Hands-on Workshop: SAS® Data Cleansing Challenge

Course Notes

Hands-on Workshop: SAS® Data Cleansing Challenge Course Notes was developed by Peter Styliadis. Editing and production support was provided by the Curriculum Development and Support Department.

Hands-on Workshop: SAS® Data Cleansing Challenge Course Notes

Book code E71220, course code S18HDCC, prepared date 12Mar2018. S18HDCC_001

Table of Contents

To learn more…

For information about other courses in the curriculum, contact the SAS Education Division at 1-800-333-7660, or send e-mail to training@sas.com. You can also find this information on the web at http://support.sas.com/training/ as well as in the Training Course Catalog.

For a list of SAS books (including e-books) that relate to the topics covered in this course notes, visit https://www.sas.com/sas/books.html or call 1-800-727-0025. US customers receive free shipping to US addresses.

Chapter 1 SAS® Data Cleansing Challenge

1.1 Introduction

Scenario

You are a SAS programmer for the National Oceanic and Atmospheric Administration (NOAA). Your job is to prepare clean data sets for the analysts on your team.

For your next project, you are tasked with cleaning the **Earthquakes_dirty** SAS data set. After you clean the data set, you are to create an **Earthquakes_valid** data set that can be used in analysis and an **Invalid** data set that will be sent back to the research team. Your manager has provided you with the documentation needed to complete the project.

Resources

All the resources that you need for the challenge are in C:\Workshop\Challenge\DataCleansing. The folder contains the following:

- **Earthquakes_dirty** SAS data set
- **Cleansing Challenge** SAS program
- **Cleansing Challenge Backup** SAS program
- **Challenge Hints** HTML file (contains links to SAS documentation)
- **Kahoot!** Trivia link

1.2 Data Layout

The table contains information about earthquakes from 1950 to the present. Below is a data dictionary to provide more detailed information about the variables in the **Earthquakes_dirty** SAS data set.

Earthquakes_dirty

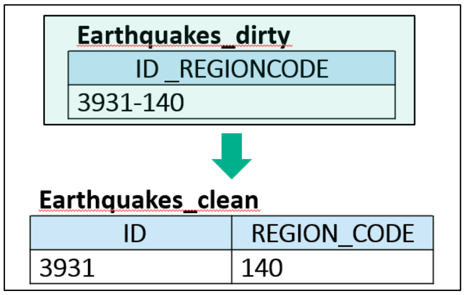
Variable	Type	Description
ID_REGIONCODE	Char	The unique ID and regional boundaries code. **ID** and **REGIONCODE** values are separated by a hyphen.
FLAG_TSUNAMI	Char	When a tsunami was generated by an earthquake, *TSU* should appear in the column. Otherwise, a blank appears.
YEAR	Num	Valid values: 1950 to present.
MONTH	Num	Valid values: 1-12.
DAY	Num	Valid values: 1-31 (where months apply).
HOUR	Num	Valid values: 0-23.
MINUTE	Num	Valid values: 0-59.
SECONDS	Num	Valid values: 0-59.
FOCAL_DEPTH	Num	The depth of the earthquake is given in kilometers. Valid values: 0 to 700 km.
EQ_MAG_MW	Num	The Mw magnitude is based on the moment magnitude scale.
EQ_MAG_MS	Num	The Ms magnitude is the surface-wave magnitude of the earthquake.
EQ_MAG_MB	Num	The Mb magnitude is the compressional body wave (P-wave) magnitude.
EQ_MAG_ML	Num	The ML magnitude was the original magnitude relationship defined by Richter and Gutenberg for local earthquakes in 1935. It is based on the maximum amplitude of a seismogram recorded on a Wood-Anderson torsion seismograph.
EQ_MAG_MFA	Num	The Mfa magnitudes are computed from the felt area, for earthquakes that occurred before seismic instruments were in general use.
EQ_MAG_UNK	Num	The computational method for the earthquake magnitude was unknown and could not be determined from the published sources.
COUNTRY	Char	The country where the earthquake occurred.
LOCATION_NAME	Char	This is an approximate geographic location.

1.3 Challenge Issues

The following requirements have been documented for you to clean the data and create the valid and invalid data sets. It is recommended you follow the requirements below to complete the challenge on time. If you are stuck, you can refer to the **Challenge Hints** section of the document for help. Start the challenge by opening the **Cleansing Challenge.sas** program.

Clean the Data

1. Create two variables from the **ID_REGIONCODE** variable. Name one variable **ID** and the other **REGION_CODE**. When you are finished, drop the original **ID_REGIONCODE** variable.



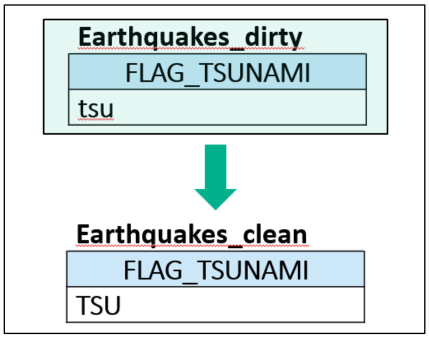
Earthquakes_dirty

ID _REGIONCODE
3931-140

Earthquakes_clean

ID	REGION_CODE
3931	140

2. All character values of **FLAG_TSUNAMI** should be converted to uppercase.



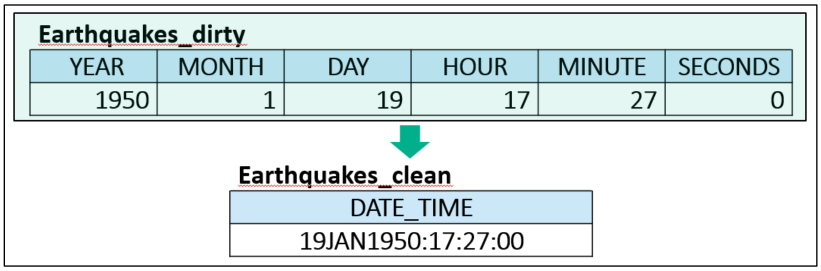
Earthquakes_dirty

FLAG_TSUNAMI
tsu

Earthquakes_clean

FLAG_TSUNAMI
TSU

3. Create a SAS datetime variable named **DATE_TIME** from the date and time variables in the **Earthquakes_dirty** data set. When you are finished, drop the original variables and format **DATE_TIME** with the SAS DATETIME format (for example, 19JAN1950:17:27:00).



Earthquakes_dirty

YEAR	MONTH	DAY	HOUR	MINUTE	SECONDS
1950	1	19	17	27	0

Earthquakes_clean

DATE_TIME
19JAN1950:17:27:00

4. Create the **EQ_PRIMARY** variable to determine the primary earthquake magnitude. There are several scales for measuring earthquake magnitudes. To determine the **EQ_PRIMARY** magnitude, choose the first nonmissing value from the available measurement variables in this order:

   a. **EQ_MAG_MW**

   b. **EQ_MAG_MS**

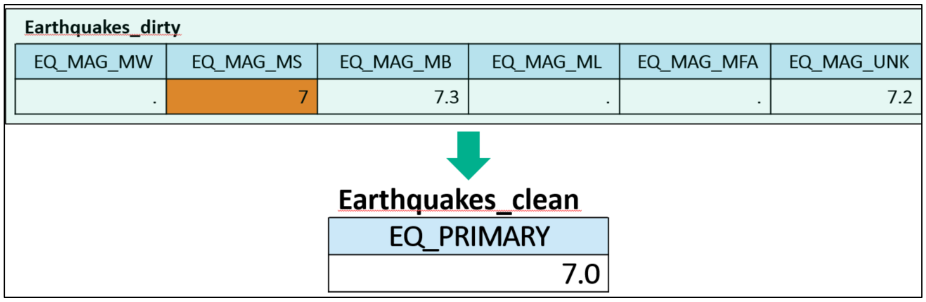
   c. **EQ_MAG_MB**

   d. **EQ_MAG_ML**

   e. **EQ_MAG_MFA**

   f. **EQ_MAG_UNK**

   When you are finished, drop the original measurement variables and format **EQ_PRIMARY** with one decimal point.



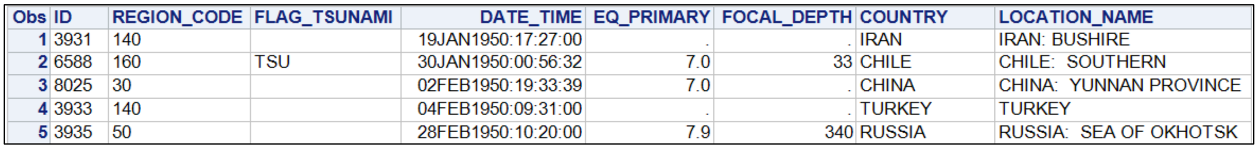
Earthquakes_dirty

EQ_MAG_MW	EQ_MAG_MS	EQ_MAG_MB	EQ_MAG_ML	EQ_MAG_MFA	EQ_MAG_UNK
.	7	7.3	.	.	7.2

Earthquakes_clean

EQ_PRIMARY
7.0

Here is a partial display capture of the desired **Earthquakes_clean** data set:



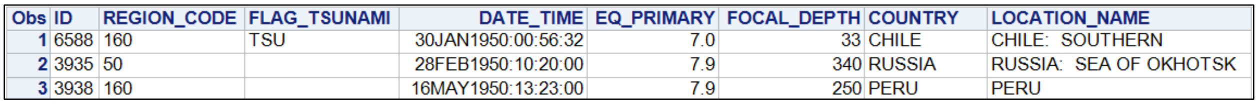
Obs	ID	REGION_CODE	FLAG_TSUNAMI	DATE_TIME	EQ_PRIMARY	FOCAL_DEPTH	COUNTRY	LOCATION_NAME
1	3931	140		19JAN1950:17:27:00	.	.	IRAN	IRAN: BUSHIRE
2	6588	160	TSU	30JAN1950:00:56:32	7.0	33	CHILE	CHILE: SOUTHERN
3	8025	30		02FEB1950:19:33:39	7.0	.	CHINA	CHINA: YUNNAN PROVINCE
4	3933	140		04FEB1950:09:31:00	.	.	TURKEY	TURKEY
5	3935	50		28FEB1950:10:20:00	7.9	340	RUSSIA	RUSSIA: SEA OF OKHOTSK

Determine Valid and Invalid Observations

Create two data sets: an **Earthquakes_valid** data set for all valid observations, and an **Invalid** data set for all invalid observations. Use the information below to determine what constitutes a valid observation.

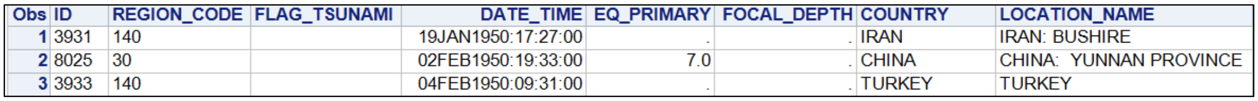
- **ID** – Unique identifier of the record. No duplicates.
- **REGION_CODE** – Valid values: *1*, *10*, *15*, *20*, *30*, *40*, *50*, *60*, *70*, *80*, *90*,*100*, *110*, *120*, 130, *140*, *150*, *160*, *170*
- **FLAG_TSUNAMI** – Valid values: blank, *TSU*
- **DATE_TIME** – Valid values: nonmissing
- **EQ_PRIMARY** – Valid values: 0.0 – 9.9
- **FOCAL_DEPTH** – Valid values: 0 – 700

Partial **Earthquakes_valid**



Obs	ID	REGION_CODE	FLAG_TSUNAMI	DATE_TIME	EQ_PRIMARY	FOCAL_DEPTH	COUNTRY	LOCATION_NAME
1	6588	160	TSU	30JAN1950:00:56:32	7.0	33	CHILE	CHILE: SOUTHERN
2	3935	50		28FEB1950:10:20:00	7.9	340	RUSSIA	RUSSIA: SEA OF OKHOTSK
3	3938	160		16MAY1950:13:23:00	7.9	250	PERU	PERU

Partial **Invalid**

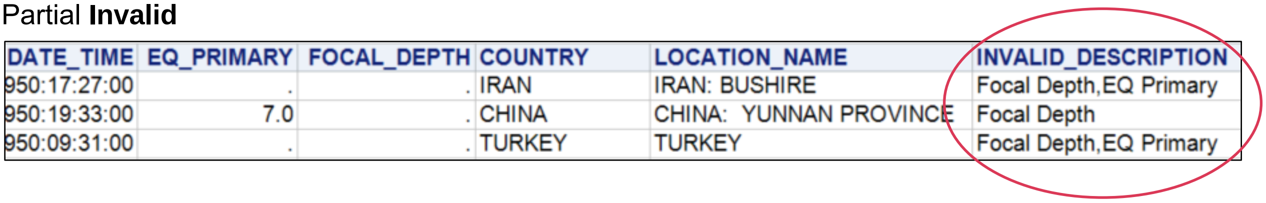


Obs	ID	REGION_CODE	FLAG_TSUNAMI	DATE_TIME	EQ_PRIMARY	FOCAL_DEPTH	COUNTRY	LOCATION_NAME
1	3931	140		19JAN1950:17:27:00	.	.	IRAN	IRAN: BUSHIRE
2	8025	30		02FEB1950:19:33:00	7.0	.	CHINA	CHINA: YUNNAN PROVINCE
3	3933	140		04FEB1950:09:31:00	.	.	TURKEY	TURKEY

Bonus

With the **Invalid** data set, create a new variable name **INVALID_DESCRIPTION** that lists the invalid variable (or variables) for each observation.

Partial **Invalid**



DATE_TIME	EQ_PRIMARY	FOCAL_DEPTH	COUNTRY	LOCATION_NAME	INVALID_DESCRIPTION
950:17:27:00	.	.	IRAN	IRAN: BUSHIRE	Focal Depth,EQ Primary
950:19:33:00	7.0	.	CHINA	CHINA: YUNNAN PROVINCE	Focal Depth
950:09:31:00	.	.	TURKEY	TURKEY	Focal Depth,EQ Primary

Validate Your Results

To validate your results, run the validation code at the bottom of the **Cleansing Challenge.sas** program when you are finished. After you run the code, check the results and fill in the answers below. These validation results are used in the end-of-class trivia.

1. What is the average magnitude for the **EQ_PRIMARY** variable in the **Earthquakes_valid** data set? ____________

2. How many earthquakes have a missing value for **DATE_TIME** in the **Invalid** data set? ____________

3. How many observations are in the **Invalid** data set? ____________

1.4 Challenge Hints

The following hints will assist you in completing the challenge. You can use the ***Challenge Hints HTML*** file in your challenge folder for direct links to the SAS documentation.

Clean the Data

1. Consider using the SCAN() or SUBSTR() function to parse the **ID_REGIONCODE** variable.
2. Consider using the UPCASE() function to standardize values of **FLAG_TSUNAMI**.
3. Consider using the DHMS() function with a nested MDY() function to create the **DATE_TIME** variable. Be sure to format **DATE_TIME** using the DATETIME*w.d* format.
4. Consider using either the COALESCE() function or the IF-THEN/ELSE statement to select the first nonmissing value in the **EQ_MAG** measurement variables.

Determine Valid and Invalid Observations

There are multiple ways to determine the valid and invalid observations. In the hint below, you determine the valid observations to write to the **Earthquakes_valid** data set, and the remaining observations are written to the **Invalid** data set. We use the IF-THEN/ELSE statement with the AND operator and with the following conditions:

- **ID**: There are multiple ways to check for a duplicate **ID** value. One way is to consider using the FREQ procedure with the ORDER= option and a TABLES statement. Find the only observation that has a frequency of 2, and use the Not Equal to comparison operator with the duplicate **ID** value to identify all valid IDs.
- **REGION_CODE:** Use the IN operator with a list of the valid region codes.
- **FLAG_TSUNAMI:** Use the IN operator with the valid values.
- **DATE_TIME:** Use the Not Equal to comparison operator to find all nonmissing date values.
- **EQ_PRIMARY:** Use comparison operators to include only the valid values.
- **FOCAL_DEPTH**: Use comparison operators to include only the valid values.

Bonus

There are multiple ways to create the **INVALID_DESCRIPTION** variable. One option is to consider using the DATA step with a LENGTH statement to create the character variable **INVALID_DESCRIPTION**. Then use an IF-THEN/ELSE statement to determine whether a variable is invalid. If the variable is invalid, use the CATX() function to append the string that identifies the invalid variable to **INVALID_DESCRIPTION**.

1.5 Suggested Answer

```
/****************************
Suggested Answer
****************************/
libname quakes "C:\Workshop\Challenge\DataCleansing";

/**********************************
Suggested Answer - Clean the Data
**********************************/
data earthquakes_clean;
   length ID $10 REGION_CODE $10 FLAG_TSUNAMI $3 DATE_TIME 8
     EQ_PRIMARY 8;
   set quakes.earthquakes_dirty;
   ID=scan(ID_RegionCode,1,'-');
   Region_Code=scan(ID_RegionCode,-1,'-');
   Flag_Tsunami=upcase(flag_tsunami);
   Date_Time=dhms(mdy(month,day,year),hour,minute,seconds);
   Eq_Primary=Coalesce(eq_mag_mw, eq_mag_ms, eq_mag_mb, eq_mag_ml,
                       eq_mag_mfa, eq_mag_unk);
   keep ID Region_Code Flag_Tsunami Date_Time EQ_Primary
        Focal_Depth Country Location_Name;
   format Date_Time datetime21. EQ_Primary 8.1;
run;

/**************************************************
Check for duplicates
**************************************************/
proc freq data=earthquakes_clean order=freq;
   tables ID;
run;

/********************************************
Suggested Answer - Valid and Invalid data sets
********************************************/
data earthquakes_valid invalid;
   set earthquakes_clean;
   if (ID ne '10301'
 and Region_code in ("1", "10", "15", "20", "30", "40", "50",
     "60", "70", "80", "90","100", "110", "120", "130", "140",
     "150", "160", "170")
 and Flag_Tsunami in ('','TSU')
 and Date_time ne .
 and (0 <= EQ_Primary <= 9.9)
 and (0 <= Focal_Depth <= 700)) then output earthquakes_valid;
   else output invalid;
run;
```

```
/********************************************
Bonus
********************************************/
data invalid;
   set invalid;
   length INVALID_DESCRIPTION $60;
   Invalid_description="";
   if ID='10301' then
      Invalid_description = catx(',','DuplicateID',
                            Invalid_description);
   if Region_code not in ("1", "10", "15", "20", "30", "40", "50",
                      "60", "70", "80", "90","100", "110", "120",
                      "130", "140", "150", "160", "170") then
      Invalid_description = catx(',','Region Code',
                            Invalid_description);
   if Flag_Tsunami not in ('','TSU') then
      Invalid_description = catx(',','Flag_Tsunami',
                            Invalid_description);
   if Date_time = . then
      Invalid_description = catx(',','Date Time',
                            Invalid_description);
   if not(0 <= Focal_Depth <= 700) then
      Invalid_description = catx(',','Focal Depth',
                            Invalid_description);
   if not(0 <= EQ_Primary <= 9.9 ) then
      Invalid_description = catx(',','EQ Primary',
                             Invalid_description);
run;

/******************************************************************
Validation Answers
******************************************************************/
1. What is the average magnitude for the EQ_PRIMARY variable in the
Earthquakes_valid data set?

6.20

2. How many earthquakes have a missing value for DATE_TIME in the
Invalid data set?";

43

3. How many observations are in the Invalid data set?

242
```