```
clear

log using "D:\jason\workshop\Constructing variables with Stata\constructing variables with stata.log", replace

/*******************************************************************************
This command file was crated on 1/31/21 to demonstrate how to constructd
the marriage history for respondents

The command file: constructing variables with Stata.do
The log file:     contructing variables with Stata.log
The original data file: D:\jason\workshop\Constructing variables with Stata\original.dta"
The outcome data file : D:\jason\workshop\Constructing variables with Stata\final.dta"


The command files complete the following tasks

#1: check what variables are in the data, how many observation are in the data,
whether the data are in long or wide format, whether some of the variables are string variables

#2 Generate numeric variables for inter_y and inter_m and replace the
impossible values of these two variables.

#3: check why there is only six valid observations for the end date of the secon marriage

#4: check if the number of valid observations for the start and end of each marriage is correct.


#5: Reshape the data and drop empty records, so it is easier to check for inconsistencies across marriages


# 6: Using the interview date to fill the missing end date of continuous marriages


#7:  check if the interview date occurred before the start or end of the marriage


#8: check if some marriages ended before they started

#9: check if some marriages are out of the temporal order

#10: check if marriage overlaps with each other

*******************************************************************************/


**********************
* Read in the data
**********************


use "D:\jason\workshop\Constructing variables with Stata\original.dta", clear
save "D:\jason\workshop\Constructing variables with Stata\temp1.dta", replace

/*****************************************************************************
Check #1: what variables are in the data, how many observation are in the data,
whether the data are in long or wide format, whether some of the variables are string variables
*****************************************************************************/
count
duplicates report id

des
sum
tab1 inter_y - mar_num, mis


/*****************************************************************************
Check #2: Generate numeric variables for inter_y and inter_m and replace the
impossible values of these two variables.
*****************************************************************************/

destring inter_y, gen(inter_y1)
destring inter_m, gen(inter_m1)


des inter_y inter_y1 inter_m inter_m1


tab2 inter_y inter_y1, mis nol
tab2 inter_m inter_m1, mis nol

label variable inter_y1 "Interview Year as a numberic variable"
label variable inter_m1 "Interview Month as a numberic variable"

label value inter_y1 year
```

```
label value inter_m1 month


/*****************************************************************
replacing impossible values in inter_y1 and inter_m1
*****************************************************************/

list id  inter_y inter_y1 inter_m inter_m1 if inter_y1 ==2012
replace inter_y1 = 2021 if id ==7 & inter_y1 ==2012
list id  inter_y inter_y1 inter_m inter_m1 if id ==7, nol
tab2 inter_y inter_y1, mis nol



list id  inter_y inter_y1 inter_m inter_m1 if inter_m1 ==14
replace inter_m1 = 1 if id ==8 & inter_m1 ==14
list id  inter_y inter_y1 inter_m inter_m1 if id ==8, nol

tab2 inter_m inter_m1, mis nol



/*****************************************************************************
Identify the correct number of valid observations for the start and end of each marraige
based on the cross-tab of mar_num and marital

There should be 14 valid start dates for the first marraige, 10 for the second
marriage, and 4 for the third marriage.

There should be 13 valid end dates for the first marraige, 7 for the second marriage,
and 3 for the third marriage.
*****************************************************************************/

tab2 mar_num marital, mis nol
sum mar_sy1 mar_sm1 mar_ey1 mar_em1 mar_sy2 mar_sm2 mar_ey2 mar_em2  mar_sy3 ///
mar_sm3 mar_ey3 mar_em3


/*****************************************************************************
* Check #3: check why there is only six valid observations for the end date of the secon marriage
* One respondent (id ==12) did not provide valid end date for the second marriage.

*****************************************************************************/

list id mar_num marital mar_sy1 mar_sm1 mar_ey1 mar_em1 mar_sy2 mar_sm2 mar_ey2 ///
mar_em2  mar_sy3 mar_sm3 mar_ey3 mar_em3 if (mar_num ==2 & marital ==0) | mar_num ==3, nol


*********************************************************
* Check #4.1:Check for respondents married once and currently married
*********************************************************

tab1 mar_sy1 mar_sm1 mar_ey1 mar_em1 if mar_num == 1 & marital ==1, mis


*************************************************************
* Check #4.2: Check for respondents married twice and currently not married
*************************************************************

tab1 mar_sy1 mar_sm1  mar_ey1 mar_em1 if mar_num == 1 & marital !=1, mis


*********************************************************
* Check #4.3:Check for respondents married twice and currently married
*********************************************************
tab1 mar_sy2 mar_sm2 mar_ey2 mar_em2 if mar_num == 2 & marital ==1, mis


*************************************************************
* Check #4.4:Check for respondents married once and currently not married
*************************************************************

tab1 mar_sy2 mar_sm2 mar_ey2 mar_em2 if mar_num == 2 & marital !=1, mis


*********************************************************
* Check #4.5:Check for respondents married three times and currently married
*********************************************************

tab1 mar_sy3 mar_sm3 mar_ey3 mar_em3 if mar_num == 3 & marital ==1, mis

*************************************************************
* Check #4.6:Check for respondents married three times and currently not married
*************************************************************
```

```
tab1 mar_sy3 mar_sm3 mar_ey3 mar_em3 if mar_num == 3 & marital !=1, mis


*********************************************************************
* check #5: Reshape the data and drop empty records, so it is easier to check for inconsistencies across marriages
*********************************************************************

reshape long mar_sy mar_sm mar_ey mar_em , i(id) j(marriage)
des

label variable marriage "the order of variable, based on the respondent's report"

label variable mar_sy "starting year of marriage"
label variable mar_sm "starting year of marriage"
label variable mar_ey "ending year of marriage"
label variable mar_em "ending year of marriage"


*****************************************************
* drop empty records created by the reshape command
*****************************************************

list, nol sepby(id)
keep if marriage <= mar_num
list, nol sepby(id)




*************************************************************************************************
* check 6: Using the interview date to fill the missing end date of continuous marriages
*************************************************************************************************

tab2 mar_num marital, mis nol


list id inter_y1 inter_m1 mar_num marital marriage mar_sy mar_sm mar_ey mar_em if marital ==1, nol sepby(id)
replace mar_ey = inter_y1 if marital ==1 & mar_ey ==.
replace mar_em = inter_m1 if marital ==1 & mar_em ==.
list id inter_y1 inter_m1 mar_num marital marriage mar_sy mar_sm mar_ey mar_em if marital ==1, nol sepby(id)



*****************************************************
* Generate the century months of the date variables
*****************************************************


gen inter_cm = (inter_y1 -1900) *12 + inter_m1
gen mar_scm = (mar_sy -1900)*12 + mar_sm
gen mar_ecm = (mar_ey -1900)*12 + mar_em


*************************
* look at the data
*************************


list, nol sepby(id)

list id mar_num marital marriage inter_y1 inter_m1 mar_sy mar_sm mar_ey mar_em inter_cm  mar_scm mar_ecm , nol sepby(id)



*********************************************************************************
* check #7:  check if the interview date occurred before the start or end of the marriage
*********************************************************************************

gen check7_1 = inter_cm - mar_scm
gen check7_2 = inter_cm - mar_ecm


tab1 check7_1 check7_2, mis



*****************************************************
* Two possible ways to solve the inconsistenties
* (1) treat it as an invalid observation
* (2) treat it as the year 2020 was miscoded as the year 2021
* I chose the first way
*****************************************************
```

```
list id inter_y1 inter_m1 mar_num marital marriage mar_sy mar_sm mar_ey mar_em inter_cm mar_scm mar_ecm check7_2 if chec

replace mar_ey =.   if id ==10 & check7_2 ==-9
replace mar_em =.   if id ==10 & check7_2 ==-9
replace mar_ecm =.  if id ==10 & check7_2 ==-9
replace check7_2 =. if id ==10 & check7_2 ==-9

list id inter_y1 inter_m1 mar_num marital marriage mar_sy mar_sm mar_ey mar_em inter_cm mar_scm mar_ecm check7_2 if id =




**************************************************
* check #8: check if some marriages ended before they started
**************************************************

gen check8 = mar_ecm-mar_scm
tab1 check8, mis

list id inter_y1 inter_m1 mar_num marital marriage mar_sy mar_sm mar_ey mar_em mar_scm mar_ecm check8 if check8 < 0, sep

list id inter_y1 inter_m1 mar_num marital marriage mar_sy mar_sm mar_ey mar_em mar_scm mar_ecm check8 if check8 ==., sep



*******************************************************************************
* I chose to recode the end date of marriage to missing for this problematic record
*******************************************************************************

list id inter_y1 inter_m1 mar_num marital marriage mar_sy mar_sm mar_ey mar_em mar_scm mar_ecm check8 if id ==9, sepby(i

replace mar_ey =.     if id ==9 & check8 ==-2
replace mar_em =.     if id ==9 & check8 ==-2
replace mar_ecm =.    if id ==9 & check8 ==-2
replace check8 =.     if id ==9 & check8 ==-2

list id inter_y1 inter_m1 mar_num marital marriage mar_sy mar_sm mar_ey mar_em mar_scm mar_ecm check8 if id ==9, sepby(i



**************************************
* check #9: check if some marriages are out of the temporal order
**************************************


sort id mar_scm mar_ecm
by id: gen mar_order=_n

label variable mar_order "the order of variable, based on the order of marriages' dates"

tab2 marriage mar_order, mis

gen check9_temp= 1 if marriage ~= mar_order
by id: egen check9 = max(check9_temp)


list id inter_y1 inter_m1 mar_num marital marriage mar_order mar_sy mar_sm mar_ey mar_em mar_scm mar_ecm check9_temp che


*****************************************
* check #10: check if marriage overlaps with each other
*****************************************

gen check10_1 = 1 if mar_order ~=1 &  mar_scm[_n] ~=. &  mar_scm[_n-1] ~=. & (mar_scm[_n] <= mar_scm[_n-1])
gen check10_2 = 1 if mar_order ~=1 &  mar_scm[_n] ~=. &  mar_ecm[_n-1] ~=. & (mar_scm[_n] <= mar_ecm[_n-1])
gen check10_3 = 1 if mar_order ~=1 &  mar_ecm[_n] ~=. &  mar_scm[_n-1] ~=. & (mar_ecm[_n] <= mar_scm[_n-1])
gen check10_4 = 1 if mar_order ~=1 &  mar_ecm[_n] ~=. &  mar_ecm[_n-1] ~=. & (mar_ecm[_n] <= mar_ecm[_n-1])



list id marriage mar_order mar_sy mar_sm mar_ey mar_em mar_scm mar_ecm check10* if check10_1 ==1 | check10_2 ==1 | check

list id marriage mar_order mar_sy mar_sm mar_ey mar_em mar_scm mar_ecm check10* if id ==13, sepby(id) nol

replace mar_sy =. if id ==13 &  marriage ==2
replace mar_sm =. if id ==13 &  marriage ==2
replace mar_ey =. if id ==13 &  marriage ==2
replace mar_em =. if id ==13 &  marriage ==2

replace mar_scm =. if id ==13 &  marriage ==2
replace mar_ecm =. if id ==13 &  marriage ==2

replace check10_2 =. if id ==13 &  marriage ==2
replace check10_4 =. if id ==13 &  marriage ==2
```

```
list id marriage mar_order mar_sy mar_sm mar_ey mar_em mar_scm mar_ecm check10* if id ==13, sepby(id) nol

****************************************
* Reshape the data back to the wide format
****************************************

keep id marriage inter_y inter_y1 inter_m inter_m1 mar_sy mar_sm mar_ey mar_em marital mar_num mar_order

reshape wide marriage mar_sy mar_sm mar_ey mar_em, i(id) j( mar_order)

list id inter_y1 inter_m1 mar_sy1 mar_sm1 mar_ey1 mar_em1 mar_sy2 mar_sm2 mar_ey2 mar_em2 mar_sy3 mar_sm3 mar_ey3 mar_em

save "D:\jason\workshop\Constructing variables with Stata\final.dta", replace

log close
```