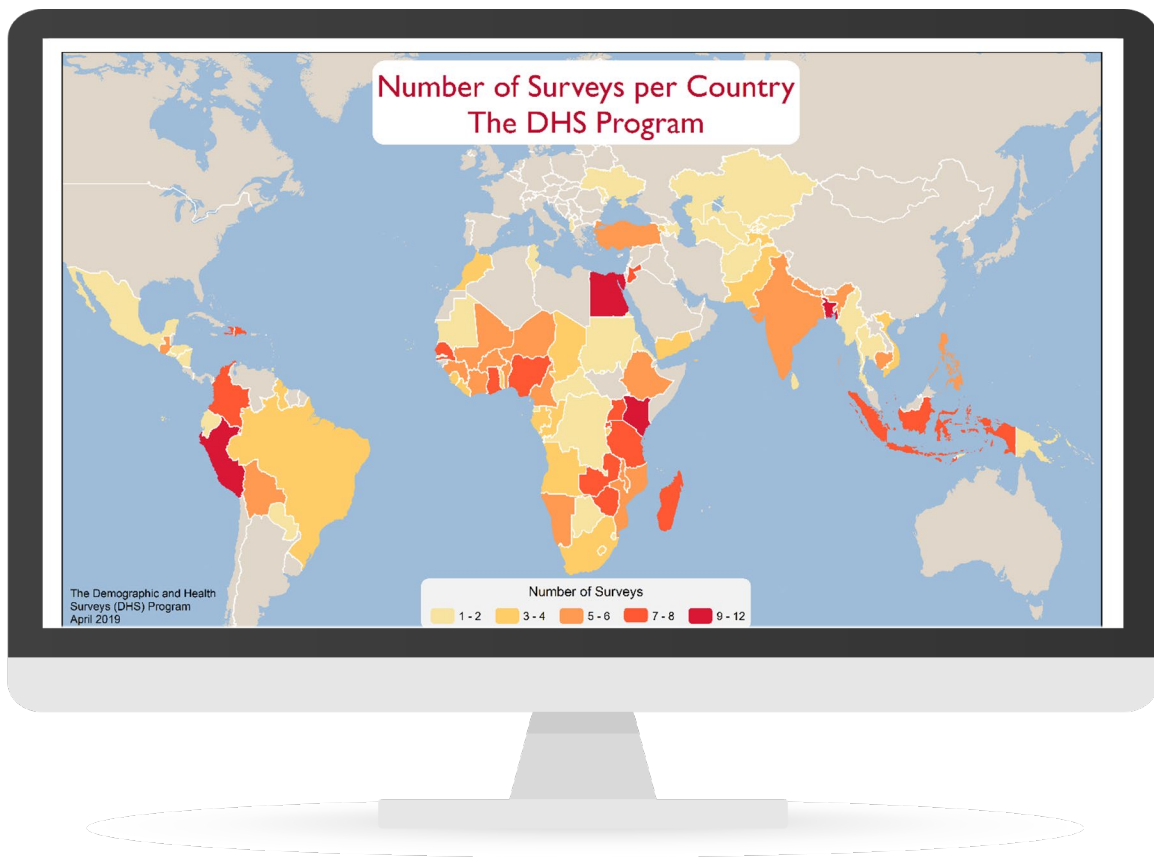


DHS Program Data Analysis Workshop Stata Exercises (version 1.0)



This workbook belongs to:



CONTENTS

INTRODUCTION 1

WORKSHOP OBJECTIVES 1

HOW TO USE THIS WORKBOOK 2

USEFUL STATA COMMANDS TO REMEMBER 5

EXERCISE 1: DHS SURVEY QUESTIONNAIRES 7

EXERCISE 2: WORK WITH STATA .do FILE 12

EXERCISE 3: USE WEIGHTS IN STATA 26

EXERCISE 4: RECODING EXAMPLE 1 37

EXERCISE 5: RECODING EXAMPLE 2 47

EXERCISE 6: SETTING THE SURVEY DESIGN AND USING SVY COMMANDS 53

EXERCISE 7: EXPORT FREQUENCY TABULATIONS TO EXCEL 57

EXERCISE 8: CROSS TABULATIONS 62

EXERCISE 9: TABULATIONS FOR A SUB-POPULATION 68

EXERCISE 10: MERGE DHS DATA FILES 76

EXERCISE 11: LOGISTIC REGRESSION 83

ANSWER KEYS 91

INTRODUCTION

[UPDATE THIS SECTION TO YOUR SPECIFIC WORKSHOP]

The purpose of this workshop is to strengthen the capacity of the workshop participants to analyze and conduct research using population-based survey data. One aim of analyzing Demographic and Health Surveys (DHS) Program data would be to help carry out evidence-based research regarding issues such as sexual and reproductive health, infant and child health and mortality, maternal health, HIV, nutrition, and other health domains. By strengthening one's skills, participants would also be able to integrate DHS survey data into their teaching and/or research as well as strengthen their ability to share what they have learned about DHS survey data at their home institutions.

The statistical software Stata will be provided for use at the start of the workshop. The content of the workshop includes DHS questionnaires, data files, use of .do files in Stata, recoding of variables, survey sampling and weighting, computation of indicators, dataset merging, regression analysis, and data use for decision making.

This workbook contains practical exercises to enable participants to apply the skills they will acquire during the workshop. It contains 10 exercises that target the competencies you need to be able to analyze DHS survey data.

WORKSHOP OBJECTIVES

[UPDATE THIS SECTION TO YOUR SPECIFIC WORKSHOP]

By the end of this analysis workshop, participants will be able to:

1. Gain a thorough understanding of DHS surveys, and their corresponding data files and variables
2. Learn about DHS survey standards for organizing and storing data
3. Understand DHS survey sample design, including weights and how to use them
4. Use Stata to open DHS survey datasets as well as find and recode variables, run frequencies, and carry out crosstabulations
5. Merge different data files to meet analysis needs
6. Understand and account for complex survey design
7. Run and interpret regression results
8. Use DHS survey data to answer research questions focused on topics including, sexual and reproductive health, maternal and child health, nutrition, and HIV
9. Create a conceptual framework based on the group research question, identify variables of interest, develop an analytical method, and conduct a preliminary analysis

HOW TO USE THIS WORKBOOK

It is recommended that you go in order, but it is not mandatory. Feel free to jump around a bit too if you prefer! What is important is that you practice, test out concepts in Stata, and reinforce what you have learned through **MORE PRACTICE**.

Note that the estimated time to complete each exercise is 30 minutes to 1 hour, depending on your background knowledge, and the pace at which you learn best.

When the font changes in the book, take note! For example, you will see instructions like these:

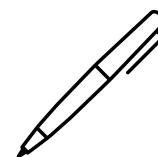
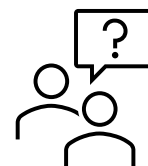
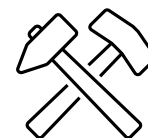
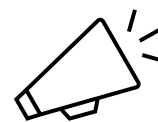
1. “When opening files in Stata, make it standard practice to use the `cd` and `use` commands.” Or this: “use `ZZHR62FL.DTA`, `clear`” The text in blue refers to syntax that should be used (and modified) in Stata.
2. Be careful though! Sometimes you will be able to copy/paste the syntax into your `.do` file; other times, you will need to modify the syntax. If you see font that is italicized (e.g., tab `var1 var2`), you will need to adapt this code with specific variables.

For each exercise, you will see a Toolbox that include Stata commands that will be highlighted. Make use of this Toolbox as these are core concepts to learn in Stata.

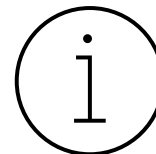
When in doubt, type the `help` or the `lookfor` commands in Stata!

1. Not sure what something in your toolbox means? Type `help tabulate`, for example. Stata provides a wealth of resources on the command’s utility.
2. Get comfortable using the `lookfor` command to help you find a variable of interest.

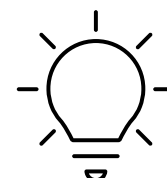
As you go through the workbook, feel free to jot down notes or questions throughout. For longer thoughts, there is a notes section at the end of each exercise where you can jot down main points you want to remember, what you found challenging about the exercise, or things you would like to go back to later and check. You can also use this space as a “parking lot” to jot down questions you want to ask the workshop facilitators when you get a chance. Use this space in a way that is most useful to you!



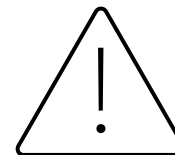
Throughout the workbook, information such as definitions and explanations of some of the concepts you will be working on in the exercises are shared. You will see the information icon to indicate a definition or explanation.



To help you work through the exercises and gain a stronger command of Stata, tips pertaining to the different exercises are included throughout the workbook.



In some cases, important information that requires extra attention is pointed out throughout the workbook.



Some of the exercises contain questions you will have to answer as you go through the exercise. All of the answers are located in the answer key at the end this workbook. Try not to peek before you answer the questions!



USEFUL STATA COMMANDS TO REMEMBER

Please use the help command for more details about the commands below.

cd	changes your working directory. See help cd
clear	drops all data and label values from the Stata memory. See help clear
describe	provides information about the variable (e.g. variable name, variable type). See help describe
use	specifies the specific dataset you would like to use. See help use
drop	removes variable or observation from the dataset. See help drop
generate or gen	helps to create a new variable. See help gen
egen	provides extensions to the generate command. See help egen
[iweight=X]	accounts for weighting by the variable X. See help weights
lookfor	helps to find variables based on variable names and labels in dataset. See help lookfor
numlabel, add	adds the number labels to the categories of a variable. See help numlabel
recode	helps with changing the original values of a variable. See help recode
save	saves a dataset to a specified location save <i>"filename.DTA", replace</i> . See help save
set more off	tells Stata not to pause or display the --more-- message. See help "set more"
set maxvar	increases the maximum number of observations. See help "set maxvar"
summarize or sum	gives a summary of the variable. See help sum
svyset	used to set up the complex survey design. See help svyset
svy: tab var	tabulates the variable <i>var</i> (must run svyset code first). See help "svy: tab"
tabout	exports frequencies and percentages of your variables to excel. See help tabout

Options for two-way tabulate commands

Tab var1 var2, option

Cell cell %

Col column %

row row %

Options for svy: tab commands

Svy: tab var1 var2, option

cell cell %

ci confidence interval

col column %

count counts for each cell

per percentage

row row %

se standard error

Logical Operators

Relational Operators

& AND > Greater than <= Less or equal

| OR < Less than == Equal

! NOT >= Greater or equal != Not equal

EXERCISE 1: DHS SURVEY QUESTIONNAIRES

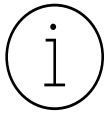
The purpose of this exercise is to introduce the DHS survey questionnaires. Since the data you will be analyzing is generated from these questionnaires, it is very important that you familiarize yourself with their content. For the purpose of this exercise, we will use the DHS6 model questionnaires corresponding to the DHS model datasets.

After completing this exercise, you should be able to:

1. Describe the content of the DHS survey questionnaires

You will be using **DHS model datasets** to carry out the analysis exercises in this book.

What are DHS model datasets and how were they created?



DHS model datasets provide opportunities to practice, manipulate, and analyze imaginary data first before analyzing or teaching while using actual country data. To reiterate, *DHS model datasets include imaginary data*. These resources are used for educational and teaching purposes.

Created by The DHS Program, DHS model datasets originated from several different DHS Phase-6 surveys and their corresponding recode files. For instance, data were manipulated in the following ways:

1. Cluster and household numbers were randomized
2. New clusters were assigned to different regions and urban/rural areas
3. New weights were produced and assigned to clusters
4. Standard variables not found in the data subset were imputed
5. And much more!

To access DHS model datasets, visit The DHS Program's website: www.dhsprogram.com > data > download model datasets

These files and resources can be downloaded for free without registering.

Once you gain a solid understanding of the following:

6. how DHS questionnaires work;
7. what kind of information is collected; and,
8. how to manipulate and analyze data files;



Before you Begin

1. Access the link for model datasets as explained above.
2. Access the DHS-6 Questionnaires on The DHS Program's website: www.dhsprogram.com > methodology > questionnaires and manuals > DHS model questionnaire – phase 6 (2008 – 2013) (English, French) or by scanning the QR code below.



3. Once you have found the questionnaires in PDF format, familiarize yourself with the content and structure of the questionnaires.
4. Work with another workshop participant to answer the following questions. Include the relevant page numbers and/or question numbers from the questionnaire to support your answers.

A- Answer the following questions:

1. Which questionnaire collects information on employment and gender roles (for example, who makes decisions regarding the household)?

2. Which questionnaire is used to ask parent/caregivers to provide consent for anemia and malaria tests in children?

3. What kind of data is collected on the topic of health insurance? What are some questions that are asked? Which respondents provide answers to these questions?

4. What are the possible response categories for the person who carried out a male circumcision?
What are the different response options for where the circumcision was done?

B- True or False (if false, make the statement true)

5. Antenatal care information is collected for only the last pregnancy that ended in a live birth in the last 5 years.

True

False _____

6. DHS surveys collect information from the respondent on her partner's opinion in terms of where she gives birth.

True

False _____

7. Men are asked to provide a full birth history of all births that have occurred (alive or not)

True

False _____

8. Hemoglobin tests are offered for all children in the household aged 0-59 months.

True

False _____

9. Information on children's history and treatment of fever is collected from mothers and fathers in standard DHS surveys.

True

False _____

10. DHS surveys collect information on the times that the respondent and her partner have used a method to avoid getting pregnant.

True

False

NOTES ON EXERCISE 1

EXERCISE 2: WORK WITH STATA .do FILE

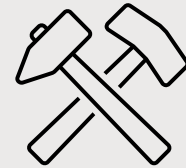
The purpose of this exercise is to explain DHS survey datasets in Stata.

After completing this exercise, you should be able to:

1. Create and save a Stata .do file
2. Open a DHS model dataset in Stata
3. Search for variables
4. Run a frequency of urban/rural residence
5. Examine differences in frequencies by data files

Stata command toolbox:

cd	changes the working directory
use	specifies the specific dataset you would like to use
clear	drops all data and label values from the Stata memory
lookfor	helps find variables of interest in the dataset



Before you Begin

1. Make sure you have the required datasets downloaded, unzipped, and stored in a location on your hard drive computer. First, create a folder called [Workshop Name] and save all datasets in a “Data” sub-folder. You will need the following datasets:

ZZHR62DT.DTA

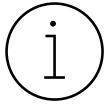
ZZIR62DT.DTA

ZZPR62DT.DTA

ZZKR62DT.DTA

2. In addition to the different model dataset files, download and unzip the file `zzfulltables.zip`, which will allow you to review and compare your findings against report tables.

What is a .do file?



A .do file is a text file that allows you to create and save a running record of the syntax and corresponding notes. You will have the advantage of doing a line-by-line run-through of your .do file or executing the entire .do file all at once! And these are just some of its capabilities. Overall, this is a very efficient way to manage and analyze data using Stata.

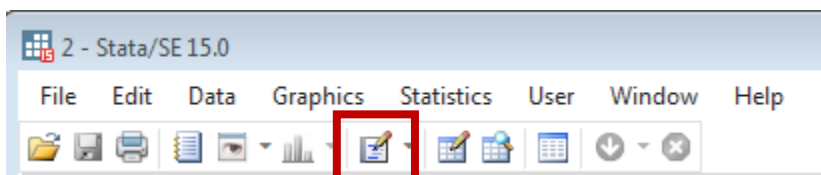
IMPORTANT!



Note that the environment in Stata may look slightly different depending on the version you are using for this workshop!

Beginning a Stata .do file

3. To create a new Stata .do file, open your Stata software and click on the “New Do file Editor” icon as shown in the box below.
4. Start by opening a dataset. There are two ways to open datasets, either by using a dropdown or via a do file. For this workshop, we will aim to **open all datasets via syntax/do-file** so we can have a record of our analysis.



When opening files in Stata, make it standard practice to use the **cd** and **use** commands.

The **cd** command stands for “change directory,” which directs Stata to the folder containing your data files of interest. This file path will be different for each person. Also, **cd** helps you easily change or save data files in a central location without having to always redirect the file path.

Once you have directed Stata to where your files are using the **cd** command, you can type the **use** command, which specifies the data file you would like to use.

Think of the **cd** and **use** commands as a basket of apples. The **cd** command tells Stata the location of your basket holding the dataset(s). The **use** command tells Stata which apple to pick up.



Tips for good .do files:

1. Maintain good Stata hygiene and ALWAYS start a .do file and work from there. Avoid using the Command window if you can help it.
2. Save your .do file as soon as you create one:
 1. Save your file with an intuitive name (for example “practice_dofile”)
 2. Save your do file in the same folder where your datasets are located, and make it a habit to revisit it so you can continue to use it throughout the workshop
3. Always describe the purpose and date of your .do file as a comment/green thinking when at the beginning of the do-file.
 1. You can make comments on a single line by starting the line with an asterisk (* (asterisk), which is the star above the number “8” letter 8 on the keyboard.
 2. You can also make multiple lines of comments/green thinking by surrounding the text with /* and closing with */

When first starting Stata you will see the following screen

Review of previously executed commands →

Space to enter commands →

Data viewer- shows commands as well as output

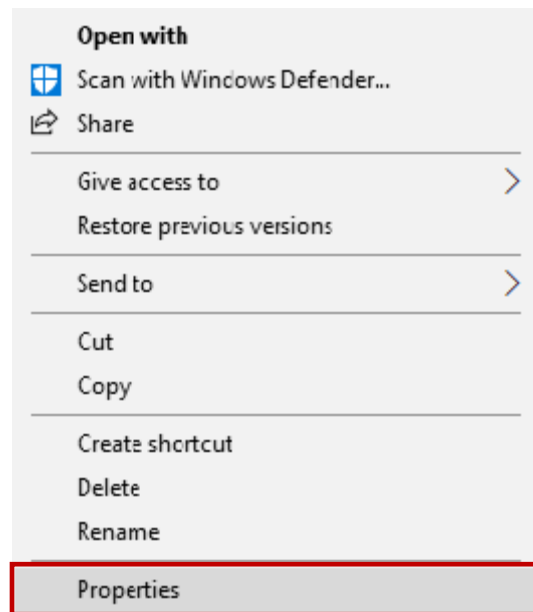
Variable list ←

Variable properties ←

Step 1: Open a .do file, then locate and open the household recode dataset

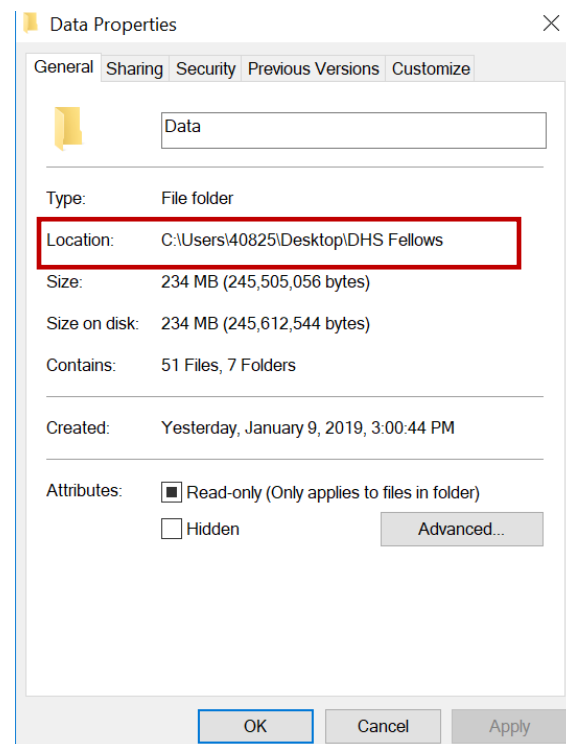
- I. To open a dataset, you must first use the `cd` command and then copy the path of the folder where your data is located. To do this, go to the folder where you saved your data and select one of the data files. Then, right click and select “Properties” from the menu. The path will be identified by the location in the box as shown below.

 ZZHR62FL	1/9/2019 3:07 PM	DTA File
 ZZIR62FL	1/9/2019 3:10 PM	DTA File
 ZZKR62FL	1/9/2019 3:13 PM	DO File



The path for where the data is saved is highlighted in the red box. In this example, the path name is:

C:\Users\40825\Desktop\DHS Fellows



Each computer has a unique path name. In your Stata .do file, type the `cd` command followed by your pasted path name in quotation marks.

```
cd "C:\Users\40825\Desktop\DHS Fellows\Data"
```

The command `cd` tells Stata to change the directory to the folder indicated by the path provided.

2. Now we can open the household recode dataset using the `use` command:

```
use ZZHR62FL.DTA, clear
```

IMPORTANT!



When using a .do file, it is always recommended to add comments in terms of what you aim to look for as well as what you have found based on the syntax that you have run. It is helpful to write these notes down to remember your process of thinking. This is also important if you are sharing code with your colleagues who may plan to collaborate on the data management and analysis with you.

For example, it is good practice to write out at the top of the .do file what your purpose is of the file as well as the data in the form comments.

What do we mean by comments in a .do file?

Stata comments start with an asterisk (*), which can be found above the number 8 on your keyboard. So, before you begin running any commands (the `cd` and `use` commands as discussed above), make comments indicating what you are working first. In the do-file itself, when you add an asterisk before the syntax, it turns green. Consider these comments as your “green thinking.”

.do file example

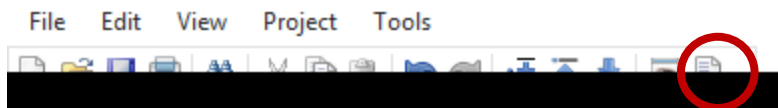
```
*Exercise 1
```

```
*January 10, 2019
```

```
cd "C:\Users\40825\Desktop\DHS Fellows\Data"
```

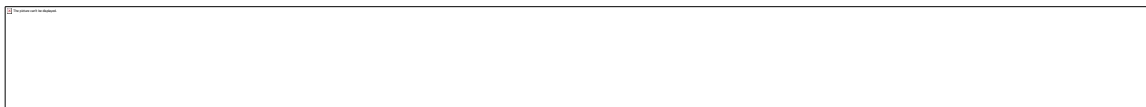
The option `clear` will clear any data you may have opened in Stata before.

3. Highlight the `cd` and `use` command lines in the .do-file and click “Execute (do)” as highlighted below:



1. **Tip:** If you like shortcuts, you can highlight the command lines and use the shortcut (Ctrl+d) to run the line of code

4. Set Stata to handle more variables. When you execute your .do-file, you may get the following alert:



This alert is likely because DHS datasets may be too large with a lot of observations or variables, and therefore cannot be opened as is.

To set Stata to handle more variables, write the following syntax in the command window:

```
set maxvar 10000, perm
```

The option perm is short for the word “permanently.”

This syntax therefore increases your computer’s memory. You only have to do write this syntax out once since Stata *permanently* stores this information. This is an added reason why including this syntax in our .do file (as opposed to in the Command window) is beneficial and more efficient.

5. After you set the memory, re-run the cd and use commands to open the HR data file.

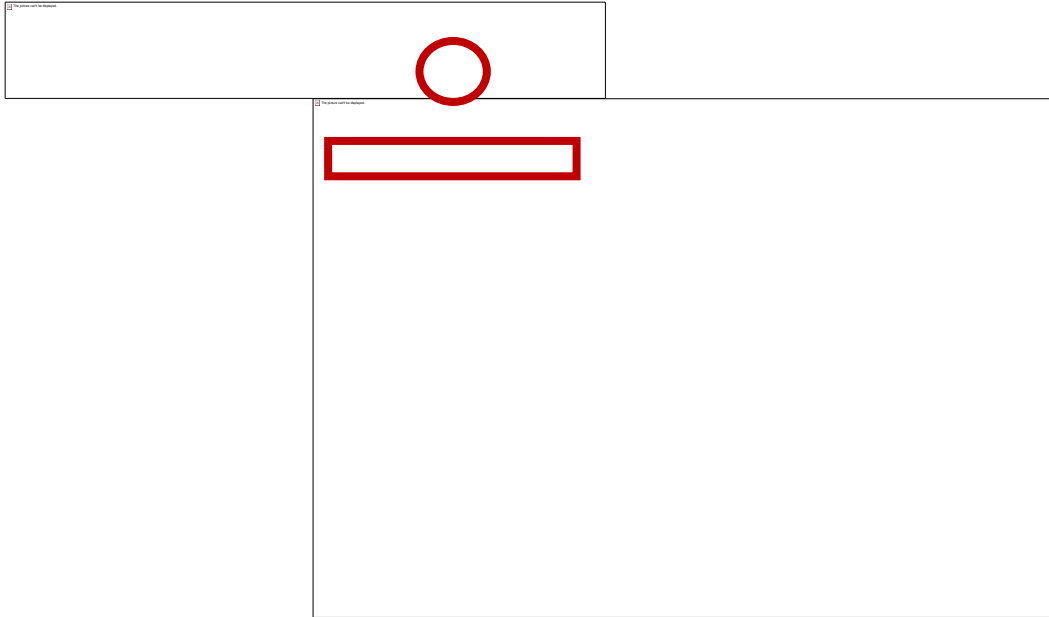
Step 2: Identify the urban/rural variable and run a frequency

1. To get started with exploring the data in the HR data file, let’s start with analyzing the variables for urban and rural residence. So, how do we know which variable is urban/rural residence? There **are two different ways to search for a variable:**

1. Use the “Variables Manager” feature; or,
2. Use the lookfor command.

Both methods will give you the same result when searching for variables.

Option 1 – Variables Manager: To browse the menu of variables included in the dataset, you can click on the button for the “Variables Manager” feature, which is found to the left of the down arrow button. From there, searching for the term “residence” at the top of the window in the box with gray font that says, “Filter variables here.”



Option 2 – lookfor: The second way to find a variable of interest in your dataset is to type the command “lookfor residence” in your .do-file and execute the syntax.



3. The variable we need is **hv025**

To run a frequency of urban/rural residence, we need to use the command tabulate (or tab for short), followed by the name of the variable of interest.

Type in your .do-file tab hv025, then highlight the syntax, and run the command.

.do file example

*Exercise 1

*January 10, 2019

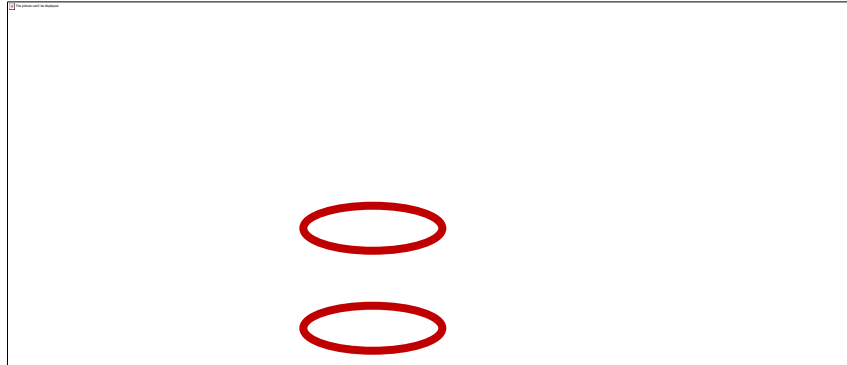
```
cd "C:\Users\40825\Desktop\DHS Fellows\Data"
```

```
use ZZHR62FL.DTA, clear
```

```
*Tabulation for urban/rural residence
```

tab hv025

You should see the following results of the frequency appear in the output window.



The image shows a screenshot of a Stata output window. The window contains a frequency table. Two rows of the table are circled in red. The first circled row shows a total of 6,290 observations. The second circled row shows the distribution of observations by area type: 2,303 in urban areas and 3,987 in rural areas.

	Number of Observations
Total	6,290
Urban	2,303
Rural	3,987

The table indicates there are a total of 6,290 observations that represent households. Out of the sample of 6,290 households, 2,303 are in urban areas and 3,987 are in rural areas.

Step 3: Open the Individual Recode (IR) file and run a frequency of urban/rural residence

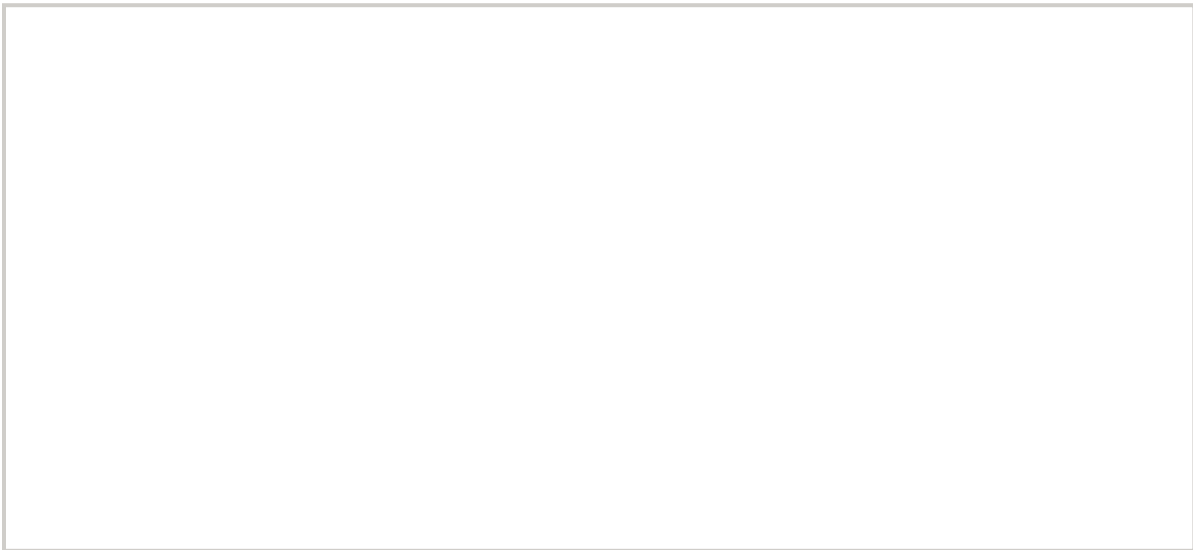
Now, open the individual recode file and run a frequency of urban/rural residence. Remember to type the use command to open a dataset (or pick up the apple in the basket!).

*Open the IR dataset

use ZZIR62FL.DTA, clear

I. Answer the following question:

What is the variable for urban/rural residence in the IR file? Use the variables manager feature or the lookfor command to figure out the correct variable. Check to see if you get the correct answer in the answer box.



2. Note that variable names are sometimes the same in HR and IR files, but HR files have an “h” in front so analysts can distinguish the difference at a glance.

Run a tabulation of the variable for urban/rural residence in the IR dataset.

.do file example

*Exercise 1

*January 10, 2019

```
cd "C:\Users\40825\Desktop\DHS Fellows\Data"
```

```
use ZZHR62FL.DTA, clear
```

*Tabulation for urban/rural residence

```
tab hv025
```

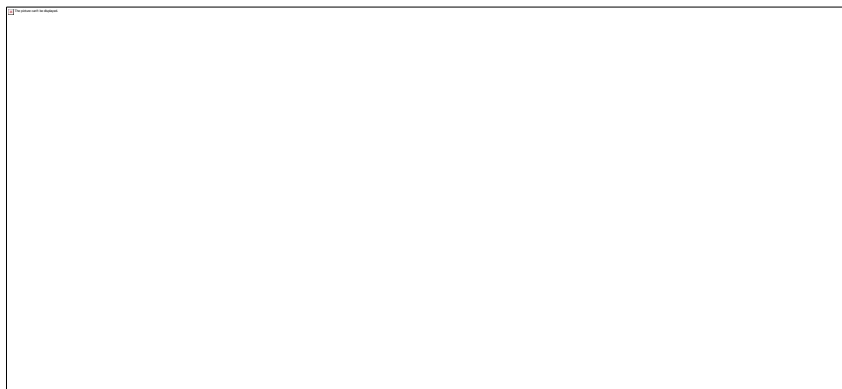
*Open the IR dataset

```
use ZZIR62FL.DTA, clear
```

*Tabulation for urban/rural residence

```
tab v025
```

You should get the following output in Stata:



When looking at the two results from the HR (reminder: this stands for household) and IR (and this stands for individual recode) datasets, why is there a difference in the frequencies?

IMPORTANT!



Remember! Save your .do file suggested name “practice_dofile” _ in the same folder where your datasets are located so that you can use the .do file throughout the analysis workshop.!

Extra Challenge!

Want to practice these important skills a bit more?

Open the PR and KR data files, and then find and run the frequencies of urban/rural, then answer the following questions.

1. What are the number of cases for each file?

2. What do these number represent?

NOTES ON EXERCISE 2

EXERCISE 3: USE WEIGHTS IN STATA

The purpose of this exercise is to practice using weights in Stata.

After completing this exercise, you should be able to:

1. Understand what a sample weight is
2. Identify which weight variable to use
3. Create a weight variable
4. Account for weighting the data in Stata
5. Save your dataset, which includes your new weight variable
6. Match the output to the final report after accounting for weights

Stata command toolbox:

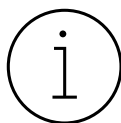
generate or gen	command for creating a new variable
[iweight=var] or [iw=var]	weights data by the variable X
save	save a dataset to a specified location
replace	replaces contents of existing variable; used with generate



Before you begin:

1. Make sure you have the required DHS datasets downloaded, unzipped, and stored in a location on your hard drive computer.
2. For this exercise you will need the following datasets:
ZZHR62FL.DTA
ZZIR62FL.DTA
1. For this exercise you will need to refer to the following tables from the DHS Model Datasets report, which can be downloaded and unzipped (named “zzfulltables.zip”):
Table 3.1 Background characteristics of respondents
2. Please continue to use the workshop .do file created in Exercise 2.
3. Understand what a weight is and which weight to use

What is a weight and how do you determine which weights to use and how you use them?



Weights are used in all analyses to make sample data representative of the entire population. Within the DHS survey, each unit of analysis or sample selection (e.g., households, household members, men, women or children, domestic violence, HIV test results) has its own weight, which needs to be applied to the sample data before carrying out analysis.

Steps for understanding which weights to use and how to use them in Stata:

Step 1: Create the weight variable

1. Before weighting the data, you must create a new variable with the calculated weight. Always remember to divide the sampling weight by 1,000,000 (1 million). DHS data values do not contain decimals. The weight variables are values with 6 decimal places. Therefore, to obtain the actual weight, you need to divide by 1,000,000
2. By convention in this workshop, we will name our new weight variable “wt” but technically, any name you choose can be given.
3. Below is a list of sample weights you will find in the various data files:

Unit of analysis/sample selection	Variable
Households	hv005
Household members	hv005
Women or children	v005
Men	mv005
Domestic Violence	d005
HIV test results	hiv05

Step 2: Apply the weight variable to the tabulation.

1. After creating the new weight (wt) variable, you then must include this variable when running tabulations to apply the

Follow the instructions below to practice these steps.

Step 1: Create the weight variable

1. For this example, open the HR file and practice creating a weight variable. Open the workshop .do file that you created in Exercise 2. Don't forget to use the **cd** and **use** commands to open your datasets. Note: If you have saved your .do file from the previous exercise, you do not need to write the commands again, you can just run the commands.

.do file example

*Exercise 2

```
cd "C:\Users\40825\Desktop\DHS Fellows\Data"
```

```
use ZZHR62FL.DTA, clear
```

2. For the HR file, determine which weight variable you need to use and write it down below.

3. Determine which number you need to divide the variable by and write it below.

4. The command we use in Stata for creating a new variable is called generate or gen for short.

gen new variable = expression

To create a new weight variable, we should type the following syntax into our .do file:

```
gen wt=hv005/1000000
```

Highlight this syntax and run it.

.do file example

*Exercise 3

```
cd "C:\Users\40825\Desktop\DHS Fellows\Data"
```

```
use ZZHR62FL.DTA, clear
```

*Generate the HR weight variable

```
gen wt=hv005/1000000
```

Step 2: Apply the weight variable to the tabulation

1. Include the following syntax in your .do file:

```
tab variable [iweight=wt]
```

or for short:

```
tab variable [iw=wt]
```

Look at the weighted frequency of the region in the HR file. Which variable should you use to examine the region?

2. To tabulate a weighted frequency of region in the HR file, we should use the following command:

*Weighted frequency of region

```
tab hv024 [iweight=wt]
```

You should see the following output in Stata:

.do file example

*Exercise 3


```
cd "C:\Users\40825\Desktop\DHS Fellows\Data"
```

```
use ZZHR62FL.DTA, clear
```

*Generate the HR weight variable

```
gen wt=hv005/1000000
```

*Weighted frequency of region

```
tab hv024 [iweight=wt]
```

IMPORTANT!



Just generating a weight does not apply it to a tabulation. Stata won't know to apply a weight unless you tell it to. One way to account for weighting is to tell Stata to apply a weight through the command `[iweight=weightname]` following your regular command. In this case the *weightname* is "wt" as you can see above when creating a new weight variable.

Step 3: Save your dataset under a new name

It is always important to not only save your .do files but also your datasets. Saving datasets with a different file name prevents you from overriding the original dataset. Since we created a new variable called "wt" included in your .do file, let's save our updated dataset that contains our new variable.

In order to save the updated dataset *without overriding the original dataset*, let's save the dataset by running the command `save`, which should also subsequently be paired with the command `replace`.

The syntax in our .do file will be the following:

```
save "newfilename.dta", replace
```

Since we are creating new variables in the HR file, let's save our dataset with the name `hrvars.dta`. However, you can choose a different name if you like, such as `ZZHRcoded.dta` for instance.

```
save "hrvars.dta", replace
```

.do file example

*Exercise 3

```
cd "C:\Users\40825\Desktop\DHS Fellows\Data"
```

```
use ZZHR62FL.DTA, clear
```

*Generate the HR weight variable

```
gen wt=hv005/1000000
```

*Weighted frequency of region

```
tab hv024 [iweight=wt]
```

*Save file with new variables

```
save hrvars.dta, replace
```

Extra Challenge!

Tabulate percentage of women living in urban/rural residence (weighted and unweighted)

1. Which dataset will you use?

2. Which variable will you use for urban/rural setting?

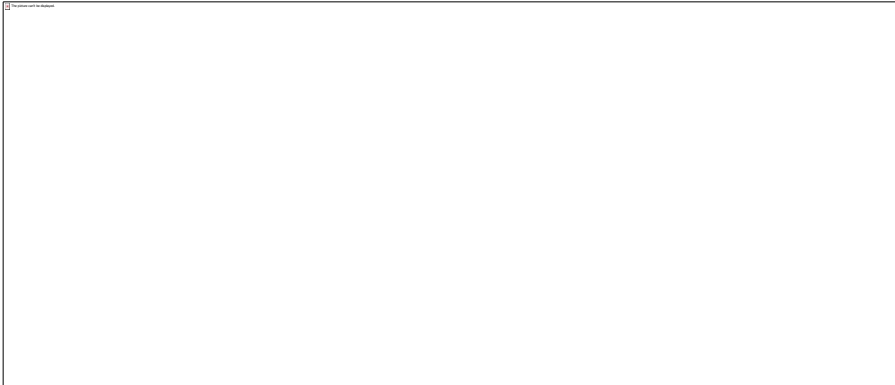
3. Which variable should be used for weighting the data?

Step 4: Match the output to the final report after accounting for weights

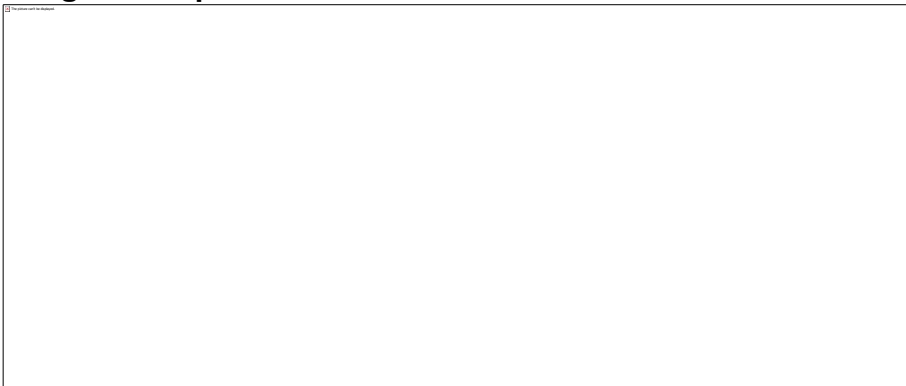
1. Now that you have practiced how to create new variables to account for sampling weights, run weighted and unweighted frequencies by doing the following:
 1. Run an **unweighted** frequency of the urban/rural residence variable
 2. Generate your weight variable (hint: create a new variable called `wt` in your .do file)
 3. Run a **weighted** frequency of the urban/rural residence variable
 4. Compare the weighted and unweighted frequencies using **Table 3.1**.

You should see the following results in Stata:

1. **Unweighted frequencies of urban and rural areas**



2. **Weighted frequencies of urban and rural areas**

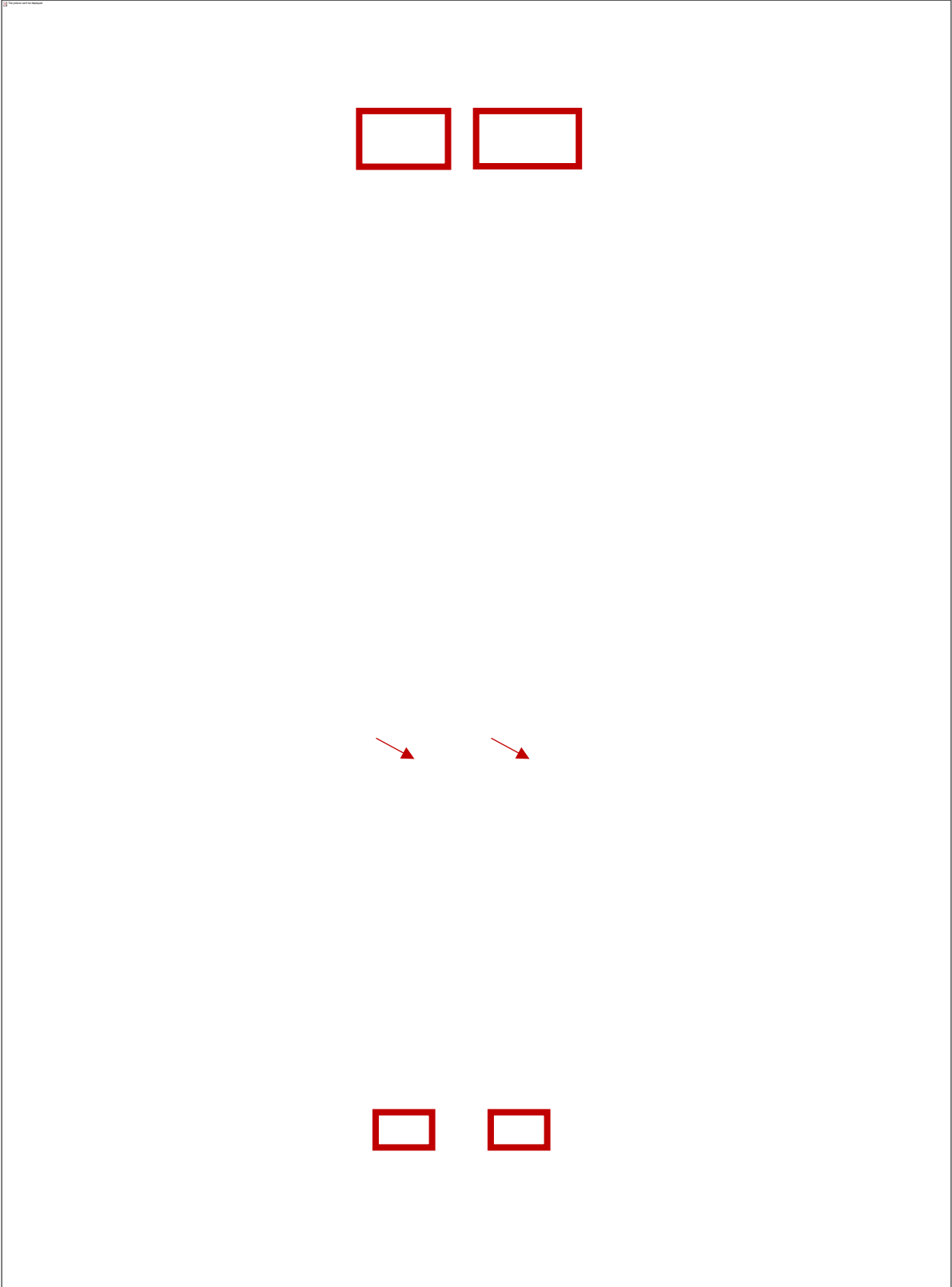


3. What do these results show us? In the unweighted frequency, there are 3,424 urban women in the sample and in the weighted frequency, there are 3,766 urban women in the sample.

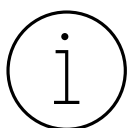
Check to see if your frequencies match Table 3.1.

Also, always check your denominators!

In Table 3.1 below, there are 8,348 women aged 15-49 in the unweighted sample. The SAME number of women are included in the weighted sample aged 15-49.



Normalizing weights



Note that weighted and unweighted **total samples (N)** will always be the same because the weights have been normalized.

Briefly, what does it mean to normalize the weights?

After the weights are initially calculated, they are normalized. Another word for normalized is standardized. Weights are therefore normalized/standardized by dividing each weight by the average of the initial weights (equal to the sum of the initial weight divided by the sum of the number of cases) so that the sum of the normalized/standardized weights equals the sum of the cases over the entire sample. The standardization is done separately for each weight for the entire sample.

On households: The entire set of household sample weights is multiplied by a constant, thus, the total weighted number of households equals the total unweighted number of households at the national level.

On individuals: Individual sample weights are normalized separately for women and men. Thus, the total weighted number of women equals the total unweighted number of women, and the total weighted number of men equals the total unweighted number of men. Women and men are normalized separately because all non-HIV calculations are performed on women and men separately. We do not provide survey estimates on the joint population of women and men combined for anything other than HIV prevalence.

Also, unlike the treatment of weighted and unweighted total samples, weighted and unweighted Ns for **subpopulations** will not always be the same, because they have not been normalized.

Please remember to save your .do file!

NOTES ON EXERCISE 3

EXERCISE 4: RECODING EXAMPLE 1

The purpose of this exercise is to learn how to recode variables in Stata.

After completing this exercise, you should be able to:

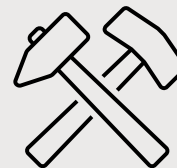
- I. Recode “Which [contraceptive] methods are you using?” (Q307) in the Model Woman’s Questionnaire into categories. You can find the Model Woman’s Questionnaire DHS-6 Questionnaires on The DHS Program’s website: www.dhsprogram.com > methodology > questionnaires and manuals > DHS model questionnaire – phase 6 (2008 – 2013) or by scanning the QR code below.



- I. Properly label new variable

Stata command toolbox:

generate or gen	creates a new variable
replace	replaces contents of existing variable; used with generate
label variable	attaches a label to the dataset in memory
label define	defines value labels
label values	attaches a value label to a variable list
numlabel, add	adds the number labels to the categories of variables



Before you begin

Please continue to use the workshop .do file used in previous exercises.

Challenge!

- I. Let’s recode “Which [contraceptive] methods are you using?” (Q307) in the Model Woman’s Questionnaire into two categories: modern contraceptive use and traditional non-user.

- I. Which data file do you need to use?

2. Which weight do you need to use?

3. Which variable do you need to use?

DON'T CHEAT! You have the tools to look it up in Stata!



Tips for effective recoding:

1. Get to know your data.
 1. Before you start recoding, find, and examine the original variable of interest (e.g., run frequency, summarize the variable).
 2. Check the details. Are some cases missing? If so, why are they missing? Check the skip patterns and always check your total N (sample size).
3. Do not change the original data file!
 1. Remember that when you are creating any new variables (like the **wt** variable that you previously created), it is important to save a new copy of the dataset as well as create a new name for the variable of interest. Avoid writing over the original variable and dataset. Always keep a copy of the original variable.
4. Give the variables that you are creating meaningful and concise names so it makes sense to you and to others who might also work with your dataset:
 1. Do not put spaces in variable name
 2. It is preferable to not use capital letters in variable name (Stata is case sensitive)
 3. Give a descriptive label and values

Step 1: Find the original variable

Let's first open the IR file and find the variable we want to use.

.do file example

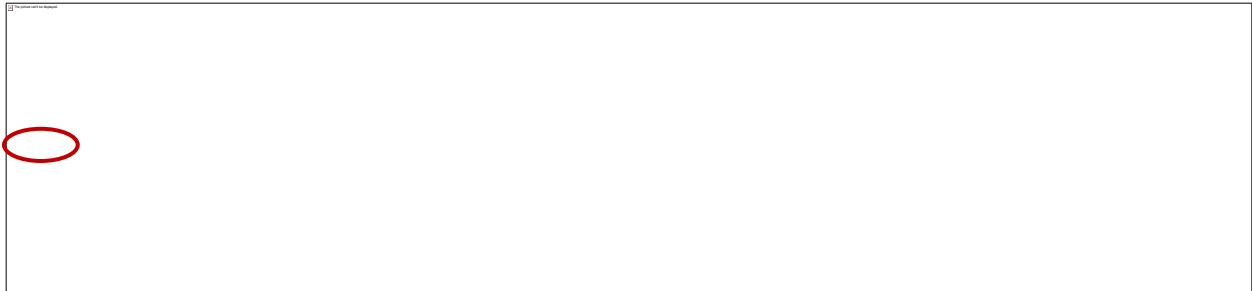
*Recode Example #1

```
cd "C:\Users\40825\Desktop\DHS Fellows\Data"
```

```
use ZZIR62FL.DTA, clear
```

*Find contraception variable

lookfor contra



We are interested in recoding the variable **v312** “current contraceptive method”.

Step 2: Check the original variable

1. Since we know we want to use variable **v312** “current contraceptive method”, tabulate **v312** to see the distribution of the variable.

tab v312

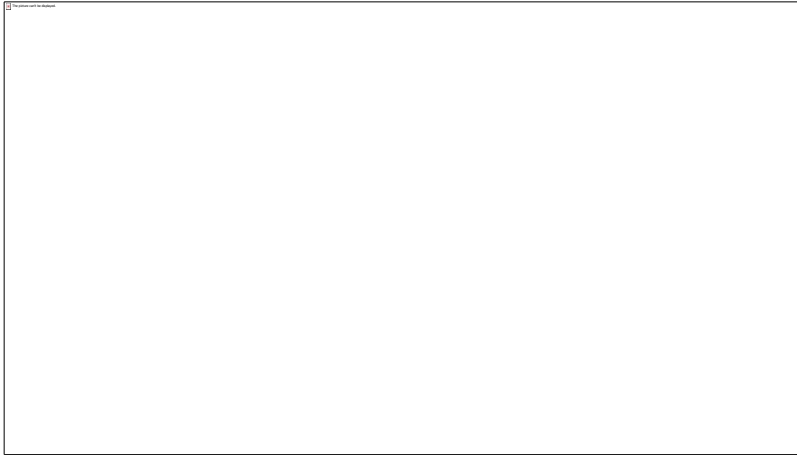
For this example, let’s group all modern methods (groups shaded below) into one category and all traditional methods and non-users (not shaded) into another category so we can use it in our analysis.



2. Notice that the output does not give us the number values for the categories. To do this we need to use the command `numlabel, add`.

numlabel, add

tab v312

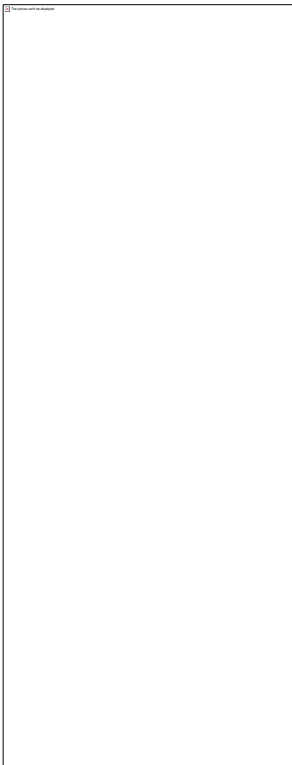


Now each category is shown with its number value. This will help us to recode the variable. Modern methods are 1, 2, 3, 4, 5, 6, 7, 11, 13, 14, and 17. Traditional methods are 8, 9, and 10 and non-users are 0.

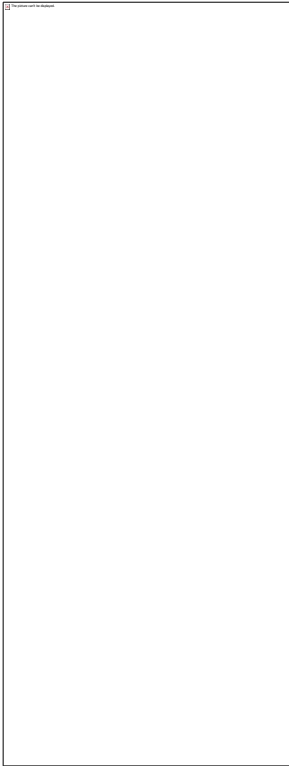
Step 3: Recode the variable with a meaningful name

- I. Since the variable name v312 doesn't mean anything to us, let's generate a new variable mcpr which can stand for "modern contraceptive use."

The first thing you want to do is generate a new variable and set it equal to missing. This means that every row in the dataset is set to missing. In Stata, this is indicated by a dot (.).



2. You then want to replace all women in the variable `mcpr` that are currently using a modern method of contraception with a value of 1. Women who are currently not using contraception or who are currently using a traditional method will be assigned the value of 0.



Type the following code into your `.do` file and create the variable `mcpr`:

.do file example:

```
*Recode current contraceptive use variable
```

```
/*Create the variable with missing values*/
```

```
gen mcpr=.
```

```
/*Replace the values with 1 if using a modern method*/
```

```
replace mcpr=1 if v312> 0 & v312<8
```

```
replace mcpr=1 if v312>10
```

```
/*Replace the values with 0 if using traditional or not using*/
```

```
replace mcpr=0 if v312==0 | (v312>7 & v312<11)
```

The | in the last line of the above code means “or”. We could have written this line as two separate lines like this:

```
replace mcpr=0 if v312==0
```

```
replace mcpr=0 if v312>7 & v312<11
```

or we could have written it using inlist, which tells Stata to set mcpr to a value of 1 if v312 is any of the listed values.

```
replace mcpr=0 if inlist(v312,0,8,9,10)
```

Similarly, we could have written the first replace line using inrange, which tells Stata to set mcpr to 1 if v312 is between the listed values. That alternative would look like this:

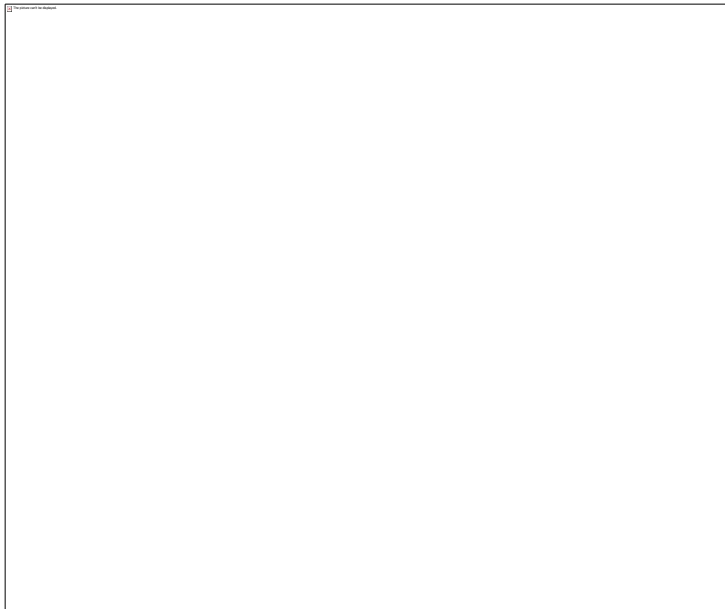
```
/*Replace the values with 1 if using a modern method*/
```

```
replace mcpr=1 if inrange(v312,1,7)
```

Step 4: Check the original variable with your recoded variable to make sure your variable recode was carried out correctly.

Make sure to crosstab the original variable (v312) and the new recoded variable (mcpr) to check whether your totals match. *If they don't, ask for help.*

```
tab v312 mcpr
```



Step 5: Properly label new variable:

Now that you have created your new variable of interest, you will need to properly label it. Dataset labels are displayed when you use the dataset and describe it (e.g. using the `describe` command).

1. First, we need to give a descriptive label to the variable. For example, we will label the variable `mcpr` as "Modern contraceptive use".

*create label for recoded variable

```
label variable mcpr "Modern contraceptive use"
```

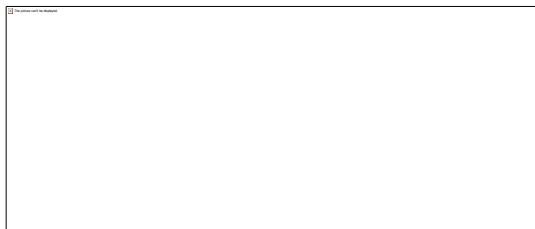
2. Next, you need to make a value label. Let's call our value label for this variable `yesno` to label the values of the variable `mcpr`. This is a two-step process where you first define the label, and then you assign the label to the variable. The `label define` command below creates the value label called `yesno` that associates 0 with no (indicating no modern method use) and 1 with yes (indicating modern method use). The `label values` command below associates the variable `mcpr` with the label `yesno`.

```
label define yesno 0 "No" 1 "Yes"
```

```
label values mcpr yesno
```

3. Now that you've created a new variable and set labels to the variable and values of interest, you should run a tabulation on `mcpr` to make sure that your labels were included correctly.

```
tab mcpr
```



CONGRATULATIONS! YOU HAVE SUCCESSFULLY RECODED AND LABELED A VARIABLE!

Extra Challenge!

Can you recode v312 so that it has three categories of non-user, traditional method user, and modern method user?

Write your work in your Stata do file.

NOTES ON EXERCISE 4

EXERCISE 5: RECODING EXAMPLE 2

The purpose of this exercise is to practice recoding variables in Stata.

After completing this exercise, you should be able to:

4. Properly recode “*In the last few months, have you heard about family planning on the radio?*” (Q714) in the Model Woman’s Questionnaire into categories.
5. Properly label new variable.

Stata command toolbox:

numlabel, add	adds the number labels to the categories of variables
recode	assigns different number codes to categories within a variable
	the vertical bar key () indicates an “or” statement in your syntax
missing	displays missing observations of the variable of interest



Before you begin

Please continue to use the workshop .do file used in previous exercises.

Challenge!

1. Let’s recode “*In the last few months, have you heard about family planning on the radio?*” (Q714) in the Model Woman’s Questionnaire into categories.

1. Which data file do you need to use?

2. Which weight do you need to use?

3. Which variable do you need to use?

HINT: You have the tools to look it up in Stata. Refer to previous exercises if you need to review what commands you could use.



4. Open the IR file and generate the weight variable. This should already be in your saved .do file.

.do file example

```
/*Recode example #2*/
```

```
cd "C:\Users\40825\Desktop\DHS Fellows\Data"
```

```
use ZZIR62FL.DTA, clear
```

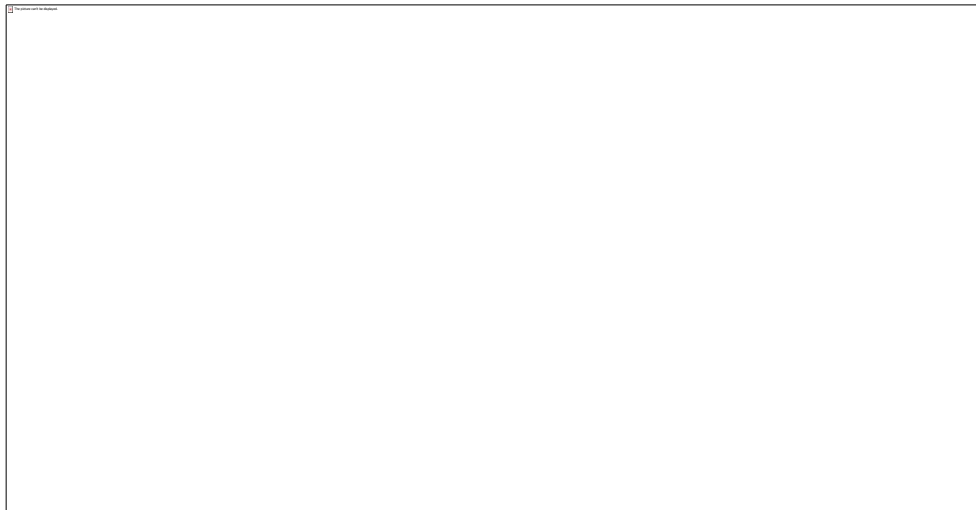
*Create relevant weight variable

```
gen wt = v005/1000000
```

Step 1: Check the original variable

1. Since we know we want to use variable v384a “*In the last few months, have you heard about family planning on the radio?*”, tabulate v384a to see the distribution of the variable.

```
tab v384a
```



2. Determine if there is missing data.

When thinking about missing data, we can use the missing command along with the tabulate command.

```
tab v384a [iweight=wt], missing
```

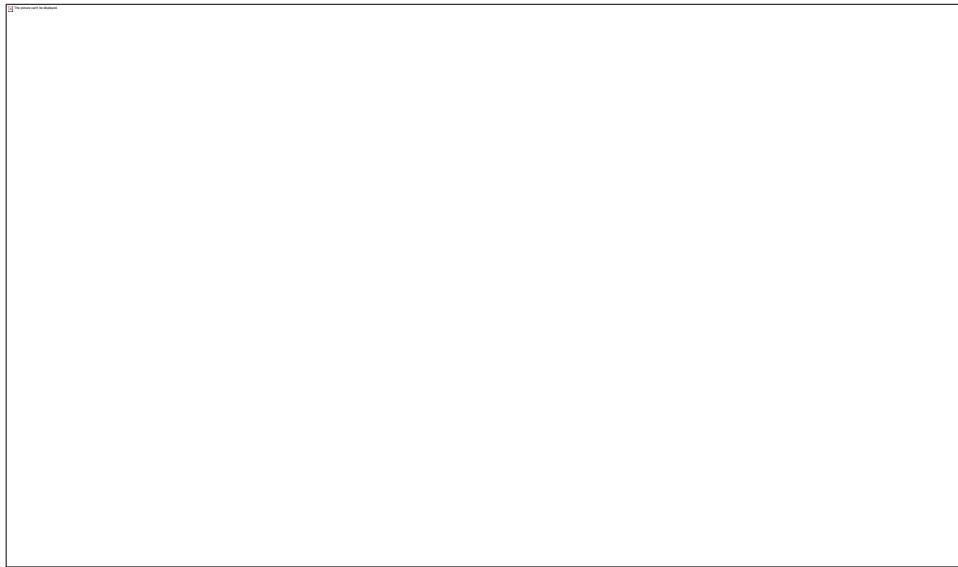
IMPORTANT! Missing data



When doing a tabulation on this variable, what are your assumptions in terms of possible survey responses? Would there be missing data? This is a possibility but might not be apparent when running a simple tabulation in Stata.

Think about the nature of the question, what types of potential questions may lead to or follow your survey question of interest.

How do you see missing data? Follow Step 1.b below



Missing data *continued*



In this example, missing data are coded as “.” So, there are 25 missing cases for the survey question: “*In the last few months, have you heard about family planning on the radio?*” In other cases, missing data might be coded as 9, 99, or 999, etc.

There are different types of missing data: system missing and missing information. System missing (when there is a dot [.] in Stata output) occurs when the question does not apply. In contrast, missing information is usually coded as 9, 99, 999, etc. when the question does apply, but the respondent did not answer for some reason, or the information is missing for other reasons. Therefore, we recommend you check for missing and these possible code categories to better understand reasons for missingness.

In this case all women should have been asked this question, so the missing cases may just be a data entry issue.

Step 2: Recode the variable with a meaningful name, and then recategorize it

Rename v384a to fp_message_radio, making the variable binary (2 categories only). One category will be: (1) women who have heard a radio message about family planning in the last few months prior to the interview; and (2) those who have not heard a radio message about family planning in the last few months prior to the interview.

Since you are renaming your variable and setting all yes responses (coded as 1) in the same way for the new coded variable, do the following:

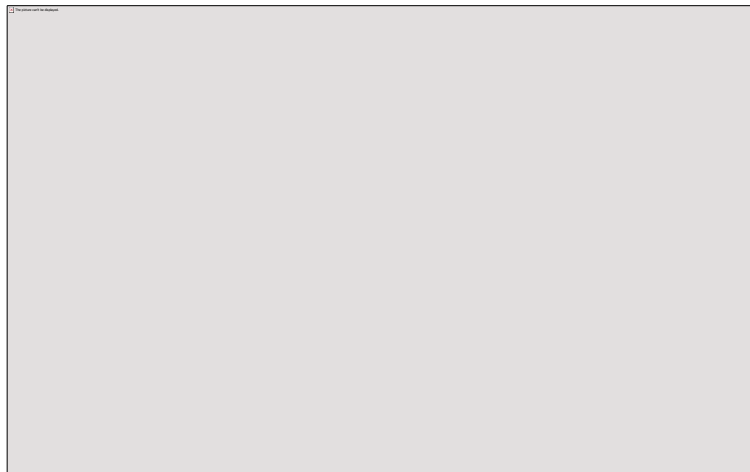
```
gen fp_message_radio=v384a==1
```

You have generated your new variable (fp_message_radio) and have set it equal to only the values coded as 1 in your original variable of interest (v384a). What this means is that the no responses (coded as 0) remain coded as 0 and the 25 missing cases are now coded as 0 as well.

Step 3: Check the original variable with your recoded variable to make sure you recoded correctly

Make sure to tabulate the old variable (v384a) and the new variable (fp_message_radio) to see if your totals match.

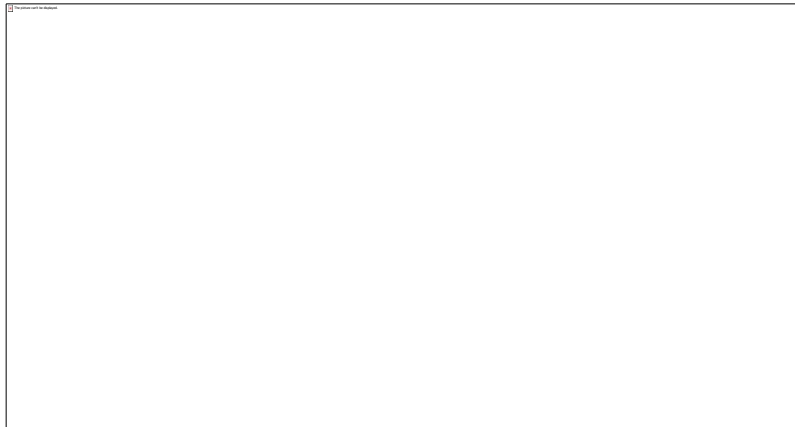
```
tab v384a fp_message_radio, missing
```



Step 4: Properly label your new variable

Your finished variable should look similar to the example below. You can check the newly labelled variable by running a tabulation and assessing whether everything ran correctly:

```
tab fp_message_radio [iweight=wt], missing
```

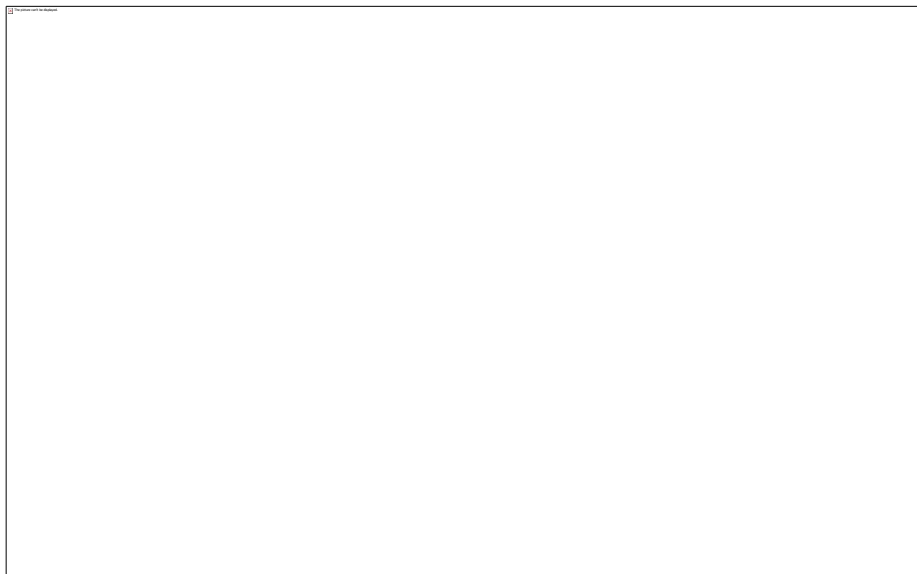


NOTE: There are other ways that you could generate new variables.

Another way to create variables is to use the **recode** command, which should give you the same variable. The **recode** command allows you to assign values of the original variable and to label the categories at the same time. This way of recoding variables takes fewer lines of syntax, so feel free to practice this approach as well.

```
recode v384a (0 . = 0 No) (1 = 1 Yes), gen(fp_message_radio2)
```

Check `fp_message_radio2` command through a tabulation and ensure that the recode makes sense.



NOTES ON EXERCISE 5

EXERCISE 6: SETTING THE SURVEY DESIGN AND USING SVY COMMANDS

The purpose of this exercise is to explain how to account for survey design and use svy commands for your analyses.

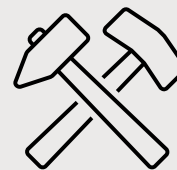
After completing the exercise, you should be able to:

1. Use svyset command to account for survey design
2. Use svy commands in a tabulation

Stata will assume your data is from a simple random sample if you do not tell it otherwise. Since, DHS surveys follow a multi-stage, clustered sample design, you need to tell Stata to account for this. After you've explored and prepared your data, it is important to account for survey design in your dataset before you run your analyses. You will want to identify the survey design characteristics to set up your svyset command. Examples of characteristics include probability weights; cluster sampling; and stratification.

Stata commands toolbox:

svyset	accounts for complex survey design in your dataset
egen	provides extensions to the generate command
svy: tab var	tabulates the variable <i>var</i>



The general command is as follows:

```
svyset [pw=x], psu(y) strata(z)
```

where pw stands for probability weight, x = weight variable, y = cluster variable, z = strata variable.

You have already learned how to create the wt (weight) variable in exercise 3.

Step 1. Identify the psu or cluster variable

The primary sampling unit (psu) or cluster variable is v021 in IR/KR/BR files, hv021 in HR/PR files, and mv021 in MR files.

Step 2. Identify or create the strata variable

The strata is usually v022, especially for recent surveys. However, for older surveys you will need to check the final report to see how the stratification was performed. It is usually by region (v024) and urban and rural areas (v025) and in this case you can create the strata variable as follows:

```
egen strata=group(v024 v025)
```

Step 3. Tell Stata to svyset your data

Once you have all the variables needed for the svyset, you can write the following command in your do file:

```
svyset [pw=wt], psu(v021) strata(v022) singleunit(centered)
```

**note: if you created the strata variable, use that variable instead of v022 above.*

This command now informs Stata that the data is coming from a complex survey design and not a simple random sample. You need only svyset the data once before analyzing the data. From this point forward, you can use the svy commands for your analysis commands.

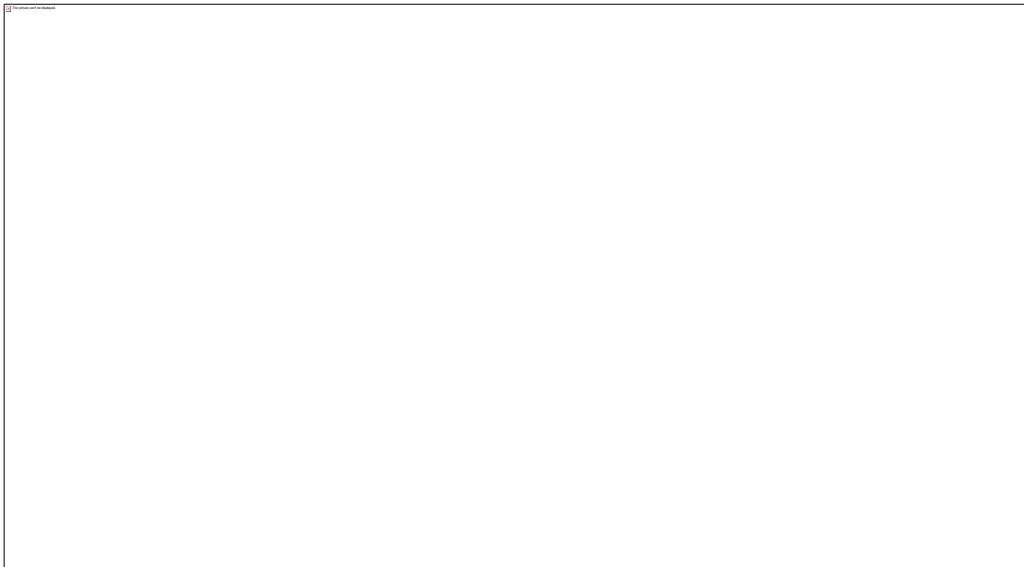
The addition of the option singleunit(centered) is added to avoid an error that could occur if there is a single primary sampling unit in a strata.

Step 4. Attach the prefix svy: to your tabulation commands

Now you can tabulate variables using svy: The svy: command works with most other analysis commands (frequency tabulations, crosstabs, regressions, and more). For now, we will work with the tabulate command. Let's try a few examples.

Tabulate region.

```
svy: tab v024
```



This gives you proportions.

If you want to see percentages, you can run the following command:

```
svy: tab v024, per
```

You may also want to see confidence intervals, in this case add the ci option:

```
svy: tab v024, per ci
```


Finally, try tabulating your recoded variables from previous exercises using `svy` (`mcpr` and `fp_message_radio`) with confidence intervals.

NOTE: You will obtain the same percentages and proportions when running a weighted tabulation and running an `svy` tabulation. For instance

```
tab v024 [iw=wt]
and
svy: tab v024, per
```

Will give you the same percentages for each region category.

However, the reason why we need to use `svy` is to get the correct standard errors (SE) and confidence intervals (CI). We also use `svy` for crosstabulations and regressions in this exercise book. When using DHS data and performing any statistical testing or reporting confidence intervals, then you must use `svy`. If you are only showing percentages, then you can just use the weighted tabulation commands.

NOTES ON EXERCISE 6

EXERCISE 7: EXPORT FREQUENCY TABULATIONS TO EXCEL

The purpose of this exercise is to export results of frequency tabulations to Excel to prepare a results table or chart. This is useful for producing a descriptive table of the background characteristics of your analytic sample.

After completing this exercise, you should be able to:

3. Download and install the about program
4. Run a frequency of selected variables
5. Export results into Excel

Stata command toolbox:

findit	finds and installs user-written programs
tab	runs a frequency tabulation or cross tabulation
about	helps to export your tabulation results to Excel

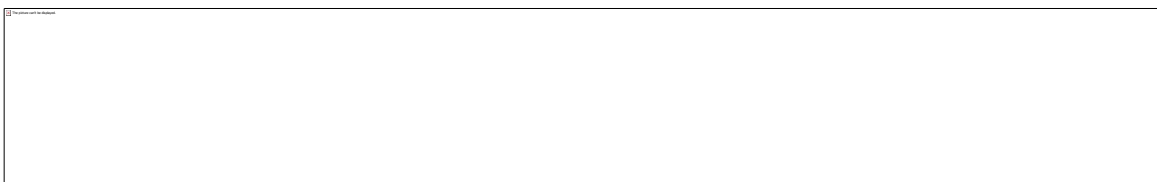


Step 1: Install the about user-written program

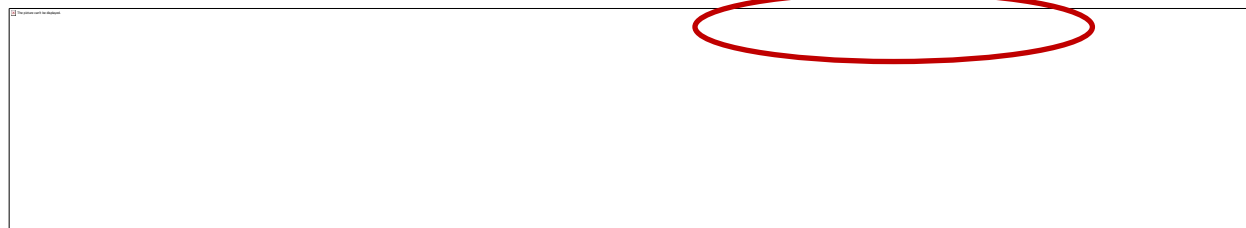
1. You will need to install the Stata program about to export frequency tabulation results into Excel.

In the command window, type findit about

2. Scroll down until you see:



3. Click on the blue link and scroll down to the installation file and press “click here to install”



This will install the `tabout` program into Stata.

Step 2: Open the individual recode file and run a frequency of selected variables

1. To open the individual file and run a frequency of selected variables, type the use command to open a dataset.

*Open the IR dataset

```
use ZZIR62FL.DTA, clear
```

2. Use the `lookfor` command to identify 3 or 4 variables you want to use in your analysis. Alternately, try urban/rural residence (`v025`), household wealth quintile (`v190`), and completed education (`v106`). Adapt the syntax you used in Exercise 2 and Exercise 3 to run weighted tabulations for each of these selected variables in the IR dataset.

.do file example

*Exercise 1

*January 10, 2019

```
cd "C:\Users\40825\Desktop\DHS Fellows\Data"
```

```
use ZZHR62FL.DTA, clear
```

*Tabulation for urban/rural residence

```
tab hv025
```

*Open the IR dataset

```
use ZZIR62FL.DTA, clear
```

*Tabulation for urban/rural residence

```
tab v025 [iw=wt]
```

* Tabulation for urban/rural residence using `svy` as described in the previous exercise. This will give the same percentages as the weighted tabulation but you need to use `svy` to get the right standard errors and confidence intervals.

svy: tab v025 , per



TIP: You do not need to run the frequency in Stata before exporting the results into Excel with the `tabout` command. However, it is useful to confirm that you exported the results in the correct format as you were expecting to. For example, did you export the weighted N and not the unweighted N?

Step 3: Export the results into Excel

To export your estimates while applying the `svy` option, you need to first use `svyset` as described in the previous exercise.

1. Copy the following command into your `.do` file and run it. This will export your frequency of rural/urban residence to an excel file called `Residence.xls`. `tabout` will save the Excel to the folder directory where your datasets are located.

```
tabout v025 [iw=wt] using "Residence.xls", ///
```

```
c(cell) svy f(1) clab(Col%) nwt(wt) per pop replace
```

This command produces a table of single table frequency variables. Note that `svy` is used in the options after the `tabout` command.

`f(1)` indicates that I want to produce estimates with 1 decimal place. You can change this to `f(2)` for two decimal places.

2. To produce a table displaying the frequency tabulation of all the variables, copy the following command into your `.do` file and run it. This will export your frequencies of all listed variables into a single Excel file called `Background_Table.xls`. The one-way option tells Stata to produce a one-way frequency table.

In the example below I also indicated that I want the confidence intervals by adding `ci` in the `c(cell ci)` option.

```
tabout v025 v190 v106 [iw=wt] using "Background_Table.xls", ///  
c(cell ci) f(2) svy nwt(wt) per pop oneway replace
```

NOTES ON EXERCISE 7

EXERCISE 8: CROSS TABULATIONS

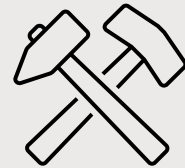
The purpose of this exercise is to explain how to carry out cross tabulations in Stata.

After completing this exercise, you should be able to:

3. Understand the associations between two variables of interest
4. Export cross tabulation results to excel

Stata commands toolbox:

<code>tab var1 var2, option</code>	options for two-way tabulate commands
<code>cell</code>	cell percentage (%)
<code>row</code>	row %
<code>column (or col)</code>	column %
<code>nofreq</code>	do not display frequencies



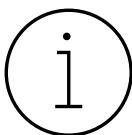
Before you begin:

For this exercise you will need:

5. the following datasets:
ZZIR62FL.DTA
6. to reference the following table(s):
Table 3.2.1 Educational attainment: Women

Please continue to use the workshop .do file created for previous exercises.

Cross Tabulations



Cross Tabulations

Rather than a frequency of one variable, cross tabulations are carried out when you want to understand relationships between two variables at once.

Tabulate one variable:

```
tab var1
```

Crosstab of two variables:

```
tab var1 var2
```

Please Note!

The order in which you list the variables in the tabulation matters. The first variable listed in your Stata syntax will be displayed in a row format. The second variable listed will be presented in a column format.

Before you carry out a cross tabulation of two variables of interest, *what are you expecting to see?* If it helps, sketch out a table of what you would like to see. *How would one variable be potentially associated with another?* These are some questions to consider.

For example:

tab v025 v024

Residence	Region			
	A	B	C	D
Urban				
Rural				

tab v024 v025

Region	Residence	
	Urban	Rural
A		
B		
C		
D		

You can additional syntax to get column percents, row percents, and the number of observations in your sample (Ns).

tab v025 v024, row

Residence	Region				Total
	A	B	C	D	
Urban	24%	23%	30%	23%	100%
Rural	48%	28%	4%	20%	100%
Total	40%	26%	12%	21%	100%

tab v025 v024, column

Step 1: Identify your variables of interest

Use syntax from the previous exercises and look for the following variables:

7. Residence
8. Region

Step 2: Run the cross tabulation

Once you've found your variables of interest, run your cross tabulation using `svy`.

Remember if you have not already done so, you need to run the `svyset` command to be able to use `svy` commands



TIP: You can use the command `nofreq` to only view the percentages, without the frequencies.

Step 3: Export the results into Excel

After running the cross-tabulation results, you then want to export the results into Excel. This is useful if you want to create a single descriptive table, disaggregating background characteristics by a dichotomous or categorical variable (for example) or if you want to create a chart showing the results of a contingency table. You can use the program called `tabout`, which you installed at the beginning of Exercise 4 to export to Excel.

tabout

Copy the following command into your `.do` file and run it. This will export your results to an Excel file called `Crosstab.xls`.

```
tabout v024 v025 [iw=wt] using "Crosstab.xls", ///  
c(col) f(1) stats(chi2) svy nwt(wt) per pop replace
```

As before, you can produce a single table with multiple cross-tabulations. The last variable in the variable list is the variable by which all previous variables are cross tabulated with. This syntax produces a table of region (`v024`), household wealth quintile (`v190`), and educational attainment (`v106`) disaggregated by urban/rural residence (`v025`).

```
tabout v024 v190 v106 v025 [iw=wt] using "Crosstab.xls", ///  
c(col) f(1) stats(chi2) svy nwt(wt) per pop ptotal(none) replace
```

NOTE!

Since you have adjusted for the complex survey design using the `svyset` command described in Exercise 6, you can export the results of a chi-square test of independence to assess if there is an association between your cross-tabulated variables. This was performed by adding the options `svy` to indicate complex survey design and `stats(chi2)` to report the chi-square in the above command.

If you also want to show confidence intervals, add `ci` in `c(col)`.

Challenge!

What is the association between modern contraceptive use and place of residence? We have coded modern contraceptive use in Exercise 4 (`mcpr`). Think of whether you want a row or column percentage.

Now write the code for how to output the cross tabulation of `mcpr` with place of residence (`v025`), region (`v024`), wealth quintile (`v190`), and education level (`v106`) and with confidence intervals.

Are the associations significant?

Check the answer key for the correct answers.

NOTES ON EXERCISE 8

EXERCISE 9: TABULATIONS FOR A SUB-POPULATION

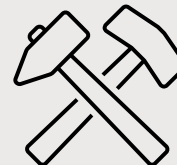
The purpose of this exercise is to explain restricting tabulations in Stata, to be able to specify tabulations based on your analysis.

After completing this exercise, you should be able to:

9. Restrict tabulations based on your analysis of interest by using the `if` command to go along other Stata commands

Stata commands toolbox

<code>if</code>	used in conjunction with other commands to restrict datasets
<code>codebook</code>	describe data contents such as variable names and labels
<code>nolabel</code> (<code>nolab</code>)	specifies in the data to remove variable label



Logical Operators		Relational Operators			
<code>&</code>	and	<code>></code>	greater than	<code><=</code>	less than or equal to
<code> </code>	or	<code><</code>	less than	<code>==</code>	equal to
<code>!</code>	not	<code>>=</code>	greater than or equal to	<code>!=</code>	not equal to



Before you begin:

For this exercise you will need to refer to the following table from the Model Questionnaire tables (found in the `zzfulltables` folder):

- I. Table 2.7

Use the following datafile:

- I. `ZZPR62FL.DTA`

Please continue to use the workshop `.do` file created in previous exercises.

Restricting tabulations using the `if` command



Restricting tabulations using the `if` command allows us to focus our analysis on what we are interested in seeing in the data.

Here's an example when using an `if` command:

```
tab variable V if variable W==1
```

What does this mean? Imagine that you wanted to tabulate the distribution of variable `V` only when the variable `W` is equal to 1 in your dataset. By running this syntax in Stata, you would only look at the relationship between variables `V` and `W` only if variable `W` is equal to 1.

Therefore, we can use the `tabulate` command in conjunction with the `if` command along with the following qualifiers in order to specify what we'd like to see in the dataset:

Logical Operators		Relational Operators			
&	and	>	greater than	<=	less than or equal to
	or	<	less than	==	equal to
!	not	>=	greater than or equal to	!=	not equal to

Example 1: Make a table to look at the percentage of urban/rural households by region 3

What variables do you need to run this syntax?

(Remember! You can use the `lookfor` command in your Stata `.do` file for variables of interest!)

Below is the syntax and the table that you can expect to see:

```
tab hv025 if hv024==3
```

Type of place of residence	Region
	3
<i>urban</i>	
<i>rural</i>	

Before moving forward, are you seeing the results in Stata that you expect to see? What does the if statement do as compared to running a cross tabulation without it? Are you getting the same numbers for the sub-group of interest? **Let's try another example where you can use an if statement!**

Example 2: Check the association between residence by region 3 or region 4

```
tab hv025 if hv024==3 | hv024==4
```

Type of place of residence	Region (only looking at 3 or 4)
<i>Urban</i>	
<i>Rural</i>	

You can experiment with different **if** statement options based on what you want to examine.

Challenge!

Match the denominator for Table 2.7 Household population by age, sex, and residence.

1. What is the denominator?

2. What does “de facto household population” mean in the subheading of this table?

3. What data file do we need to use?

4. Which weight do we need to use?

5. What variable do we need to use for the age of household members?

(Remember! You can use the lookfor command in your Stata .do file for variables of interest!)

6. Now that we know the age of the household members, which members are we interested in exactly? Based on the sub-heading, we are interested in the de facto household members, so those who “slept [in the household] last night”

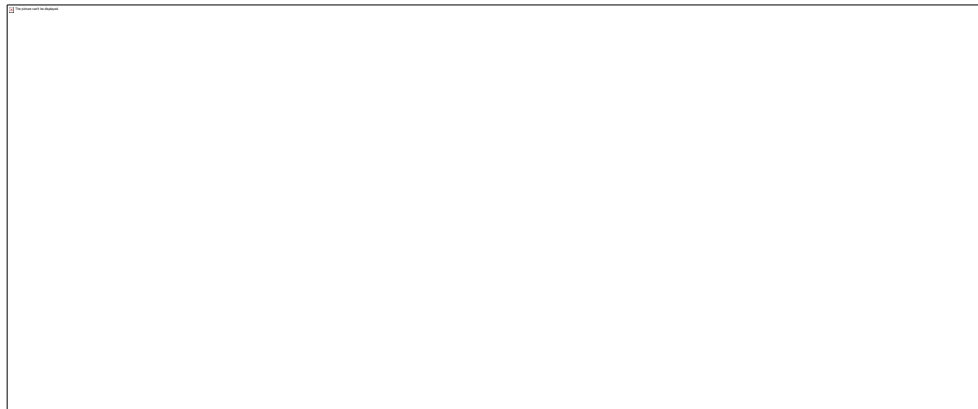


Let's restrict the tabulation to household members who reported sleeping in the household the night before the survey.

Since the variable for "slept last night" is hv103, let's run a weighted tabulation. From there, we will want to understand the frequency and percentage of those who have slept in the household the previous night or who answered "yes."

numlabel, add

tab hv103 [iweight=wt]

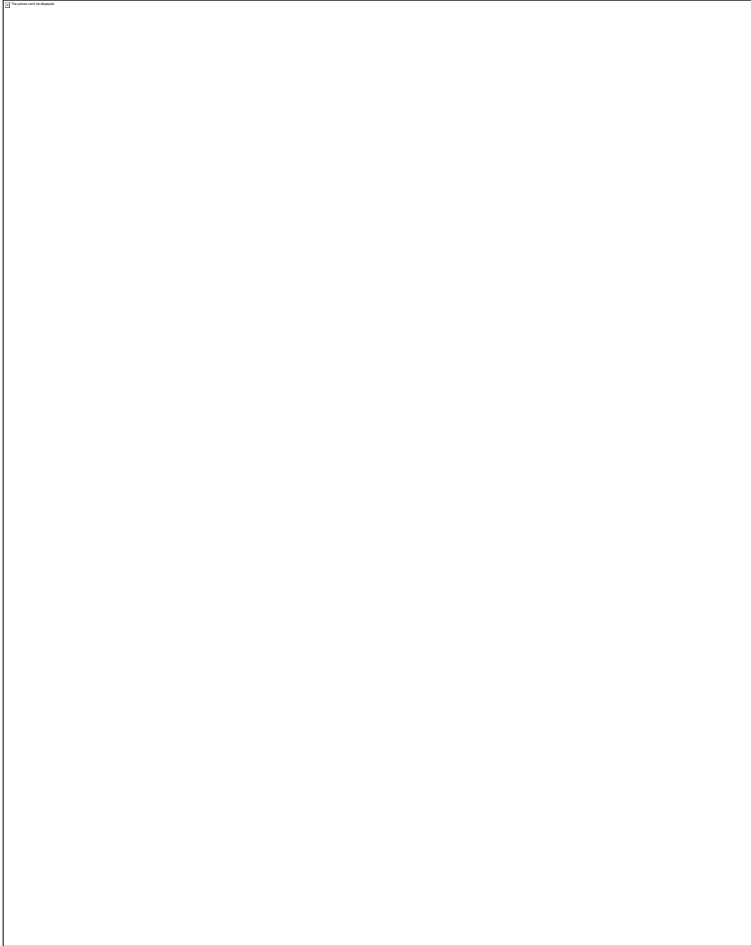


For more information on the difference between de jure and de facto terms within DHS surveys, watch our 5-minute tutorial. On The DHS Program's YouTube channel, type "De Jure and De facto," or scan the QR code below:



7. Put all this together to try to match the denominator of this table.

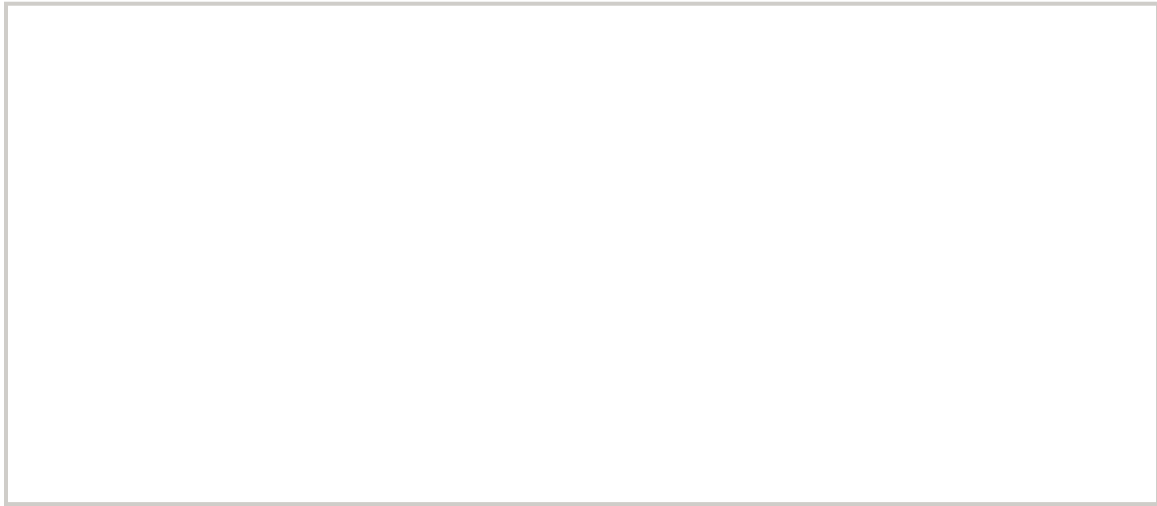
Check your .do file with the answer box at the end of this workbook.



-
-
-



8. Lastly, match the total number of respondents who slept in the household last night to Table 2.7 in the model dataset files.



NOTES ON EXERCISE 9

EXERCISE 10: MERGE DHS DATA FILES

The purpose of this exercise is to demonstrate merging different data files in Stata.

After completing this exercise, you should be able to:

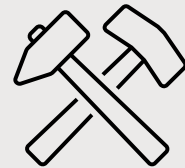
9. Learn how to merge the household (HR) dataset with the individual recode (IR) file and analyze the data across the two files.

To download Stata code for other types of merges please visit our [Analysis Repository on GitHub](#). You can find the code on [github.com>DHSProgram>DHS-Analysis-Code>tree>main>MergeCode](#) or by scanning the QR code below:



Stata commands toolbox:

describe	used to get to know your data file (e.g. variables that included, type of variable, variable label details)
merge	joins corresponding observations from the dataset currently in memory with those from another dataset, matching on one or more key variables

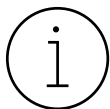


Before you begin:

For this exercise you will need the following datasets:

10. ZZIR62FL.DTA
11. ZZHR62FL.DTA

Merging



Merging 101

First, what is a merge?

The merge command adds new variables from a second dataset to existing observations in the dataset that is currently open/in Stata's memory. A merge therefore joins corresponding observations from the data file currently in memory – referred to as the *master* data file – with those from another data file – referred to as the *using* data file – matching on one or more key variables.

Secondly, why merge?

Merging data files is necessary when the information you need to answer your research question is available but is not contained in a single data file. Some variables of interest in the DHS survey household data file, for instance, are not included in the individual files. We therefore need to find a way to incorporate necessary variables into one dataset.

So, what do we need to merge?

In addition to your variables of interest, a variable that is common across datasets is needed to successfully merge data files. You will need to identify – or create – a common variable (think of this variable as your unique ID) that is the same in both data files. By identifying or creating that unique ID variable, you will be able to successfully match the data from different datasets to the right case.

For this exercise, let's consider the following research question, which calls for merging of two different DHS data files:

What are the effects of parental survivorship and household characteristics on women's reproductive behaviors and childbearing intentions?

To conduct a merge, there are just 7 simple steps to carry out!

Step 1: Determine the master and using data files

Based on the research question, what is our unit of analysis?

What is the name of the *master* data file?

What is the name of the *using* file?

How do you know which data file is the *master*?

Circle the type of merge that we are carrying out using the *master* and *using* files.

1:1

m:1

.do file example

*Merging DHS Data Files

```
cd "C:\Users\40825\Desktop\DHS Fellows\Data"
```

*Open dataset with your primary unit of analysis

```
use ZZIR62FL.DTA, clear
```

Step 2: Get to know your data to identify—or create—the unique ID variable

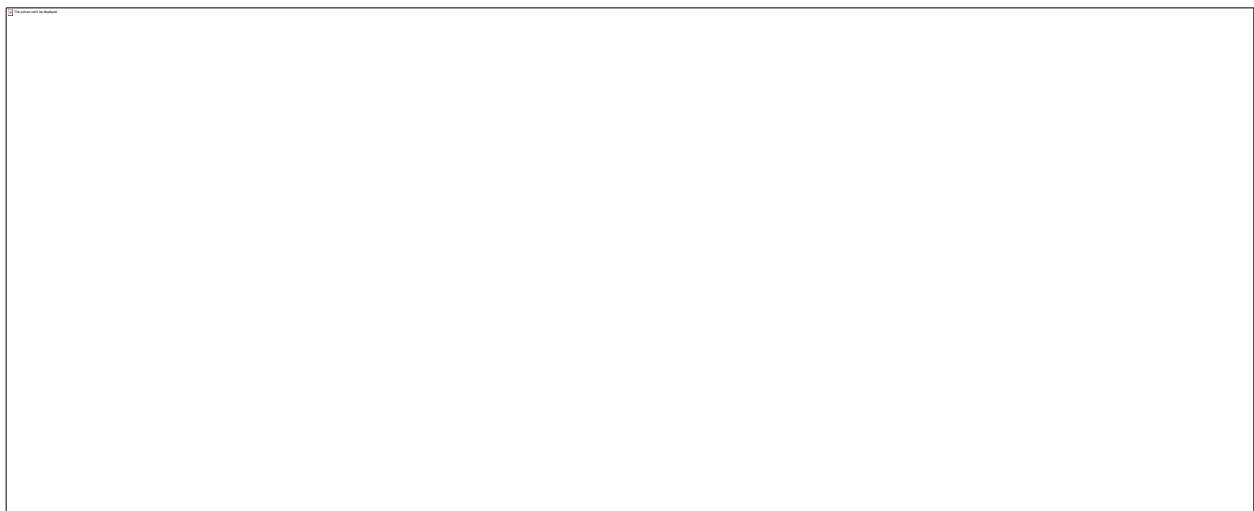
Before merging datasets, get to know your data and make sure that your datasets are sorted by a common variable.

Given that DHS datasets vary, it's important to get to know your data first by entering the describe command. Alternatively, you can look at the variable list on the righthand side of the Stata output window. Variable names can be found on the left-hand side when running the describe command or scrolling through the Variables menu. The Variables Manager (an icon can be found at the top of the Stata output window) is also a helpful way to understand what variables are included in each dataset.

In this case, you are looking for variable v001 (labeled as “cluster number”) and variable v002 (labeled as “household number”). These are common variables in both datasets *with different variable names*.

Here is an example of what your Stata output will look like if you use the describe command for the using dataset, which is in the household dataset.

When you look at the Variables Manager found at the top of the Stata window (hover over the different icons at the top of the output window), you'll find something like this:



Therefore, to merge the two data files, the common variable(s), also known as our unique ID variables, are: **hv001 (cluster number)** and **hv002 (household number)**

Step 3: Make sure the unique ID variable is the same in both datasets

If you couldn't find a common variable in the two datasets, let's create a new variable name found in the *master* data file, which will be the same name as the variable in the *using* data file.

Here, we name the new variables **v001** and **v002**, which are found in the *master* dataset.

```
generate v001=hv001
```

```
generate v002=hv002
```

You can rename variables instead of generating new ones. For example:

*Rename merging variables(identifiers)

```
rename (hv001 hv002) (v001 v002)
```

Step 4: Save the using file under a temporary name

If you renamed any variables in your *using* data file to match your *master* data file, you must save your updated *using* data file under a temporary file name.

```
save HRtemp, replace
```

Step 5: Open the master file, identify the unique ID variable (common variable)

Open the master file

```
use ZZIR62FL.DTA, clear
```

*check whether the variables of interest are in data file

```
describe (v001 v002)
```



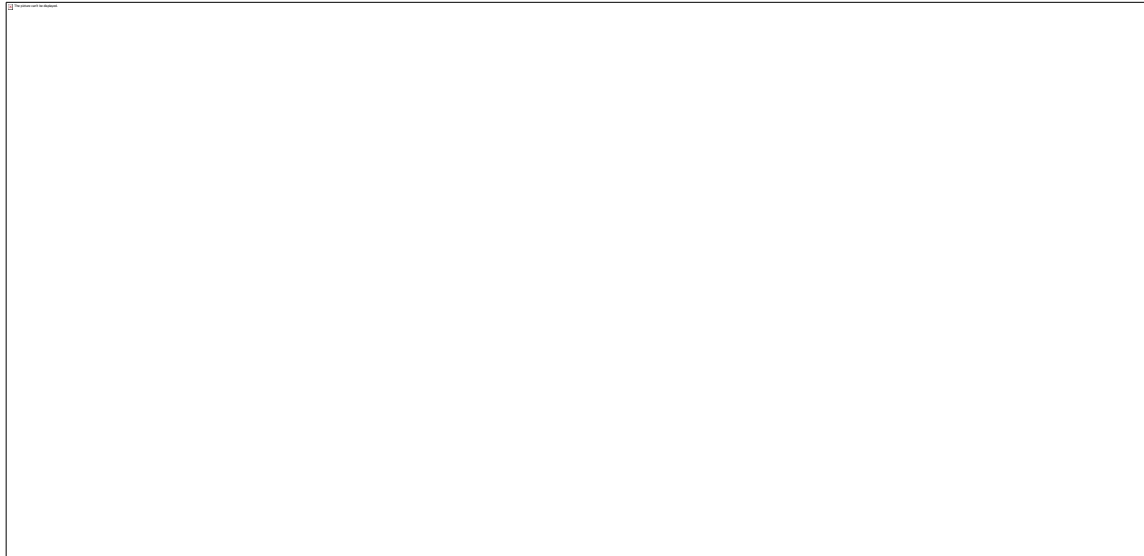
Step 6: Merge the IR and HR datasets

Now that we have our common variables to merge our two data files, let's go ahead and merge the master individual recode (IR) file (ZZIR62FL.dta) with the temporary household recode (HR) file (HRtemp.dta) that you created.

In terms of the type of merge we want to carry out, this is a **many:1 merge** because the common variables (v001 and v002) can correspond to many observations in the master dataset, but uniquely identifies individual observations in the using dataset. Specifically, many individuals can have the same cluster number and household number per household.

```
merge m:1 (v001 v002) using HRtemp.dta
```

Step 7: Check results of the merge and handle unnecessary cases



Note that 974 observations did not match, and these came from the using data file, which is the HR file.

Question: Why are some cases not matched from the using file?

Once you've understood why some files are not merged, we can drop these files and save our newly merged dataset with the name of your choosing. For example:

Keep only the observations that are matched (`_merge==3`, or the 8,348 observations):

```
keep if _merge==3
```

```
save IR_HR_merged, replace
```



Tips for Merging:

Keep in mind:

1. Before merging any data files, think conceptually about the type of merge you want to conduct and what you expect your updated data file to look like. Examine the `_merge` variable to ensure that running your merge worked as you expected (`many:1`; `1: many`; `1:1`).
2. If something doesn't match, this doesn't necessarily mean that the merge was wrong, but

NOTES ON EXERCISE 10

EXERCISE 11: LOGISTIC REGRESSION

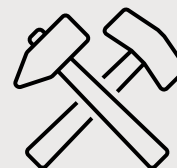
The purpose of this exercise is to explain logistic regression in Stata.

After completing this exercise, you should be able to:

12. Carry out logistic regression to assess whether there's an association between current family planning (FP) method use to and hearing about FP on the radio among women aged 15-49.
13. Output regression results to Excel.

Stata commands toolbox:

findit	finds and installs user-written programs
logistic	carries out logistic regression
outreg2	helps to export your regression results to Excel
putdocx	exports your regression results to Word

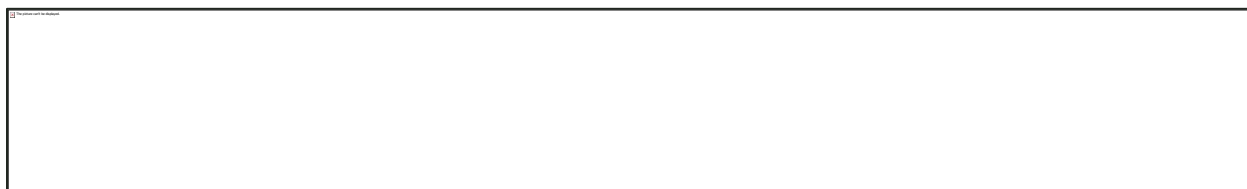


Before you begin

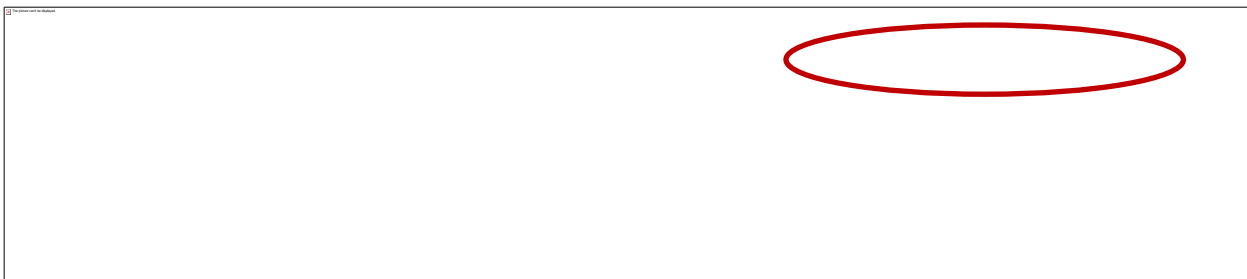
You will need to install the Stata program outreg2 to export the regression results into Excel.

In the command window, type `findit outreg2`

Scroll down until you see:



Click on the blue link and scroll down to the installation file and press “click here to install”



This will install the outreg2 program into Stata.

In this exercise, our *dependent variable* (also referred to as *outcome variable*) is:

14. FP method use among women aged 15-49 (“Are you currently doing something or using any method to delay or avoid getting pregnant?” – Q303) (response options: yes/no)

Our *independent variables* (also referred to as *exposure variables*) include:

15. Hearing about FP on the radio
16. Residence
17. Region
18. Education
19. Wealth

Step 1: Prepare your data

Using your .do file:

1. open the IR file (the Woman’s Questionnaire) in Stata
2. generate wt
3. svyset

Step 2: Create and label variables of interest

The first thing you should always do before attempting to run a logistic regression is to match your variables to the report. For this example, match the prevalence of contraceptive use, and exposure to family planning messages on the radio to tables from report.

Calculate “Percentage of women aged 15-49 who currently use anything to delay or avoid getting pregnant

This is different from the variable we coded in Exercise 4 which is women using a modern method.

1. **FPmethod_use** (v313):

1. Create a 2-category binary variable:
 1. 1=yes – folkloric method, traditional method, or modern method
 2. 0=no AND missing (.)
2. Match figures with Table 7.3: Current use of contraception by age

Calculate “Percentage of women aged 15-49 who heard about family planning on the radio in the last few months”.

You already coded this variable in Exercise 5.

2. **FPradio** (v384a)

1. 1=yes if woman heard about family planning on radio

2. 0= no if woman did not hear about family planning on radio AND missing (.)
3. Match figures with Table 7.14 Exposure to family planning messages: Women

Step 3: Account for complex survey design in your dataset

As shown in Exercise 6, we run the svyset command.

```
svyset [pw=wt], psu(v021) strata(v022) singleunit(centered)
```

Step 4: Run unadjusted regression model (using svy)

```
svy: logistic FPmethod_use FPradio
```

```
. svy: logistic FPmethod_use FPradio
(running logistic on estimation sample)
```

Survey: Logistic regression

```
Number of strata =      27          Number of obs   =    8,348
Number of PSUs   =     217          Population size = 8,347.9996
                                          Design df      =      190
                                          F( 1, 190)    =    27.48
                                          Prob > F      =    0.0000
```

FPmethod_use	Linearized				
	Odds Ratio	Std. Err.	t	P> t	[95% Conf. Interval]
FPradio	2.122694	.3047679	5.24	0.000	1.599159 2.817623
_cons	.194733	.0154482	-20.62	0.000	.1665254 .2277186

Note: `_cons` estimates baseline odds.

Note: Strata with single sampling unit centered at overall mean.

Step 5: Run adjusted model

Run an adjusted model by including residence (v025), region(v024), education(v106), and wealth(v190). In this case, you don't have to recode these covariates, but in others you may have to. It is always recommended to check all the variables before entering them into the model.



Tip:

If your variable names have a capital letter in them, you must always capitalize that letter or Stata will not be able to identify the variable. Remember that **Stata is case sensitive.**

```
svy: logistic FPmethod_use i.FPradio i.v025 i.v024 i.v106 i.v190
```



```
. svy: logistic FPmethod_use i.FPradio i.v025 i.v024 i.v106 i.v190
(running logistic on estimation sample)
```

Survey: Logistic regression

```
Number of strata =      27          Number of obs   =    8,348
Number of PSUs   =    217          Population size = 8,347.9996
                                          Design df      =     190
                                          F( 12, 179)    =    21.56
                                          Prob > F       =    0.0000
```

FPmethod_use	Linearized				[95% Conf. Interval]	
	Odds Ratio	Std. Err.	t	P> t		
FPradio						
Yes	1.635151	.1723029	4.67	0.000	1.328275	2.012925
v025						
rural	.5757628	.0717281	-4.43	0.000	.4503206	.7361484
v024						
region 2	1.373992	.1699425	2.57	0.011	1.076535	1.753639
region 3	1.132195	.2414875	0.58	0.561	.7433668	1.724404
region 4	1.001698	.1222447	0.01	0.989	.7873942	1.274327
v106						
primary	.9599534	.0939896	-0.42	0.677	.7913602	1.164464
secondary	1.900805	.1863577	6.55	0.000	1.566569	2.306352
higher	3.923351	.8297516	6.46	0.000	2.585128	5.954321
v190						
poorer	.9905755	.1039056	-0.09	0.928	.8054325	1.218277
middle	1.051418	.105682	0.50	0.618	.8623224	1.281981
richer	1.190218	.1415178	1.46	0.145	.9413889	1.504817
richest	.8960422	.1683573	-0.58	0.560	.6185446	1.298034
_cons	.1994569	.0357834	-8.99	0.000	.1400106	.2841431

Note: _cons estimates baseline odds.

Note: Strata with single sampling unit centered at overall mean.

```
.
end of do-file
```

Step 6: Export results into Excel or Word

After running the adjusted logistic regression results, we then want to export the results into Excel or Word. We can use the program called `outreg2`, which you installed at the beginning of the exercise to export to Excel. You can also use `putdocx` to export the results to Word.

Exporting to Excel using outreg2

Copy the following command into your `.do` file and run it. This will export your results to an Excel file called `Regression.xls`. `outreg2` will save the Excel to the folder directory where your datasets are located.

```
outreg2 using regression.xls, eform stats(coef ci) sideways dec(2) label(insert) alpha(0.001, 0.01, 0.05)
replace
```

VARIABLES	LABELS	(1)	(2)
		FPmethod_us e coefEform	ciEform
FPmethod_use	RECODE of v313 (current use by method type)	.	.-.
	RECODE of v384a (heard family planning on radio last few months) = 1,		
1.FPradio	Yes	1.64***	1.33 - 2.01
2.v025	type of place of residence = 2, rural	0.58***	0.45 - 0.74
2.v024	region = 2, region 2	1.37*	1.08 - 1.75
3.v024	region = 3, region 3	1.13	0.74 - 1.72
4.v024	region = 4, region 4	1.00	0.79 - 1.27
1.v106	highest educational level = 1, primary	0.96	0.79 - 1.16
2.v106	highest educational level = 2, secondary	1.90***	1.57 - 2.31
3.v106	highest educational level = 3, higher	3.92***	2.59 - 5.95
2.v190	wealth index = 2, poorer	0.99	0.81 - 1.22
3.v190	wealth index = 3, middle	1.05	0.86 - 1.28
4.v190	wealth index = 4, richer	1.19	0.94 - 1.50
5.v190	wealth index = 5, richest	0.90	0.62 - 1.30
Constant	Constant	0.20***	0.14 - 0.28
Observations		8,348	

seEform in parentheses

*** p<0.001, ** p<0.01, * p<0.05

Exporting to Word using putdocx

You can also use a new Stata 15 command `putdocx` to export the results to Word.

```
putdocx clear
```

```
putdocx begin
```

```
putdocx paragraph
```

```
putdocx table mytable = etable
```

putdocx save regression_results.docx, replace

FPmethod_use	Odds Ratio	Linearized Std. Err.	t	P> t	[95% Conf. Interval]	
FPradio						
Yes	1.635151	.1723029	4.67	0.000	1.328275	2.012925
v025						
rural	.5757628	.0717281	-4.43	0.000	.4503206	.7361484
v024						
region 2	1.373992	.1699425	2.57	0.011	1.076535	1.753639
region 3	1.132195	.2414875	0.58	0.561	.7433668	1.724404
region 4	1.001698	.1222447	0.01	0.989	.7873942	1.274327
v106						
primary	.9599534	.0939896	-0.42	0.677	.7913602	1.164464
secondary	1.900805	.1863577	6.55	0.000	1.566569	2.306352
higher	3.923351	.8297516	6.46	0.000	2.585128	5.954321
v190						
poorer	.9905755	.1039056	-0.09	0.928	.8054325	1.218277
middle	1.051418	.105682	0.50	0.618	.8623224	1.281981
richer	1.190218	.1415178	1.46	0.145	.9413889	1.504817
richest	.8960422	.1683573	-0.58	0.560	.6185446	1.298034
_cons	.1994569	.0357834	-8.99	0.000	.1400106	.2841431

NOTES ON EXERCISE 11



ANSWER KEYS

Exercise 1

1. Which questionnaire collects information on employment and gender roles (for example, who makes decisions regarding the household)?

The woman's questionnaire.

2. Which questionnaire is used to ask parent/caregivers to provide consent for anemia and malaria tests in children?

The household questionnaire.

3. What kind of data is collected on the topic of health insurance? What are some questions that are asked? Which respondents provide answers to these questions?

Whether or not the respondent is covered by health insurance, and what type of health insurance they have. This is asked to both men and women.

4. What are the possible response categories for the person who carried out a male circumcision? What are the different response options for where the circumcision was done?

The response options for the person who carried out a male circumcision are: 1- Traditional practitioner/family friend 2- Health worker/professional 3- Other 4- Don't know

The response options for where it was done are: 1- Health facility 2- Home of a health worker/Professional 3- Circumcision done at home 4- Ritual Site 5- Other home/place 8- Don't know

5. Antenatal care information is collected for only the last pregnancy that ended in a live birth in the last 5 years.

False. Antenatal care information is collected for all pregnancies in the last 5 years regardless of survival status.

6. DHS surveys collect information from the respondent on her partner's opinion in terms of where she gives birth.

False. DHS surveys do not collect information from the respondent on her partner's opinion in terms of where she gave birth.

7. Men are asked to provide a full birth history of all births that have occurred (alive or not)

False. Women are asked to provide a full birth history of all births that have occurred (alive or not).

8. Hemoglobin tests are offered for all children in the household aged 0-59 months.

False. Hemoglobin tests are offered for all children in the household aged 6-59 months.

9. Information on children's history and treatment of fever is collected from mothers and fathers in standard DHS surveys.

False. Information on children's history and treatment of fever is collected only from mothers.

10. DHS surveys collect information on the times that the respondent and her partner have used a method to avoid getting pregnant.

True

Exercise 2

Step 3.a. What is the variable for urban/rural residence in the IR file?

The variable we are interested in is v025.

Quick Tip: To get this result in Stata, you could type the following syntax: lookfor residence. The "residence" word in our syntax is part of the variable label in DHS questionnaires. Note that if you type something like lookfor urban, this syntax would yield different variable results. Try it and see.

Step 3: Extra Challenge

1. What are the number of cases for each file?
2. What do these number represent?

What is the number of cases in the **PR** file?

ANSWER: Cases= 37,673

This represents the number of people in the recode file

What is the number of cases in the **KR** file?

ANSWER: Cases=5,968

This represents the number of children under five years of age in the recode file

Exercise 3

Step 1.b: For the HR file, determine which weight variable you need to use and write it down below.

In the HR file we need to use hv005.

Step 1.c: Determine which number you need to divide the variable by and write it below.

One million or 1000000.

Step 2.a: Which variable should you use to examine the region?

We should use the variable hv024 to examine region.

Step 3: Extra challenge exercise

1. Which dataset should you use? 2- Which variable for urban/rural?

ZZIR62FL

v025 for urban/rural

2. Which variable should account for weighting of the data?

v005 for weight, divided by
1000000

Step 4


```
cd "C:\Users\40825\Desktop\DHS Fellows\Data"
```

```
use ZZIR62FL.DTA, clear
```

```
*Unweighted frequency of urban/rural
```

```
tab v025
```

```
*Create IR weight variable
```

```
gen wt= v005/1000000
```

```
*Weighted frequency of urban/rural
```

```
tab v025 [iweight=wt]
```

Exercise 4

Before you Begin:

Challenge!

I: Which data file do you need to use?

```
IR File  
ZZIR62FL.DTA
```

2: Which weight?

```
v005/1000000
```

3: Which variable do you use?

```
v312
```

Exercise 5

Before you Begin:

Challenge!

I: Which data file do you need to use?

```
IR File  
ZZIR62FL.DTA
```

2: Which weight?

```
v005/1000000
```

3: Which variable do you use?

v384a

Exercise 8

Step 1

v025	Residence
v024	Region

Step 2

```
cd "C:\Users\40825\Desktop\DHS Fellows\Data"
use ZZIR62FL.DTA, clear
*Generate the IR weight variable
gen wt= v005/1000000

*Set the survey design
svyset [pw=wt], psu(v021) strata(v022) singleunit(centered)

*Residence and region
svy: tab v025 v024

*Compared to addition of the row syntax
svy:tab v025 v024 , row

*Different format when displayed as columns
svy: tab v024 v025 [iweight=wt], col
```

Challenge!

*Cross tabulation of mcpr with place of residence

Svy: tab v025 mcpr, row per

*the tabulation above is by row. We would want to know the percentage of women in urban areas that use modern methods compared to the percentage of women in rural areas that use modern methods.

*exporting the crosstabulation of several variables with mcpr and with confidence intervals

tabout v025 v024 v190 v106 mcpr [iw=wt] using "Crosstab.xls", c(row ci) f(1) stats(chi2) svy nwt(wt) per pop ptotal(none) replace

*again we use row and add ci for confidence intervals. The addition of the stats(chi2) option allows us to see the significance of the associations. As the output shows, all of the variables are significantly associated with modern contraceptive use.

Exercise 9

Example I: What variables do you need to run this syntax?

Residence:

hv025

Region:

hv024

Challenge

1. What is the denominator?

36,610

2. What does “de facto household population” mean in the subheading of this table?

De facto is a Latin term for “in fact” or “in practice.” Within the context of the DHS survey, de facto refers to people who have stayed in the household the night before the survey, compared to all who are classified as usual household residents (de jure – the Latin term for “in law”).

3. What data file do we need to use?

PR File
ZZPR62FL.DTA

4. Which weight?

hv005/1000000

5. What variable do we need to use for the age of household members?

hv105

6. Now that we know the age of the household members, which members are we interested in exactly? Based on the sub-heading, we are interested in the de facto household members, so those who “slept [in the household] last night”.

hv103

7. Put all this together to try to match the denominator of this table.

```
cd "C:\Users\40825\Desktop\DHS Fellows\Data"
```

```
use ZZPR62FL.DTA, clear
```

```
*Generate the PR weight variable
```

```
gen wt = hv005/1000000
```

```
tab hv105 if hv103==1 [iweight=wt], missing
```