

# Successful Ways to Add and Drop Data, While Also Reformatting Data

Adetosoye Oladokun

## ABSTRACT

For my project my goal is to go through the process of explaining how to write codes in SAS ® 9.4. The main codes of focus for this project will be how to drop variables, and reformat variables. Besides that there will be codes that discuss how I uploaded my data set, created output for my data set and also how I used various frequency tables. I have highlighted the areas that contain codes, procedures statements, log statements, and output statements. To differentiate between the highlighted areas, I will put the heading in bold letters. I will also provide a brief explanation so that will act as a better guide.

## INTRODUCTION

The first step I did in this process was to import my data set in SAS 9.4. Below is the code I used to import my data (UCLA, 2015).

**MY PROCEDURE CODE IS HIGHLIGHTED IN YELLOW.**

```
proc import out= ade
datafile= "C:\Users\Guest\Desktop\ade updated dataset 2.xlsx"
dbms=xlsx REPLACE ;
getnames=yes;
run;
```

Below is the log that shows the procedure was correct. Checking the log is important when you don't have a step that produces an output. This is a way to check before you move on. Below is how my log read for my above mentioned code.

**LOG EXPLANATION WILL BE IN GREEN**

```
NOTE: The import data set has 110 observations and 16 variables.
NOTE: WORK.ADE data set was successfully created.
NOTE: PROCEDURE IMPORT used (Total process time):
      real time           0.03 seconds
      cpu time            0.03 seconds
```

An alternative to having to looking at the log would be to run a procedure such as PROC contents. I ran a PROC contents because I wanted an output of the dataset. Having an output is something that I like to have along with the log. PROC contents will provide a description of the variables in your dataset. The output shows you what variables are characters (Char), which variables are numeric (Num), and the labels for each variable. (SAS , CONTENTS, 2015)

My code for the PROC content is in light blue; Output section is also indicated by light blue

CODE: PROC contents data=ade; RUN;

OUTPUT, Figure 1:

The SAS System			
The CONTENTS Procedure			
Data Set Name	WORK.ADE	Observations	110

&lt;Successful Ways to Add and Drop Data, While Also Reformatting Data&gt;, continued

<b>Member Type</b>	DATA	<b>Variables</b>	16
<b>Engine</b>	V9	<b>Indexes</b>	0
<b>Created</b>	07/27/2015 00:52:27	<b>Observation Length</b>	160
<b>Last Modified</b>	07/27/2015 00:52:27	<b>Deleted Observations</b>	0
<b>Protection</b>		<b>Compressed</b>	NO
<b>Data Set Type</b>		<b>Sorted</b>	NO
<b>Label</b>			
<b>Data Representation</b>	WINDOWS_32		
<b>Encoding</b>	wlatin1 Western (Windows)		

**Engine/Host Dependent Information**

<b>Data Set Page Size</b>	65536
<b>Number of Data Set Pages</b>	1
<b>First Data Page</b>	1
<b>Max Obs per Page</b>	409
<b>Obs in First Data Page</b>	110
<b>Number of Data Set Repairs</b>	0
<b>ExtendObsCounter</b>	YES
<b>Filename</b>	C:\Users\Guest\AppData\Local\Temp\SAS Temporary Files\_TD4312_IPHADMINISTRATO\_ade.sas7bdat
<b>Release Created</b>	9.0401M1
<b>Host Created</b>	W32_7PRO

**Alphabetic List of Variables and Attributes**

#	Variable	Type	Len	Format	Informat	Label
4	birth_city	Char	15	\$15.	\$15.	birth_city

<Successful Ways to Add and Drop Data, While Also Reformatting Data>, continued

### Alphabetic List of Variables and Attributes

#	Variable	Type	Len	Format	Informat	Label
6	birth_country	Char	3	\$3.	\$3.	birth_country
7	birth_date	Char	10	\$10.	\$10.	birth_date
5	birth_state	Char	2	\$2.	\$2.	birth_state
8	college	Num	8	BEST.		college
11	draft_pick	Num	8	BEST.		draft_pick
10	draft_round	Num	8	BEST.		draft_round
9	draft_team	Char	20	\$20.	\$20.	draft_team
12	draft_year	Num	8	BEST.		draft_year
2	first_name	Char	7	\$7.	\$7.	first_name
3	last_name	Char	12	\$12.	\$12.	last_name
1	name	Char	18	\$18.	\$18.	name
13	position	Char	15	\$15.	\$15.	position
14	weight	Num	8	BEST.		weight
16	year_end	Num	8	BEST.		year_end
15	year_start	Num	8	BEST.		year_start

### SECTION: DROP=STATEMENT USAGE

The reason I wanted to write a code for dropping data is this was something I became comfortable with when I initially learned SAS. The Syntax I am utilizing is DROP=variable. You are able to list one or more variable names. DROP= statements are used in the DATA steps and PROC steps. If the option is associated with an input data set, the variables are not available for processing. If the DROP=data set option is associated with an output data set, SAS does not write the variables to the output data set, but they are available for processing. It is important to make note that there is a difference between a DROP statement and a DROP=data set option statement. (SAS, DROP, 2015)

- The DROP= data set option differs from the DROP statement in these ways
- In DATA steps, the DROP= data set option can apply to input and output data sets. The DROP statement applies only to output data sets
- In DATA steps, when you create multiple output data sets, use the DROP=data set option to write different variables to different data sets. The DROP statement applies to all output data sets.
- In PROC steps, you can use only the DROP=data set option, not the DROP statement.(SAS, DROP, 2015)

## &lt;Successful Ways to Add and Drop Data, While Also Reformatting Data&gt;, continued

My next part will include the code I utilized for my DROP=statement. Along with that I will show a code utilized to show output for the changes I made to my data set.

## CODE FOR DROP=VARIABLE IS YELLOW

```
PROC contents data=ade; RUN;
data adel(drop = draft_pick draft_round draft_year year_end year_start
first_name
last_name name birth_city birth_country birth_date );
set ade;
run;
```

The green highlighted code show to produce a code that will present the five variables that weren't dropped

```
CODE: PROC CONTENTS = ADE1; RUN;
```

Pink highlights show the output the for the codes ran

## OUTPUT, Figure 2:

The SAS System			
The CONTENTS Procedure			
Data SetName	WORK.ADE1	Observations	110
Member Type	DATA	Variables	5
Engine	V9	Indexes	0
Created	07/27/2015 00:58:27	Observation Length	56
Last Modified	07/27/2015 00:58:27	Deleted Observations	0
Protection		Compressed	NO
Data SetType		Sorted	NO
Label			
DataRepresentation	WINDOWS_32		
Encoding	wlatin1 Western (Windows)		
Engine/Host Dependent Information			
Data Set Page Size	65536		
Number of Data Set Pages	1		
First Data Page	1		

<Successful Ways to Add and Drop Data, While Also Reformatting Data>, continued

### Engine/Host Dependent Information

<b>Max Obs per Page</b>	1167
<b>Obs in First Data Page</b>	110
<b>Number of Data Set Repairs</b>	0
<b>ExtendObsCounter</b>	YES
<b>Filename</b>	C:\Users\Guest\AppData\Local\Temp\SAS Temporary Files\_TD4312_IPHADMINISTRATO\_ade1.sas7bdat
<b>Release Created</b>	9.0401M1
<b>Host Created</b>	W32_7PRO

### Alphabetic List of Variables and Attributes

#	Variable	Type	Len	Format	Informat	Label
1	birth_state	Char	2	\$2.	\$2.	birth_state
2	college	Num	8	BEST.		college
3	draft_team	Char	20	\$20.	\$20.	draft_team
4	position	Char	15	\$15.	\$15.	position
5	weight	Num	8	BEST.		weight

After viewing the output, this lets me know my DROP=data set option, dropped the unwanted variables and kept the variables I will need.

### FORMATTING SECTION

The FORMAT statement can use standard SAS formats or user-written formats that have been previously defined in PROC FORMAT. A single FORMAT statement can associate the same format with several variables, or it can associate different formats with different variables. If a variable appears in multiple FORMAT statements, SAS uses the format that is assigned last. You use a FORMAT statement in the DATA step to permanently associate a format with a variable. SAS changes the descriptor information of the SAS data set that contains the variable. You can use a FORMAT statement in some PROC steps, but the rules are different (SAS, FORMAT, 2015).

This is the code for my format of the variable weight into a four levels. This also includes a code utilized to produce a frequency table of the weight in the four levels. Format code is yellow. The output code is blue.

```
data ADE2;
set ADE1;
```

```
WEIGHT2= .;
```

<Successful Ways to Add and Drop Data, While Also Reformatting Data>, continued

```
if 145 <= weight <= 199 then WEIGHT2= 1;
if 200 <= weight <= 249 then weight2= 2;
if 250 <= weight <= 299 then weight2= 3;
if 300 <= weight <= 450 then weight2= 4;
```

This secondary code is utilized to create a new frequency table of the new label weight2

```
label weight2= "weight";
run;
proc freq data=ade2;
tables weight weight2;
run;
```

This is beginning of the OUPUT, Figure 3 for codes ran above for formatting:

The SAS System
----------------

The FREQ Procedure

**weight**

weight	Frequency	Percent	Cumulative Frequency	Cumulative Percent
145	1	0.91	1	0.91
165	1	0.91	2	1.82
168	1	0.91	3	2.73
169	1	0.91	4	3.64
170	1	0.91	5	4.55
172	1	0.91	6	5.45
175	2	1.82	8	7.27
178	1	0.91	9	8.18
180	1	0.91	10	9.09
182	2	1.82	12	10.91
183	1	0.91	13	11.82
184	1	0.91	14	12.73
185	2	1.82	16	14.55
187	1	0.91	17	15.45
189	1	0.91	18	16.36

&lt;Successful Ways to Add and Drop Data, While Also Reformatting Data&gt;, continued

<b>weight</b>				
<b>weight</b>	<b>Frequency</b>	<b>Percent</b>	<b>Cumulative Frequency</b>	<b>Cumulative Percent</b>
<b>190</b>	3	2.73	21	19.09
<b>193</b>	2	1.82	23	20.91
<b>194</b>	3	2.73	26	23.64
<b>195</b>	5	4.55	31	28.18
<b>197</b>	2	1.82	33	30.00
<b>198</b>	3	2.73	36	32.73
<b>200</b>	1	0.91	37	33.64
<b>204</b>	1	0.91	38	34.55
<b>205</b>	2	1.82	40	36.36
<b>207</b>	1	0.91	41	37.27
<b>208</b>	1	0.91	42	38.18
<b>209</b>	1	0.91	43	39.09
<b>210</b>	2	1.82	45	40.91
<b>212</b>	1	0.91	46	41.82
<b>213</b>	3	2.73	49	44.55
<b>214</b>	2	1.82	51	46.36
<b>215</b>	5	4.55	56	50.91
<b>218</b>	1	0.91	57	51.82
<b>220</b>	4	3.64	61	55.45
<b>223</b>	1	0.91	62	56.36
<b>225</b>	5	4.55	67	60.91
<b>226</b>	1	0.91	68	61.82
<b>227</b>	1	0.91	69	62.73
<b>228</b>	1	0.91	70	63.64

&lt;Successful Ways to Add and Drop Data, While Also Reformatting Data&gt;, continued

<b>weight</b>				
<b>weight</b>	<b>Frequency</b>	<b>Percent</b>	<b>Cumulative Frequency</b>	<b>Cumulative Percent</b>
<b>229</b>	1	0.91	71	64.55
<b>230</b>	3	2.73	74	67.27
<b>233</b>	2	1.82	76	69.09
<b>235</b>	4	3.64	80	72.73
<b>238</b>	1	0.91	81	73.64
<b>240</b>	1	0.91	82	74.55
<b>242</b>	1	0.91	83	75.45
<b>246</b>	1	0.91	84	76.36
<b>250</b>	1	0.91	85	77.27
<b>251</b>	1	0.91	86	78.18
<b>255</b>	2	1.82	88	80.00
<b>256</b>	3	2.73	91	82.73
<b>258</b>	2	1.82	93	84.55
<b>260</b>	1	0.91	94	85.45
<b>264</b>	1	0.91	95	86.36
<b>265</b>	1	0.91	96	87.27
<b>267</b>	1	0.91	97	88.18
<b>270</b>	1	0.91	98	89.09
<b>280</b>	1	0.91	99	90.00
<b>283</b>	1	0.91	100	90.91
<b>285</b>	1	0.91	101	91.82
<b>290</b>	1	0.91	102	92.73
<b>295</b>	1	0.91	103	93.64
<b>300</b>	3	2.73	106	96.36



<Successful Ways to Add and Drop Data, While Also Reformatting Data>, continued

weight				
weight	Frequency	Percent	Cumulative Frequency	Cumulative Percent
301	1	0.91	107	97.27
305	1	0.91	108	98.18
335	1	0.91	109	99.09
350	1	0.91	110	100.00

weight				
WEIGHT2	Frequency	Percent	Cumulative Frequency	Cumulative Percent
1	36	32.73	36	32.73
2	48	43.64	84	76.36
3	19	17.27	103	93.64
4	7	6.36	110	100.00

Now that I have been able to format weight from my initial observation of 110 into four levels, I now plan to run a code that changes the names of the categories. I will change 1 to equal 140-199 pounds, 2 to equal the weight range of 200-249 pounds, 3 is equal to 250-299 pounds, and 4 is equal to 300-450 pounds.

**THE YELLOW REPRESENTS THE FORMAT CODE I USED.**

```
Proc format ;
value weight2fmt 1="140-199" 2="200-249" 3="250-299" 4="300-450";
run;
```

```
data ade3;
set ade2;
format weight2 weight2fmt.; run;
```

**THIS CODE WILL PRESENT OUTPUT IN THE FORM OF A FREQUENCY TABLE OF THE NEW FORMATTED VARIABLE WEIGHT2, WHICH WAS PREVIOUSLY WEIGHT**

```
proc freq data= ade3;
tables weight2; run;
```

OUTPUT, Figure 4 for the two codes above:

<Successful Ways to Add and Drop Data, While Also Reformatting Data>, continued

### The FREQ Procedure

#### weight

WEIGHT2	Frequency	Percent	Cumulative Frequency	Cumulative Percent
140-199	36	32.73	36	32.73
200-249	48	43.64	84	76.36
250-299	19	17.27	103	93.64
300-450	7	6.36	110	100.00

## CONCLUSION

The utilization of various codes in SAS 9.4 was able to produce meaningful data. I was able to upload an excel data set that I cleaned, provide the log for data with no output, then provide output of data with Proc contents, provide the code and output for dropping and formatting data. Dropping and formatting data were the main purpose of this paper, as they very essential. It is important to note the different in a DROP and a DROP= statement, there can be confusion if a coder is not aware of which one they need at the time. A FORMAT statement helps greatly when you have a large number of observations and you want to reduce the levels of those observations.

## REFERENCE

UCLA: Statistical Consulting Group "SAS FAQ: How do I read/write Excel files in SAS?" Institute for Digital Research and Education. < <http://www.ats.ucla.edu/stat/sas/faq/rwxls8.htm> > (accessed June 24, 2015).

Statistical Analysis Software. "CONTENTS Procedure". Base SAS(R) 9.4 Procedures Guide, Fourth Edition. <<http://support.sas.com/documentation/cdl/en/proc/67916/HTML/default/viewer.htm#n1hqa4dk5tay0an15nrys1iwr5o2.htm>> (accessed June 25, 2015)

Statistical Analysis Software. "DROP=Data set option". SAS® 9.4 Data Set Options: Reference, Third Edition. <<http://support.sas.com/documentation/cdl/en/ledsoptsref/68025/HTML/default/viewer.htm#n15goor3q758g5n1eykstufkpdhy.htm>> (accessed July 11, 2015)

Statistical Analysis Software. "FORMAT Statement". SAS® 9.4 Statements: Reference, Fourth Edition. <<http://support.sas.com/documentation/cdl/en/lestmtsref/68024/HTML/default/viewer.htm#n0d5oq7e0oia0wn13nsins0x8nmh.htm>> (accessed July 15, 2015)

## CONTACT INFORMATION:

Name: Adetosoye Oladokun

Enterprise: Florida A&M University

Address: 630 West Virginia Street Apt 227

City, State ZIP: Tallahassee, FL 32304

Work Phone: 202-210-2271

Fax: N/A

Email: Adetosoye1.oladokun@fam.u.edu

Web: N/A

<Successful Ways to Add and Drop Data, While Also Reformatting Data>, continued

Trademark Citation: SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are registered trademarks or trademarks of their respective companies.