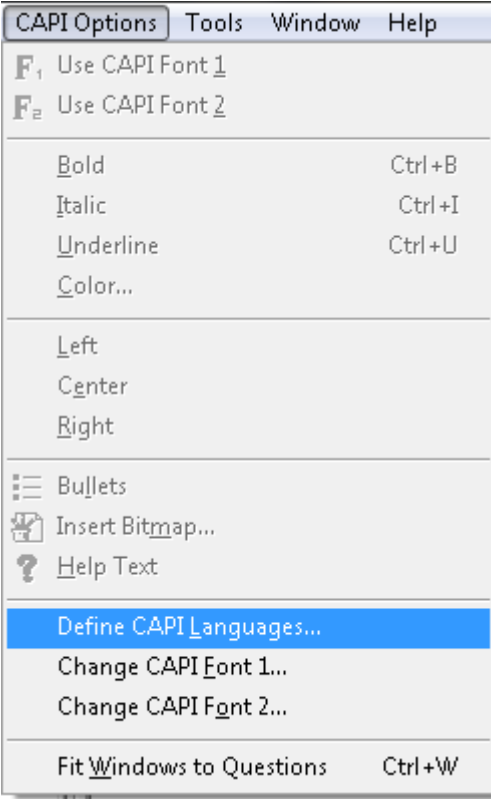
Notes: You can also add bitmaps by right-clicking inside the question window, and choosing the Insert Bitmap option.” (CSPRO User's Guide)

## **III. Add multi language capi question**

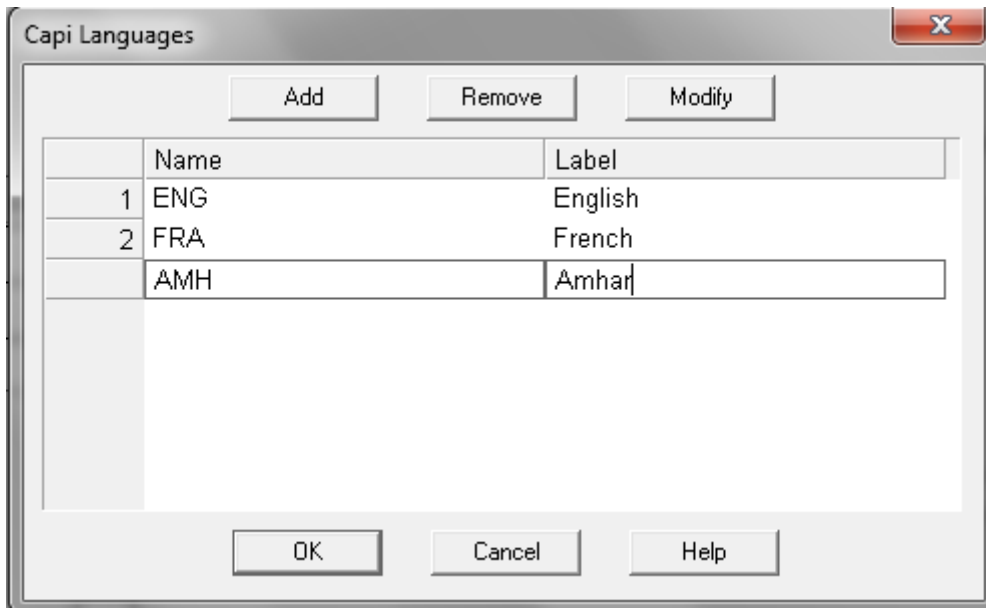


A CAPI application can contain question text in multiple languages. The interviewer can switch among available languages as needed.

1. From the CAPI Options menu, select Change CAPI Languages. The CAPI Languages dialog box will be displayed, showing the languages currently defined for the application.



2. You can add, remove, or modify the your application's CAPI languages

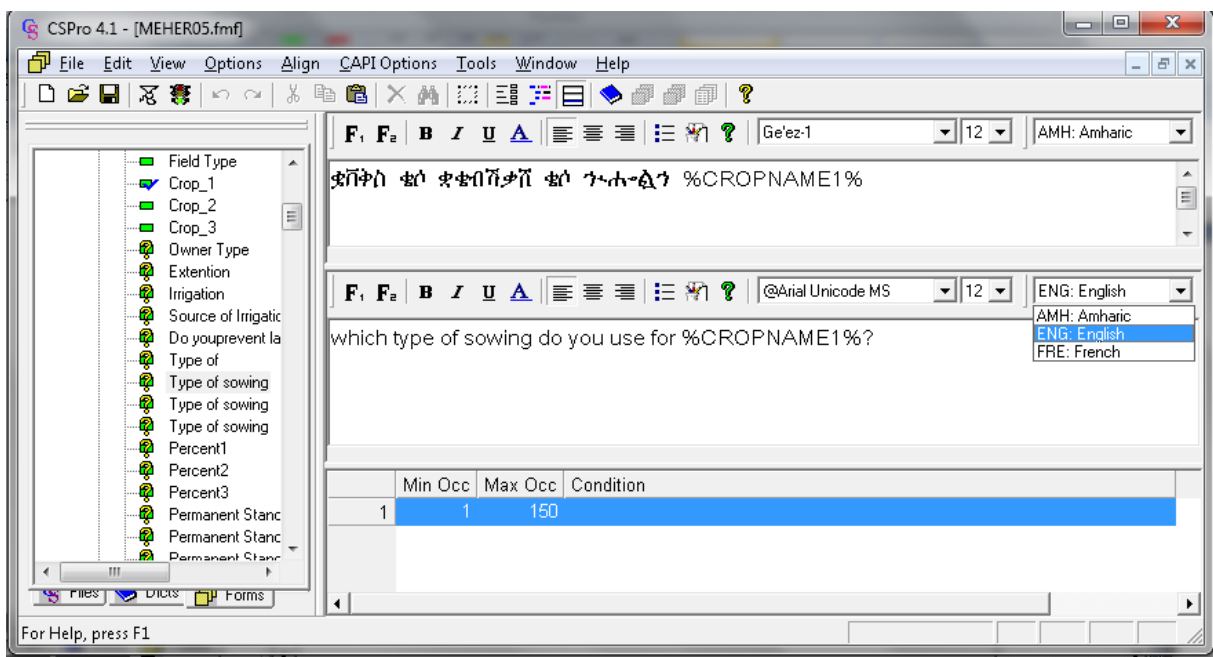


**Notes:**

- Language names follow the same rules as names; that is, they must be unique and cannot contain spaces. Try using abbreviations like ENG (English), SPA (Spanish), FRA (French), or POR (Portuguese).
- Language labels can contain any text to describe the language.
- During data entry, the interviewer can easily switch among languages

After adding a new language, you can now write the question in it:

- From the “Capi Question Editor” tools, use the “down arrows” to scroll the list of available languages (those which have been already defined);
- Write the question, instruction or help text



To manage with a CAPI Multilanguage, the following functions can be useful: getlanguage and setlanguage

**Example1 :**

```
PROC GLOBAL
```

```
numeric I;
```

```
PROC MEHER05_FF
```

```
preproc
```

```
  I = 0;
```

```
  do until I in 1:3
```

```
    I = accept("interview language", "AMHARIC", "ENGLIST", "FRENCH");
```

```
  enddo;
```

```
if I = 1 then
```

```
  setlanguage("AMH");
```

```
elseif I = 2 then
```

```
  setlanguage("ENG");
```

```
else setlanguage("FRE");
```

```
endif;
```

**Example 2 :**

```
  if getlanguage() = "ENG" then
```

```
    errmsg("Thank you");
```

```
  elseif getlanguage() = "AMH" then
```

```
    errmsg("ሰባቱቶቶቶቶቶቶቶቶቶ");
```

```
  endif;
```

NOTE: Amharic can be written in the Capi Question editor, in a form (DE) but not in the logic and in the dictionary

Note: After entering all the CAPI questions (and instructions if any), we are ready to test the first form of the application.

#### IV. Read, Write In an External File

## ***A. Using lookup files***

Lookup files as described in CSPro User's Guide can be used for

- Geographic codes and names. Your application could show the name corresponding to the code the user keyed.
- Industry and occupation codes. Your application could make sure the user keys a valid code. It could also display the names associated with the code.
- Last year's data. Your application could look up a corresponding field from last year's data and calculate a percentage change.
- Dynamically create value sets according to the input of the interviewer
- Generalized menu choices. Your application could read a lookup file and show the contents on the screen as a menu, then convert the user's choice to a code.

“To use lookup files, we first need to create an extra data dictionary and associate this with the file to use as a lookup file. Then we create the main dictionary, and connect the relevant items in it with the items of the lookup dictionary.” (CSPro User's Guide)

Example of Geographic codes:

For conducting the MEHER agricultural survey in Ethiopia, it is important to fill informations about geographic Ids: region, zone, EA and village.


We would like to ensure that the following combination (region, zone, AE and village) exists. In other words, we must ensure that the village does indeed belong to EA and the EA in turn belong to the ZONE which in turn belongs to the Region.

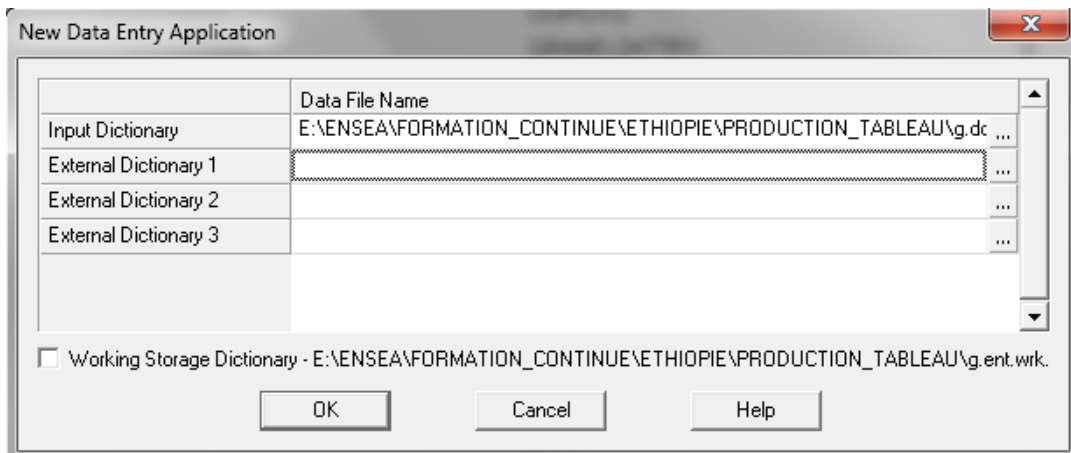
To do this, a lookup file should be created. A lookup file is a file containing the concatenated codes of region, zone, AE and village (according to the administrative division).

Implementation

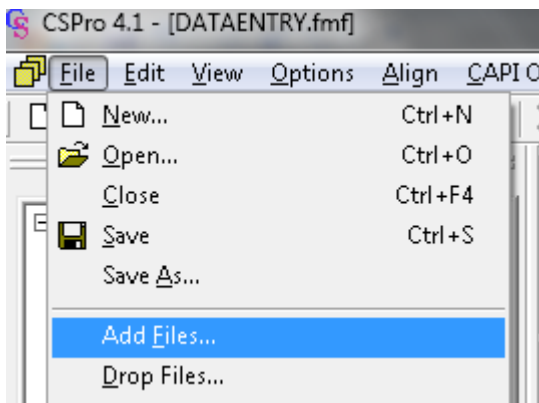
- We assume that the main dictionary (called MEHER05) contains the following IDs variables: region, zone, AE and village
- We assume that an external dictionary called AREA, to use with the lookup file has been created. And the following variables have to be declared in the dictionary: Lregion, Lzone, LAE and Lvillage

To create a DE application containing an external dictionary (file), you can proceed in two ways:

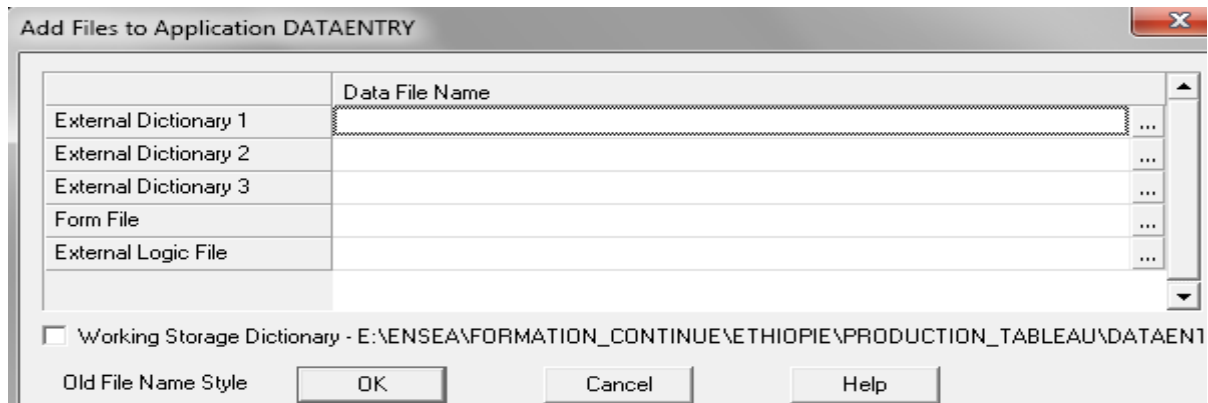
- Way1: Click  on the toolbar or from the File menu, select New. And follow the instruction as describe in the module 1 of the manual. but in the window below, after selecting the Input Dictionary, Click on the button in front of “External Dictionary 1” and select the external one (AREA);



- Way2: After creating the data entry application, add the external dictionary after creating the Data Entry application.



Click on the button in front of “External Dictionary 1” and select the external one (AREA);



The following logic have to be written as postproc on «village ITEM» :

```
PROC VILLAGE
E_REGION = REGION;
E_ZONE = ZONE;
E_AE=AE
E_VILLAGE=VILLAGE
```

```
read = loadcase (AREA_DICT, E_REGION, E_ZONE, E_AE, E_VILLAGE);
if read <> 1 then //record is not found
errmsg(" Could not find region=%d, Zone=%d, AE=%d, Village=%d", region, Zone, AE,
Village)
reenter;
endif
```

## ***B. Write in an external file***

To write in an external file, you need to create an external dictionary and link it to the main data entry form as shown above.

To write the IDs Items in the external file, the following logic have to be written as postproc on «village ITEM».

```
PROC VILLAGE
postproc
Clear (AREA_DICT);
E_REGION = REGION;
E_ZONE = ZONE;
E_AE=AE
```

E\_VILLAGE=VILLAGE

writecase (AREA\_DICT, E\_REGION, E\_ZONE, E\_AE, E\_VILLAGE);

### ***C. Write in an external form***

We assume that we are conducting a household survey. And there are two questionnaires:

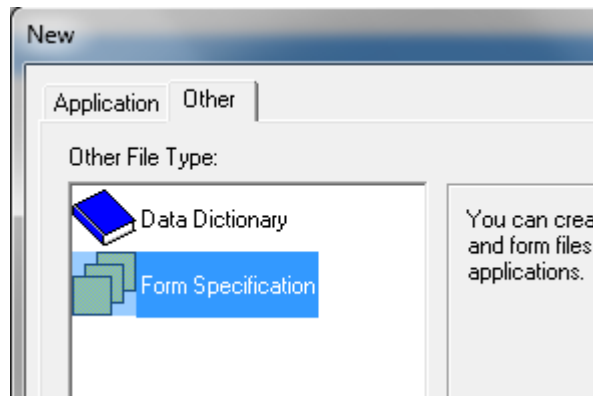
- Household questionnaire
- Reproductive health questionnaire (concerns only women aged 18 - 45 years)

During the interview of household members, Reproductive health questionnaire will be addressed to women aged 18- 45 years.

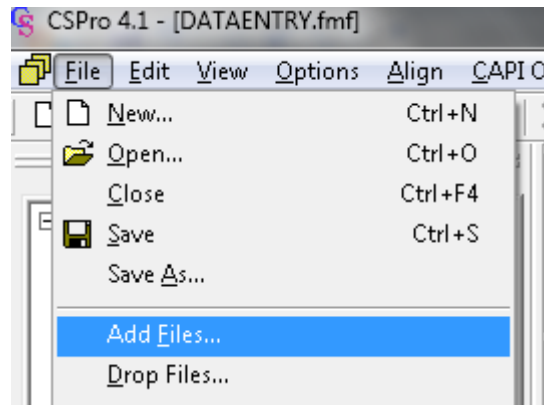
Reproductive health form can be linked to the Household Data Entry Application as an external form.

#### Implementation

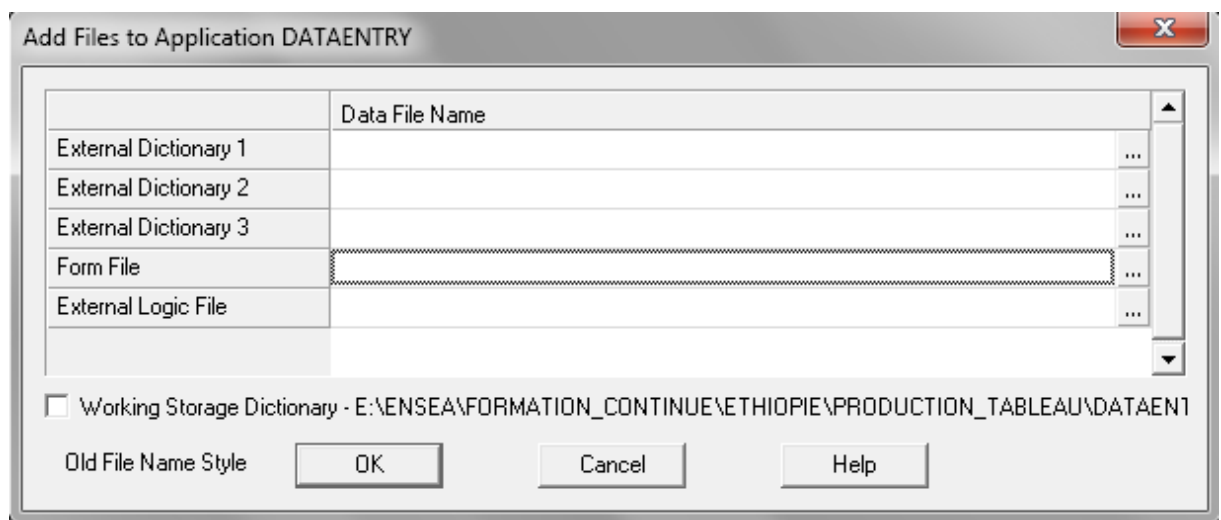
1. Design the Data Entry application for Household questionnaire.
2. Create the data dictionary corresponding to the Reproductive health questionnaire and generate the form specification



3. Add the external form as shown below.



Click on the button in front of “Form file” and select the external one (Reproductive\_Health)



The following logic have to be written.

```

PROG AGE
postproc
if gender=2 and $ in 15 :45 then
clear(REPROD_HEALTH_DICT);
RPH_REGION = REGION;
RPH_ZONE = ZONE;
RPH_AE=AE
RPH_VILLAGE=VILLAGE
RPH_HHID= HHID
ENTER REPROD_HEALTH_FF;
  
```

If the respondent is a woman and his age is between 15 and 45 years old, then the cursor will move from the household DE to the reproductive health DE.



## V. Example of logic

### *Dynamic valueset change*

```
PROC CITYTOWN
preproc
set attributes(CITYTOWN), assisted on;
if REGION=1 then
    setvalueset(CITYTOWN, CITYTOWN_VS1);
elseif REGION=2 then
    setvalueset(CITYTOWN, CITYTOWN_VS2);
elseif REGION=3 then
    setvalueset(CITYTOWN, CITYTOWN_VS3);
elseif REGION=4 then
    setvalueset(CITYTOWN, CITYTOWN_VS4);
endif;
```

*Change language in CAPI : it is possible to switch among multiple language using the logic below*

#### **Example 1 :**

**« this statement will help you to select the language of interview at the beginning of the data keying »**

```
PROC GLOBAL
numeric I;
PROC MEHER05_FF
preproc
    I = 0;
    do until I in 1:3
        I = accept("interview language", "AMHARIC", "ENGLIST", "FRENCH");
    enddo;
    if I = 1 then
setlanguage("AMH");
elseif I = 2 then
setlanguage("ENG");
else setlanguage("FRE");
endif;
```

#### **Example 2 :**

**«This statement will help you to select the appropriate error message according to the language of interview»**

```
if getlanguage() = "ENG" then
    errmsg("Thank you");
elseif getlanguage() = "AMH" then
    errmsg("ሰላም ላይ ስለሆነ ደብዳቤ ይጻፉ");
endif;
```

### ***Demonstrate visualvalue***

```
PROC SEX {DEMONSTRATE VISUALVALUE}
preproc
{Example of VISUALVALUE}
if VISUALVALUE(SEX(curocc())) <> notappl then
  display ("Skipping sex because value already there");
  skip to AGE;
endif.
```

### ***Demonstrate put/editnote***

```
postproc
{demonstrate put/editnote}
{example of putnote then editnote [display note].}
if sex = 2 then
  putnote ("check this lady out!");
  cnote = editnote ();
endif;
```

### ***Combine "execsystem" and "Onkey" functions for starting another windows application or process***

```
PROC GLOBAL
```

```
function OnKey(x)
if x = 113 then {after typing F2}
execsystem("C:\ProgramFiles(x86)\MicrosoftOffice\Office12\WINWORD.EXE"
"E:\FORMATION_CONTINUE\ETHIOPIE\aid.docx"); {open a word file}
onkey = 0;
elseif x=114 then {after typing F3}
execsystem("C:\Program Files (x86)\Microsoft Office\Office12\EXCEL.EXE"); {open excel
spreadsheet}
onkey = 0;
elseif x=115 then {after typing F4}
execsystem("C:\ProgramFiles(x86)\VideoLAN\VLC\vlc.exe"
"E:\ENSEA\FORMATION_CONTINUE\ETHIOPIE\tools\prog\prog3\MOV02616.avi"); {open
MOV02616.avi }
onkey=0
else
OnKey = x;
Endif;
end;
```

## VI. REFERENCES:

- [1] «La conception et la mise en œuvre des enquêtes 1-2-3 en UEMOA », les enseignements méthodologiques STATECO N°99, 2005
- [2] Anne Abelsæth (2012) “Tutorial: Development of data entry and CAPI applications in CSPro”
- [3] Center for Disease Control and Prevention. EpiInfo, Version 6.  
<http://www.cdc.gov/epiinfo/epi6man/epi6titl.html>
- [4] CSPro user’s Guide, Version 4.1
- [5] Guide Pratique BIT : « Traitement des données et stockage des fichiers électroniques concernant une enquête sur le travail des enfants », Genève, Version révisée en décembre 2003.
- [6] Insee Résultats : «Recensement général de la population et des logements du Territoire des îles de Wallis et Futuna 22 juillet 2003», n°47 Décembre 2005
- [7] MICS4 Survey Design Workshop (2010): “Use of PDAs in MICS”
- [8] U.S. Census Bureau, CSPro. <http://www.census.gov/ipc/www/cspro/index.html>

