Putting the Meta into the Data: Managing Data Processing for Large Scale CDC Surveillance Projects with SAS®

Louise S. Hadden
Abt Associates Inc.

Putting the Meta into the Data: Managing Data Processing for Large Scale CDC Surveillance Projects with SAS®

Louise Hadden presented at her first SAS conference in 1996 and has never looked back, presenting at multiple conferences across the continent over the years. She supports analytic processing at Abt Associates Inc., a social science research company, and specializes in reporting and data visualization in the division of health and environment. Most recently her work portfolio has included large scale surveillance projects for the CDC and claims-based quality measure development, as well as the creation of data visualizations.

THE ENVIRONMENT

- 2 large scale, multi site CDC surveillance studies involving COVID testing and other health metrics
- Different study designs
  - Combination REDCap and EMR abstraction
  - REDCap / Serology
- Common and differing data management and SAS programming challenges
- Common and different data management and SAS programming solutions

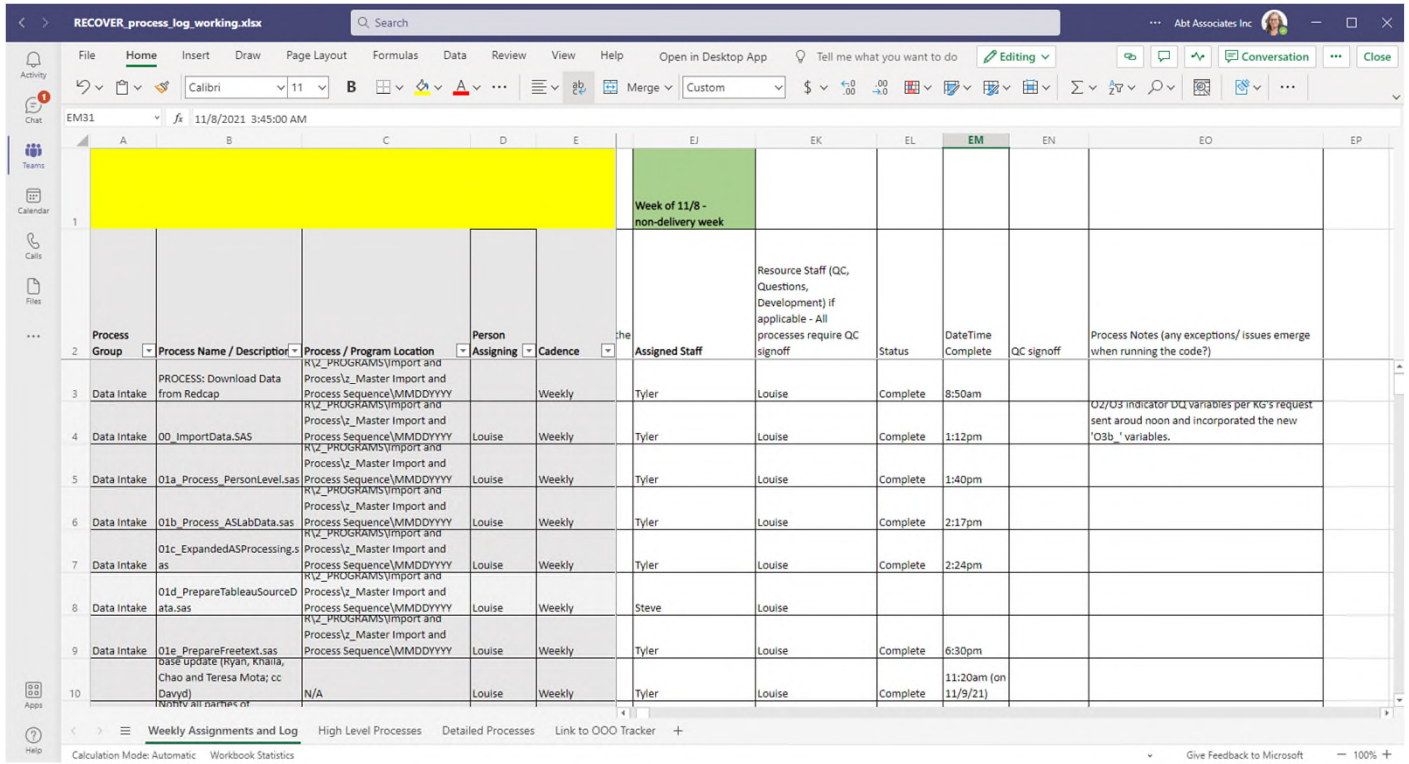
Obstacles

- Vast amount of data
  - Different sources
  - Multiple data areas
  - Multiple analytic agendas
- Quick turnaround of data deliveries!
  - Monthly on one project
  - Weekly+ on another project
- Super high visibility
  - Actionable quality assurance a must

Tools we use

- Data management tools
  - Data receipt and delivery logs
  - Shared process trackers
  - Data management and code quality plan
  - Staffing plan
  - Data dictionaries

TOOLS WE USE – SHARED PROCESS TRACKERS



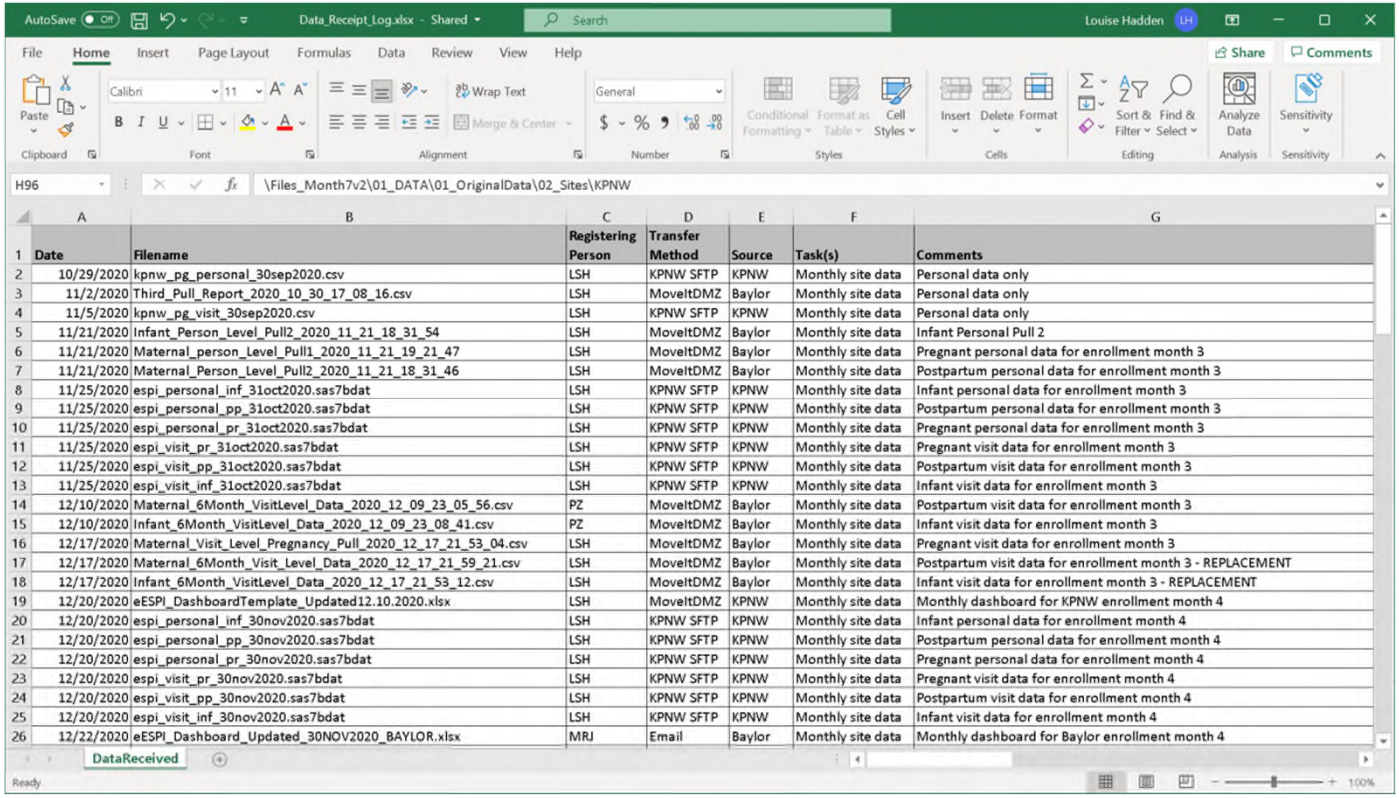
RECOVER_process_log_working.xlsx

EM31 | 11/8/2021 3:45:00 AM

	A	B	C	D	E	EJ	EK	EL	EM	EN	EO	EP
1						Week of 11/8 - non-delivery week						
2	Process Group	Process Name / Description	Process / Program Location	Person Assigning	Cadence	Assigned Staff	Resource Staff (QC, Questions, Development) if applicable - All processes require QC signoff	Status	DateTime Complete	QC signoff	Process Notes (any exceptions/ issues emerge when running the code?)	
3	Data Intake	PROCESS: Download Data from Redcap	R\Z_PROGRAMS\Import and Process\z_Master Import and Process Sequence\MMDDYYYY		Weekly	Tyler	Louise	Complete	8:50am			
4	Data Intake	00_ImportData.SAS	R\Z_PROGRAMS\Import and Process\z_Master Import and Process Sequence\MMDDYYYY	Louise	Weekly	Tyler	Louise	Complete	1:12pm		O2/O3 indicator DQ variables per KG's request sent aroud noon and incorporated the new 'O3b_' variables.	
5	Data Intake	01a_Process_PersonLevel.sas	R\Z_PROGRAMS\Import and Process\z_Master Import and Process Sequence\MMDDYYYY	Louise	Weekly	Tyler	Louise	Complete	1:40pm			
6	Data Intake	01b_Process_ASLabData.sas	R\Z_PROGRAMS\Import and Process\z_Master Import and Process Sequence\MMDDYYYY	Louise	Weekly	Tyler	Louise	Complete	2:17pm			
7	Data Intake	01c_ExpandedASProcessing.sas	R\Z_PROGRAMS\Import and Process\z_Master Import and Process Sequence\MMDDYYYY	Louise	Weekly	Tyler	Louise	Complete	2:24pm			
8	Data Intake	01d_PrepareTableauSourceData.sas	R\Z_PROGRAMS\Import and Process\z_Master Import and Process Sequence\MMDDYYYY	Louise	Weekly	Steve	Louise					
9	Data Intake	01e_PrepareFreetext.sas	R\Z_PROGRAMS\Import and Process\z_Master Import and Process Sequence\MMDDYYYY	Louise	Weekly	Tyler	Louise	Complete	6:30pm			
10		base update (Ryan, Khalia, Chao and Teresa Mota; cc Davyd) Notify all parties of	N/A	Louise	Weekly	Tyler	Louise	Complete	11:20am (on 11/9/21)			

Weekly Assignments and Log | High Level Processes | Detailed Processes | Link to OOO Tracker

Tools we use – Data receipt log



H96 \Files_Month7v2\01_DATA\01_OriginalData\02_Sites\KPNW

Date	Filename	Registering Person	Transfer Method	Source	Task(s)	Comments
10/29/2020	kpnw_pg_personal_30sep2020.csv	LSH	KPNW SFTP	KPNW	Monthly site data	Personal data only
11/2/2020	Third_Pull_Report_2020_10_30_17_08_16.csv	LSH	MoveItDMZ	Baylor	Monthly site data	Personal data only
11/5/2020	kpnw_pg_visit_30sep2020.csv	LSH	KPNW SFTP	KPNW	Monthly site data	Personal data only
11/21/2020	Infant_Person_Level_Pull2_2020_11_21_18_31_54	LSH	MoveItDMZ	Baylor	Monthly site data	Infant Personal Pull 2
11/21/2020	Maternal_person_Level_Pull1_2020_11_21_19_21_47	LSH	MoveItDMZ	Baylor	Monthly site data	Pregnant personal data for enrollment month 3
11/21/2020	Maternal_Person_Level_Pull2_2020_11_21_18_31_46	LSH	MoveItDMZ	Baylor	Monthly site data	Postpartum personal data for enrollment month 3
11/25/2020	espi_personal_inf_31oct2020.sas7bdat	LSH	KPNW SFTP	KPNW	Monthly site data	Infant personal data for enrollment month 3
11/25/2020	espi_personal_pp_31oct2020.sas7bdat	LSH	KPNW SFTP	KPNW	Monthly site data	Postpartum personal data for enrollment month 3
11/25/2020	espi_personal_pr_31oct2020.sas7bdat	LSH	KPNW SFTP	KPNW	Monthly site data	Pregnant personal data for enrollment month 3
11/25/2020	espi_visit_pr_31oct2020.sas7bdat	LSH	KPNW SFTP	KPNW	Monthly site data	Pregnant visit data for enrollment month 3
11/25/2020	espi_visit_pp_31oct2020.sas7bdat	LSH	KPNW SFTP	KPNW	Monthly site data	Postpartum visit data for enrollment month 3
11/25/2020	espi_visit_inf_31oct2020.sas7bdat	LSH	KPNW SFTP	KPNW	Monthly site data	Infant visit data for enrollment month 3
12/10/2020	Maternal_6Month_VisitLevel_Data_2020_12_09_23_05_56.csv	PZ	MoveItDMZ	Baylor	Monthly site data	Postpartum visit data for enrollment month 3
12/10/2020	Infant_6Month_VisitLevel_Data_2020_12_09_23_08_41.csv	PZ	MoveItDMZ	Baylor	Monthly site data	Infant visit data for enrollment month 3
12/17/2020	Maternal_Visit_Level_Pregnancy_Pull_2020_12_17_21_53_04.csv	LSH	MoveItDMZ	Baylor	Monthly site data	Pregnant visit data for enrollment month 3
12/17/2020	Maternal_6Month_Visit_Level_Data_2020_12_17_21_59_21.csv	LSH	MoveItDMZ	Baylor	Monthly site data	Postpartum visit data for enrollment month 3 - REPLACEMENT
12/17/2020	Infant_6Month_VisitLevel_Data_2020_12_17_21_53_12.csv	LSH	MoveItDMZ	Baylor	Monthly site data	Infant visit data for enrollment month 3 - REPLACEMENT
12/20/2020	eESPI_DashboardTemplate_Updated12.10.2020.xlsx	LSH	MoveItDMZ	KPNW	Monthly site data	Monthly dashboard for KPNW enrollment month 4
12/20/2020	espi_personal_inf_30nov2020.sas7bdat	LSH	KPNW SFTP	KPNW	Monthly site data	Infant personal data for enrollment month 4
12/20/2020	espi_personal_pp_30nov2020.sas7bdat	LSH	KPNW SFTP	KPNW	Monthly site data	Postpartum personal data for enrollment month 4
12/20/2020	espi_personal_pr_30nov2020.sas7bdat	LSH	KPNW SFTP	KPNW	Monthly site data	Pregnant personal data for enrollment month 4
12/20/2020	espi_visit_pr_30nov2020.sas7bdat	LSH	KPNW SFTP	KPNW	Monthly site data	Pregnant visit data for enrollment month 4
12/20/2020	espi_visit_pp_30nov2020.sas7bdat	LSH	KPNW SFTP	KPNW	Monthly site data	Postpartum visit data for enrollment month 4
12/20/2020	espi_visit_inf_30nov2020.sas7bdat	LSH	KPNW SFTP	KPNW	Monthly site data	Infant visit data for enrollment month 4
12/22/2020	eESPI_Dashboard_Updated_30NOV2020_BAYLOR.xlsx	MRJ	Email	Baylor	Monthly site data	Monthly dashboard for Baylor enrollment month 4

Tools we use

- Programming tools
  - Software and systems: R, REDCap, SAS, Stata, SQL server
  - Data dictionaries (CSV, Excel)
  - Control files
  - Data Step programming (import/export, read in, put out)
  - SAS procedures (FORMAT, COMPARE, CONTENTS, etc.)

SCENARIOS

- Large scale EMR abstraction based surveillance
  - Data dictionary used to provide specs for abstraction, and to drive data processing
  - Multiple sites, different EMR systems
- Large scale REDCap surveillance and serology
  - REDCap Data dictionary used to inform read-in and create formats
  - Multiple sites, different REDCaps

Scenario 1

- Large scale, multi site study involving COVID testing and other health metrics
- Data dictionaries are used as both documentation and specs for data abstraction from Electronic Medical Records, data derivation and data extraction from REDCAP
- # signs in variable names indicate an unknown number of iterations of a variable

Scenario 1

- Data files for the study require variable labels
- Variable labels need to be file specific
- A format library is required for delivery of data
- Format assignment statements are required, and need to be specific to each deliverable file
- Files are received on a monthly basis from sites and data is cumulative

Scenario 1 Solutions

- Ingest data dictionaries to collect metadata allowing identification of groups for processing
- Drive processing with information derived from data dictionaries
- SAS tools used:
  - SAS functions
  - SAS macro facility
  - Ability to write and read text files within a process
  - PROC FORMAT CNTLIN and CNTLOUT

Scenario 1 Data Dictionary

- Data dictionaries in use for the study have multiple tabs and thousands of variables
- Individual variables can be present in one or more of three separate files at two different levels for: pregnant women, post-partum women, and infants
- Individual variables have up to two # signs (iteration flags), appear in a single row in the data dictionary, and may occur 60+ times

SCENARIO 1 DATA DICTIONARY

FILE HOME INSERT PAGE LAYOUT FORMULAS DATA REVIEW VIEW SAS

Louise Hadden

J10

	A	B	C	D
1	**Visit-level file (variables pulled during the data pull that covers the time window in which the visit took place)**			
2	**4. Virus Testing (Neonate)**			
3	These variables describe the characteristics of virus testing received by the neonate at the delivery hospitalization, including dates of tests and test results. For details on iterating variables endi _NEO#_VTST#, see the Key (Guidance for variables with multiple iterations)			
4	**Variable Name**	**Variable Description**	**Variable Values**	**Notes**
5	COVID_IgM_NEO#_VTST#	First test for IgM for SARS-CoV-2 antibody (during the visit/admission)	0 = SARS-CoV-2 negative	<**Multiple Testing AND Multiple Fetuse** NEO# will iterate with each fetus/newbor iterate with each test performed on that sp fetus/newborn.
6			1 = SARS-CoV-2 positive	
7			2 = No SERUM testing	
8			888 = Missing	
9			999 = Unknown	

Key | Revisions | Identifiers | Event Description | Virus Testing | Virus Testing_Neo | ARFI and other Dx Codes | Medications_Flu | Medi ...

READY 100%

Scenario 1 Data Dictionary – Read In Goals

- Import information from data dictionaries for different purposes
  - Variable labels
  - Expected variables
  - Value labels (formats)
  - Label, format, retain, attrib statements
  - SAS procedure statements
  - Drive macro processing

Scenario 1 Data Dictionary – Read in Challenges

- Multiple tabs
- Read in specified ranges
- Clean up of special characters (tabs, carriage returns)
- Disaggregation of some fields
- Feedback loop for errors found
- Requirements for two levels
  - Variable information
  - Value Label information

Scenario 1 Data Dictionary Read In

```
*****************************************************************;
*** Import Personal Data Dictionary one tab at a time    ***;
*****************************************************************;

%macro imptabs(tabn=1, tabnm=identifiers, intab=Identifiers,
startrow=10, endcol=H);

 proc import dbms=xlsx out = temp datafile = " \file.xlsx"
 replace;
            RANGE="&intab.$A&startrow.:&endcol.999";
            getnames=YES;
run;
```

SCENARIO 1 DATA DICTIONARY READ IN

```
data labels&tabn.;
    length label labelstr $ 300 variable_type $ 8;
    set &tabnm (keep=variable_: pw_preg pw_pp inf
        where=(variable_name ne ‘’
        and variable_description ne ''));

    variable_length=length(variable_name);
    length_flag=(variable_length+7 GT 32);
    label variable_length="Length of Variable"
    length_flag="Variable Length + 7 exceeds 32";
```

Scenario 1 Data Dictionary Read In

```
/* find out the # of iterations in a variable name */

    iteration_flag=(indexc(variable_name,'#') gt 0);
    iteration_count=countc(variable_name,'#’);
    label iteration_flag="Binary: Variable iterations"
    iteration_count="# of iteration points within
    variable name";
```

Scenario 1 Data Dictionary Read In

```
data labels&tabn.;
    length label labelstr $ 300 variable_type $ 8;
    . . .
    label=catx(": ","&tabnm.",variable_description);
    labelstr=cats(variable_name,'="',label,'"');
```

ITERATION

- It is a simple task to replace a single iterator in a variable name. It is more complicated when you don't know how many there are or where they are.
- Once we locate the iterator(s), we use the SUBSTR function to replace the iterator. Note that the SUBSTR function is sensitive to the size of the replacement value.
- You can use functions to discover the number of iterations needed, including COUNTC, REVERSE, or ANYNUM.

Iteration: Simple Loop

```
%macro do_list1(maxiter=1,suffix=neo);
%do i=1 %to &maxiter;
data iter&suffix.1_&i (drop=loc);
    length variable $ 50 labelstr $ 300;
    set formats0 (where=(count(variable,"#")=1
and index(variable,"IDENTIFIER#")>0));
    *get the first indexed # location;
    loc=index(variable,"#");
    substr(variable,loc,1)="&i";
    labelstr=catt(labelstr," #&i");
run;
%END;
%MEND DO_LIST1;
```

Iteration: Complex Loop

```
%macro do_list2(maxiter=20,suffix=vtst);
%if &maxiter le 9 %then %do i=1 %to &maxiter;
data iter&suffix.1_&i (drop=loc);
    length variable $ 50 labelstr $ 300;
    set formats0 (where=(count(variable,"#")=1
and index(variable,"VTST#")>0));
    *get the first indexed # location;
    loc=index(variable,"#");
    substr(variable,loc,1)="&i";
    labelstr=catt(labelstr," #&i");
run;
```

Iteration: Complex Loop

```
%if &maxiter gt 9 %then %do;
%do i=1 %to 9;
data iter&suffix.1_&i (drop=loc);
    length variable $ 50 labelstr $ 300;
    set formats0 (where=(count(variable,"#")=1
and index(variable,"VTST#")>0));
    *get the first indexed # location;
    loc=index(variable,"#");
    substr(variable,loc,1)="&i";
    labelstr=catt(labelstr," #&i");
run;
%END;
```

Iteration: Complex Loop

```
%do i=10 %to &maxiter;
data iter&suffix.1_&i (drop=loc);
    length variable $ 50 labelstr $ 300;
    set formats0 (where=(count(variable,"#")=1
and index(variable,"VTST#")>0));
loc=index(variable,"#");
    substr(variable,loc,2)="&i";
    labelstr=catt(labelstr," #&i");
run;
%END;
%END;
%MEND DO_LIST2;
```

APPLYING THE TECHNIQUE TO CREATE LABELS

```
filename label1 ".\&short._Labels.txt";
%do_list1(maxiter=3,suffix=id);
%do_list2(maxiter=4,suffix=vtst); . . .
data expand_labels;
    set iterid: itervtst: . . ._ ;
run;
data labels;
    length variable $ 32;
    set labels0 (where=(index(variable,"#")=0))
        expand_labels
(where=(index(variable,"#")=0));
run;
```

LABEL CREATION

```
data tolabel;
    retain VARIABLE_CATEGORY ...;
    file label1 lrecl=400;
    set matchtest (keep=VARIABLE_CATEGORY ...);
    by NUM;
STATEMENT=compbl(cats(variable,'=“’,
labelstr,'"'));
    if inlabels=1 and inpos=1
    then put statement;
run;
```

Label creation

```
filename label1 ".\&short._Labels.txt";
filename retain1 ".\&short._retain.txt";
run;

data &outfi. (label="Labeled &short");
    retain
        %include retain1;
    ;
    set &infi.;
    label
        %include label1;
    ;
run;
```

LABEL CREATION

B_P_IN_7v2_Labels.txt - Notepad

File Edit Format View Help

```
IN_DATA_EXTRCT_DT="Mo 7v2: Date of data extraction"
INF_IDENTIFIER="Mo 7v2: Infant identifier #1"
IN_DOB="Mo 7v2: Date of birth"
IN_HR_AC_ENDOCRINE_PL="Mo 7v2: Acute endocrine complications"
IN_HR_AC_GASTRO_PL="Mo 7v2: Acute gastrointestinal complications"
IN_HR_AC_HEART_FAIL_PL="Mo 7v2: Acute heart failure diagnosis"
IN_HR_AC_HEART_PL="Mo 7v2: Acute heart disease diagnosis"
IN_HR_AC_HEMATO_PL="Mo 7v2: Acute hematological complications"
IN_HR_AC_LIVER_FAIL_PL="Mo 7v2: Acute liver failure diagnosis"
IN_HR_AC_NEURO_PL="Mo 7v2: Only Acute Neurologic disease diagnosis"
```

DATA DICTIONARY READ IN FOR FORMATS

FILE HOME INSERT PAGE LAYOUT FORMULAS DATA REVIEW VIEW SAS Louise Hadden

J10

	A	B	C	D
1	Visit-level file (variables pulled during the data pull that covers the time window in which the visit took place)			
2	4. Virus Testing (Neonate)			
3	These variables describe the characteristics of virus testing received by the neonate at the delivery hospitalization, including dates of tests and test results. For details on iterating variables endi _NEO#_VTST#, see the Key (Guidance for variables with multiple iterations)			
4	**Variable Name**	**Variable Description**	**Variable Values**	**Notes**
5	COVID_IgM_NEO#_VTST#	First test for IgM for SARS-CoV-2 antibody (during the visit/admission)	0 = SARS-CoV-2 negative	<**Multiple Testing AND Multiple Fetuse** NEO# will iterate with each fetus/newbor iterate with each test performed on that sp fetus/newborn.
6			1 = SARS-CoV-2 positive	
7			2 = No SERUM testing	
8			888 = Missing	
9			999 = Unknown	

Key | Revisions | Identifiers | Event Description | Virus Testing | Virus Testing_Neo | ARFI and other Dx Codes | Medications_Flu | Medi ...

READY 100%

DATA DICTIONARY READ IN FOR FORMATS

```
*****************************************************************;
*** Import Personal Data Dictionary one tab at a time    ***;
*****************************************************************;

%macro imptabs(tabn=1, tabnm=identifiers, intab=Identifiers,
startrow=10, endcol=H);

 proc import dbms=xlsx out = temp datafile = " \file.xlsx"
 replace;
            RANGE="&intab.$A&startrow.:&endcol.999";
            getnames=YES;
run;
```

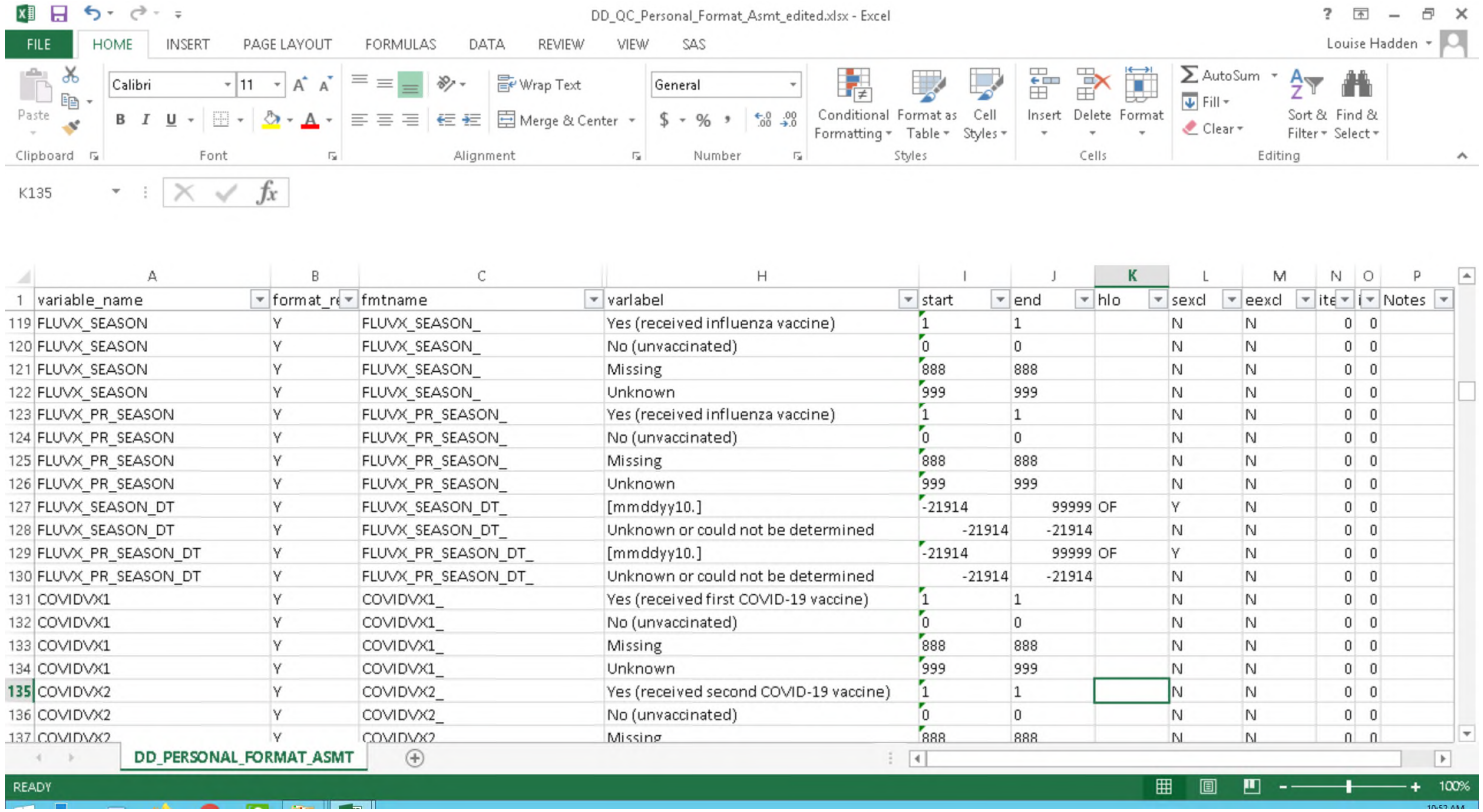
DATA DICTIONARY READ IN FOR FORMATS

```
data labels&tabn.;
    length variable_name $ 32 variable_values_edited varlabel $ 300 start $ 8
           variable_type $ 8 ;
    set &tabnm (keep=variable_: pw_preg pw_pp inf
        where=(variable_values ne '' or variable_name ne ''));

/* replace special characters such as tabs with blank and remove extraneous
blanks */
    variable_values_edited=translate(variable_values,' ','09'x);
    variable_values_edited=translate(variable_values_edited,' ','0A'x);
    variable_values_edited=translate(variable_values_edited,' ','0D'x);
    variable_values_edited=compbl(variable_values_edited);

    /* create start and label variables for a start on building formats */
    if variable_type not in('ID','DATE') then do;
        start=scan(variable_values_edited,1,"=");
        varlabel=scan(variable_values_edited,2,"=");
    end;
```

DATA DICTIONARY READ IN FOR FORMATS

```
 retain _variable_type;
    if not missing(variable_type) then
_variable_type=variable_type;
    else variable_type=_variable_type;
    drop _variable_type;

    formatstr=variable_values_edited;

    if variable_name ne '' then
fmtname=cats(variable_name,'_');
```

DATA DICTIONARY READ IN FOR FORMATS



DD_QC_Personal_Format_Asmt_edited.xlsx - Excel

	A	B	C	H	I	J	K	L	M	N	O	P
1	variable_name	format_re	fmtname	varlabel	start	end	hlo	sexcl	eexcl	ite	i	Notes
119	FLUVX_SEASON	Y	FLUVX_SEASON_	Yes (received influenza vaccine)	1	1		N	N	0	0	
120	FLUVX_SEASON	Y	FLUVX_SEASON_	No (unvaccinated)	0	0		N	N	0	0	
121	FLUVX_SEASON	Y	FLUVX_SEASON_	Missing	888	888		N	N	0	0	
122	FLUVX_SEASON	Y	FLUVX_SEASON_	Unknown	999	999		N	N	0	0	
123	FLUVX_PR_SEASON	Y	FLUVX_PR_SEASON_	Yes (received influenza vaccine)	1	1		N	N	0	0	
124	FLUVX_PR_SEASON	Y	FLUVX_PR_SEASON_	No (unvaccinated)	0	0		N	N	0	0	
125	FLUVX_PR_SEASON	Y	FLUVX_PR_SEASON_	Missing	888	888		N	N	0	0	
126	FLUVX_PR_SEASON	Y	FLUVX_PR_SEASON_	Unknown	999	999		N	N	0	0	
127	FLUVX_SEASON_DT	Y	FLUVX_SEASON_DT_	[mmddyy10.]	-21914	99999	OF	Y	N	0	0	
128	FLUVX_SEASON_DT	Y	FLUVX_SEASON_DT_	Unknown or could not be determined	-21914	-21914		N	N	0	0	
129	FLUVX_PR_SEASON_DT	Y	FLUVX_PR_SEASON_DT_	[mmddyy10.]	-21914	99999	OF	Y	N	0	0	
130	FLUVX_PR_SEASON_DT	Y	FLUVX_PR_SEASON_DT_	Unknown or could not be determined	-21914	-21914		N	N	0	0	
131	COVIDVX1	Y	COVIDVX1_	Yes (received first COVID-19 vaccine)	1	1		N	N	0	0	
132	COVIDVX1	Y	COVIDVX1_	No (unvaccinated)	0	0		N	N	0	0	
133	COVIDVX1	Y	COVIDVX1_	Missing	888	888		N	N	0	0	
134	COVIDVX1	Y	COVIDVX1_	Unknown	999	999		N	N	0	0	
135	COVIDVX2	Y	COVIDVX2_	Yes (received second COVID-19 vaccine)	1	1		N	N	0	0	
136	COVIDVX2	Y	COVIDVX2_	No (unvaccinated)	0	0		N	N	0	0	
137	COVIDVX2	Y	COVIDVX2	Missing	888	888		N	N	0	0	

DD_PERSONAL_FORMAT_ASMT

Building a Format Library

- Variable Name – variable name
- Format Name – name of the format
- Varlabel – value label
- Start – start of a range or value
- End – end of a range
- Hlo – specialized formats – high, low, other
- Sexcl – exclude the start of a range
- Eexcl – exclude the end of a range

Building a Format Library

```
data temp;
    d1='01jan1900'd; d2='01jan1960'd; d3=today(); d4='01jan1940'd;
run;
proc print data=temp;
run;
proc print data=temp;
format d1 d2 d3 d4 mmddyy10.;
run;
proc format fmtlib;
     value foo '01jan1900'd='Invalid'
               '01jan1940'd='Still in'
               '01jan1960'd='SAS zero'
               other=[mmddyy10.];
run;
proc print data=temp;
format d1 d2 d3 d4 foo.;
run;
proc format cntlout=foo2;
run;
proc print data=foo2;
run;
```

Building a Format Library

Obs	d1	d2	d3	d4
1	-21914	0	22475	-7305

Obs	d1	d2	d3	d4
1	01/01/1900	01/01/1960	07/14/2021	01/01/1940

Obs	d1	d2	d3	d4
1	Invalid	SAS zero	07/14/2021	Still in

Building a Format Library

```
-----------------------------------------------------------------------------
|       FORMAT NAME: FOO      LENGTH:   10   NUMBER OF VALUES:    4         |
|   MIN LENGTH:   1  MAX LENGTH:  40  DEFAULT LENGTH:  10  FUZZ: STD        |
|---------------------------------------------------------------------------|
|START           |END             |LABEL  (VER. V7|V8   14JUL2021:13:08:59)|
|----------------+----------------+-------------------------------------|
|          -21914|          -21914|Invalid                                |
|           -7305|           -7305|Still in                               |
|               0|               0|SAS zero                               |
|**OTHER**       |**OTHER**       |[MMDDYY10.]                            |
-----------------------------------------------------------------------------
```

Building a Format Library

01.1_Person_CNTLIN.lst - Notepad

File Edit Format View Help

Obs	FMTNAME	START	END	LABEL	MIN	MAX	DEFAULT	DELEN	FUZZ	PREFIX	MULT	FILL	NOEDIT	TYPE	SEXCL	EXCL	HOLE	DECSEP	DIGSEP	DATATYPE	LANGUAGE
1	FOO	-21914	-21914	Invalid	1	40	10	10	1E-12		0		0	N	N	N					
2	FOO	-7305	-7305	Still in	1	40	10	10	1E-12		0		0	N	N	N					
3	FOO	0	0	SAS zero	1	40	10	10	1E-12		0		0	N	N	N					
4	FOO	**OTHER**	**OTHER**	MMDDYY10.	1	40	10	10	1E-12		0		0	N	N	N	OF				

Ln 44, Col 1

Building a Format Library

01.1_Person_CNTLIN.sas - Notepad

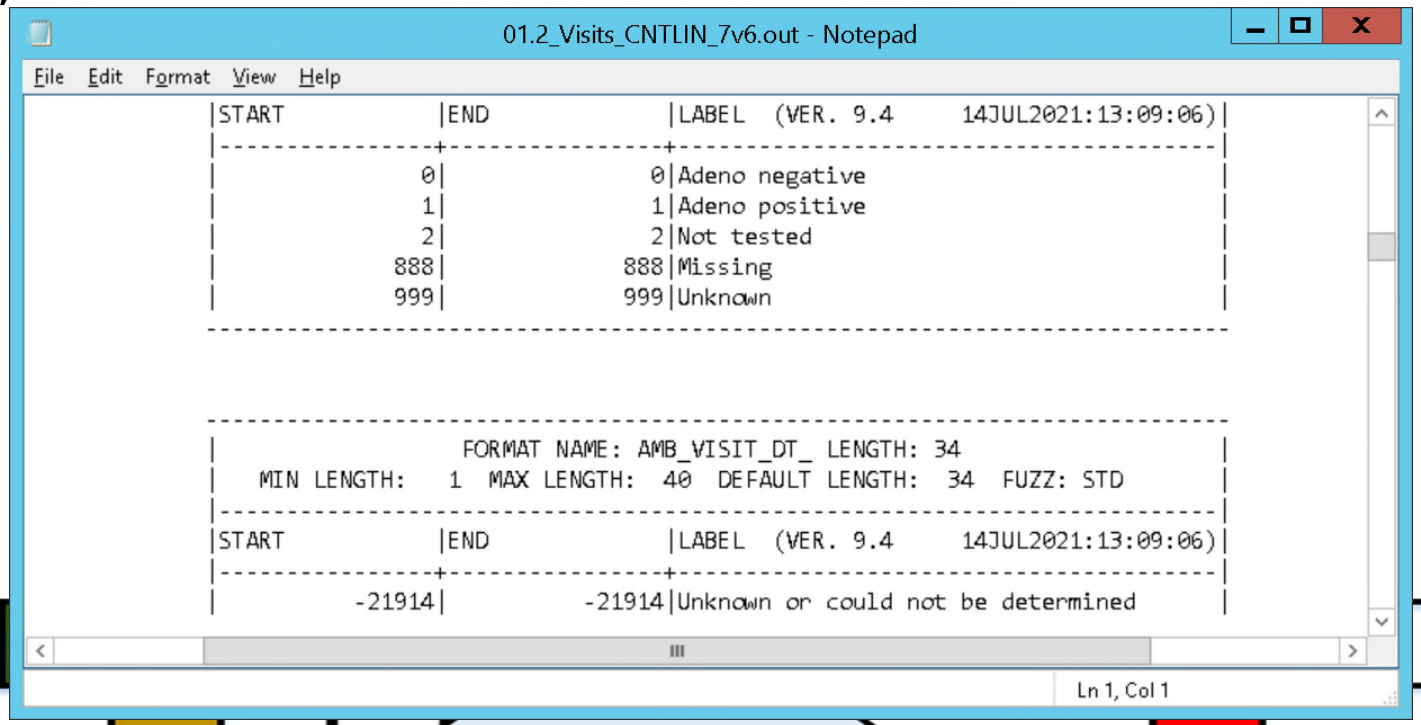
File Edit Format View Help

```
*****************************************************************************************
*** Create CNTLIN file
*****************************************************************************************

data personformats_&procmo (keep=variable_name fmtname start end
    label type hlo sexcl eexcl iterated);
    length fmtname $32 type $1 start $14 label $300;
    set person_cntlin (where=(format_req ne 'N') rename=(varlabel=label));
    if variable_type in('CHAR','ID') then type='c';
    else type='n';
    iterated=(index(variable_name,'#')>0);

label fmtname = 'Name of Format'
      start = 'Start of Range for Format'
      end = 'End of Range for Format'
      sexcl = 'Starting value excluded from Range'
      eexcl = 'Ending value excluded from Range'
          label = 'Label for Format'
          type = 'Type of Format'
          hlo = 'High-Low-Other flag'
          iterated = 'Iterated variable' ;

run;
```

Ln 151, Col 27

Building a Format Library

```
proc format library=library.personformats
cntlin=personformats_&procmo fmtlib ;
run;
```



Creating a format assignment statement

```
data fmtstm;
    length fmtstm $ 80 fmtdot $ 33 variable_name $ 32;
    file ".\&outfi._fmt_statement.inc" lrecl=80;
    set temp (where=(indata ne 0));
    fmtdot=cats(fmtname,'.');
    fmtstm=catx(' ',variable_name,fmtdot);
    put fmtstm;
run;
```

Creating a format assignment statement

p_pp_fmt_statement.inc - Notepad

File Edit Format View Help

```
PREGNANCY_COUNTER PREGNANCY_COUNTER_.
PP_MAT_MEDICAID MAT_MEDICAID_.
PP_DATA_EXTRCT_DT DATA_EXTRCT_DT_.
PP_FLUVX_SEASON FLUVX_SEASON_.
PP_FLUVX_PR_SEASON FLUVX_PR_SEASON_.
PP_FLUVX_SEASON_DT FLUVX_SEASON_DT_.
PP_FLUVX_PR_SEASON_DT FLUVX_PR_SEASON_DT_.
PP_COVIDVX1 COVIDVX1_.
PP_COVIDVX2 COVIDVX2_.
PP_COVIDVX_DT1 COVIDVX_DT1_.
PP_COVIDVX_DT2 COVIDVX_DT2_.
PP_COVIDVX2_EXP COVIDVX2_EXP_.
PP_COVIDVX_EXP_DT1 COVIDVX_EXP_DT1_.
PP_COVIDVX_EXP_DT2 COVIDVX_EXP_DT2_.
PP_MAT_DEATH MAT_DEATH_.
PP_MAT_DEATH_DT MAT_DEATH_DT_.
PP_HR_BLOOD_PL HR_BLOOD_PL_.
```

Ln 1, Col 1

Useful By-Products

The same process of iteration and concatenation based on metadata elements is used to create macro calls to create a codebook, a range report and a “missingness” report.

Useful By-Products

Variable Name	Variable Description	# of Variable values	Missing Value Levels	Missing Value Levels
PR_ASSISTED_REP	Mo 7v2: Was the pregnancy a result of Assisted Reproduction?	3	1	2
PR_DATA_EXTRCT_DT	Mo 7v2: Date of data extraction	1	0	1
PR_FLUVX_SEASON	Mo 7v2: Current season influenza vaccination (August 1st 2020 to May 31st, 2021)	4	0	4
PR_FLUVX_PR_SEASON	Mo 7v2: Prior season influenza vaccination (August 1st 2019 to May 31st, 2020)	4	0	4
PR_FLUVX_SEASON_DT	Mo 7v2: Current season influenza vaccination date (August 1st 2020 to May 31st, 2021)	132	0	132
PR_FLUVX_PR_SEASON_DT	Mo 7v2: Prior season influenza vaccination date (August 1st 2019 to May 31st, 2020)	232	0	232
PR_COVIDVX1	Mo 7v2: First COVID-19 vaccination (if vaccine available)?	1	1	0
PR_COVIDVX2	Mo 7v2: Second COVID-19 vaccination (if vaccine available)?	3	1	2

Useful By-Products

RangeChecks_kpnw_IN_person_7v6.xlsx - Excel

F16 | 0, 1, 999

	A	B	C	D	E	F
1	VARNUM	ANALVAR	LABEL	TYPE	LEN	VALUELIST
2	1	STUDY_ID	Mo 7v6: Participant ID	Char	8	Suppressed
3	2	SITE	Mo 7v6: Sub-site or region	Char	4	KPNW
4	3	INF_IDENTIFIER	Mo 7v6: Infant identifier #1	Char	8	Suppressed
5	4	PREGNANCY_COUNTER	Mo 7v6: Counter indicating which pregnancy this is during the study window for each eligible pregnant woman.	Num	8	1, 2
6	5	IN_DOB	Mo 7v6: Date of birth	Num	8	03/01/2020 - 02/28/2021
7	6	IN_SEX	Mo 7v6: Infant sex	Num	8	1, 2, 999
8	7	IN_DATA_EXTRCT_DT	Mo 7v6: Date of data extraction	Num	8	04/21/2021
9	8	IN_LVSYS_DT	Mo 7v6: Approximate date of person leaving the health system	Num	8	03/31/2020 - 01/31/2021
10	9	IN_INF_DEATH	Mo 7v6: Infant death	Num	8	0, 1
11	10	IN_INF_DEATH_DT	Mo 7v6: Date of infant death	Num	8	., 05/16/2020 - 08/22/2020
12	11	IN_HR_BLOOD_PL	Mo 7v6: Blood disorder diagnosis	Num	8	0, 1, 999
13	12	IN_HR_BLOOD_EXCLD_ANEI	Mo 7v6: Blood disorder diagnosis OTHER THAN ANEMIA	Num	8	0, 1, 999
14	13	IN_HR_DIABT_PL	Mo 7v6: Diabetes diagnosis	Num	8	0, 999
15	14	IN_HR_AC_HEART_FAIL_PL	Mo 7v6: Acute heart failure diagnosis	Num	8	0, 999
16	15	IN_HR_HEART_FAIL_PL	Mo 7v6: (Acute or chronic) heart failure diagnosis	Num	8	0, 1, 999
17	16	IN_HR_CARD_ARRST_PL	Mo 7v6: Cardiac arrest diagnosis	Num	8	0, 999

PRINTALL

READY 100%

Scenario 2

- Large scale, multi site study involving surveillance, COVID testing and other health metrics
- Data dictionaries are used as both documentation and specs for REDCap forms, and to document files sent to CDC on a weekly+ basis
- Need a system to compare data and documentation from REDCap (and more) with final data and documentation sent to CDC

Scenario 2 solutions

- Produce preliminary metadata on incoming weekly surveillance data
- Produce metadata from curated codebook from prior delivery plus changes
- Use curated metadata to create preliminary deliverable file (position, etc.)
- Produce metadata on preliminary deliverable file
- Compare metadata and format libraries

SCENARIO 2 - USING CONTROL FILE

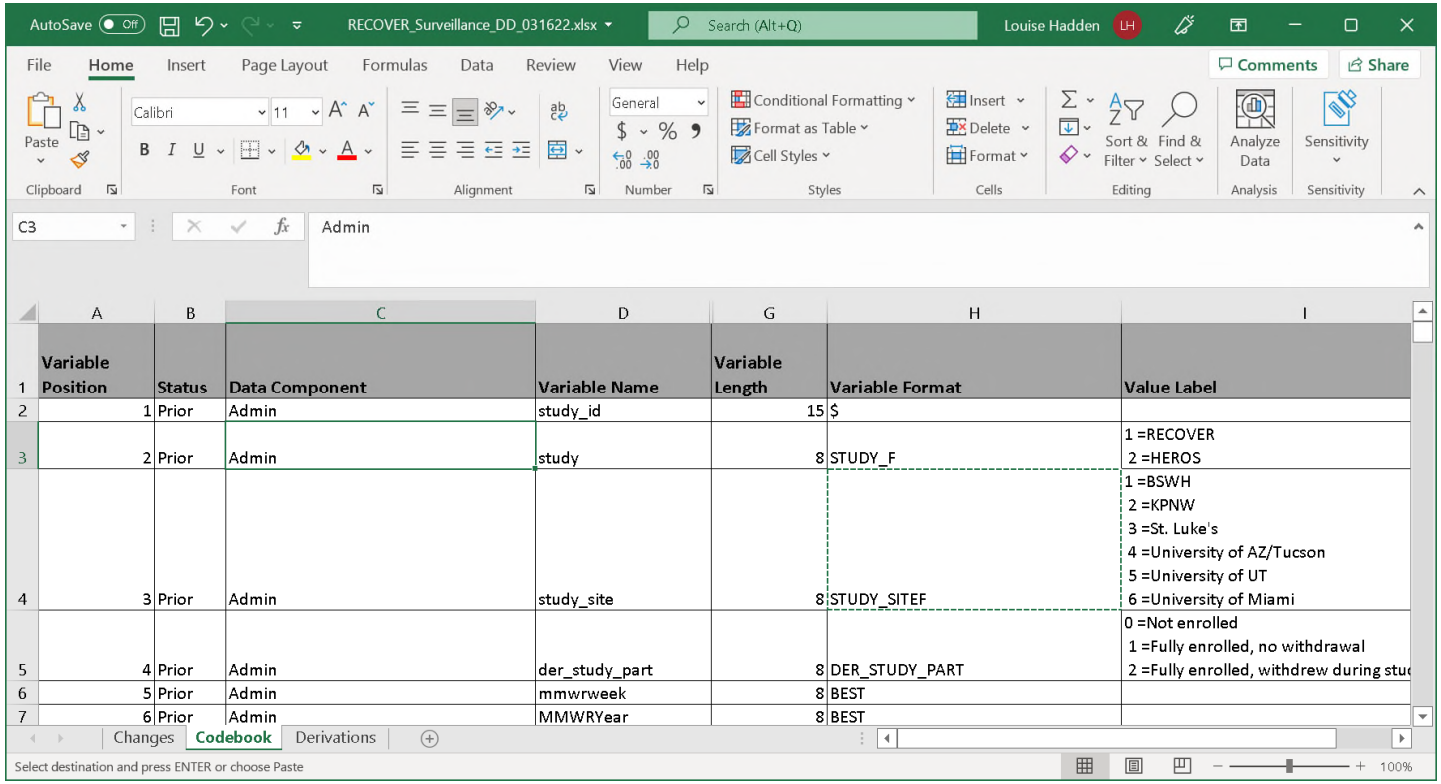
SurveillanceDateInclude.sas - Notepad

```
File Edit Format View Help
*****************************************************************;
** Macro variables for Intermediate Surveillance Processing     **;
** Reset dates each run week based on input file date           **;
** and delivery dates                                           **;
** Reset split spreadsheet file name as needed                  **;
** Set originating directory                                    **;
** Set delivery directory                                       **;
*****************************************************************;
*****************************************************************;
** General Processing Notes                                     **;
** Create Folder for Biweekly Surveillance in                   **;
** S:\Projects\RECOVER_HCP_and_FR\2_PROGRAMS\Surveillance       **;
** Folder name shouild be the delivery date mmddyy format       **;
** Create subfolder named QC under folder                       **;
** Copy SAS Programs including this file into new folder        **;
** Edit this file for new dates                                 **;
** Copy program tracking spreadsheet into new folder            **;
** Rename tracker with new delivery date                        **;
*****************************************************************;
*****************************************************************;
** input file date                                              **;
*****************************************************************;
%let inputfiledate=110821;
*****************************************************************;
** delivery folder dates                                        **;
*****************************************************************;
%let delivdate=11082021;
%let priordeliv=110321;
%let shortdeliv=111021;
*****************************************************************;
** Handoff folder path                                          **;
*****************************************************************;
%let handoffdir=S:\Projects\RECOVER_HCP_and_FR\1_DATA\SAS Data\Hand_
*****************************************************************;
** Handoff folder path                                          **;
*****************************************************************;
%let delivdir=S:\Projects\RECOVER_HCP_and_FR\6_DELIVERABLES\Data Del
*****************************************************************;
** general libname and filename statements                      **;
*****************************************************************;

libname in "&handoffdir";
libname out "S:\Projects\RECOVER_HCP_and_FR\2_PROGRAMS\Surveillance\
libname del "&delivdir";
libname library "S:\Projects\recover_hcp_and_fr\1_data\sas formats\"
libname curr ".";
filename odsout ".";
Ln 28, Col 1
```

02surveillancereorder.sas - Notepad

```
File Edit Format View Help
options ps=55 ls=175 errorabend compress=char nofmterr;
options mprint symbolgen mlogic;

libname formats "S:\Projects\RECOVER_HCP_and_FR\2_PROGRAMS\Ad Hoc Tasks

*Assign formats location;
options fmtsearch=(formats.formats);

%let pgmname=02surveillancereorder;

%include ".\surveillancedateinclude.sas";


*****************************************************************
** Program: &pgmname..sas                                      **
** Creation Date: April 1, 2021                                **
** Project: RECOVER                                            **
** Programmer: Louise Hadden                                   **
** Purpose: Produce Contents Spreadsheets for Surveillance Input **
** Input(s): \1_data\sas data\hand_off\
                    SURVEILLANCE_LAB_CLEAN_&inputfiledate..SAS7BDAT **
** Input(s): \1_data\sas data\hand_off\
                    surveil_lab_clean_exp_&inputfiledate..SAS7BDAT **
** Output: Contents Spreadsheet for Biweekly Surveillance file  **
** Notes:                                                      **
** Peer Review (Date, QCer, any remarks)                       **
** Revision History (Date, Programmer, Change Description)     **
*****************************************************************


*****************************************************************
** library names are set in include file                       **
*****************************************************************


*****************************************************************
** import tabs in spreadsheet for keep statements              **
*****************************************************************


%macro imptabs(tabn=1,tabnm=Codebook,intab=Codebook,startrow=1,endcol=I

proc import dbms=xlsx out = temp
    datafile = ".\QC\RECOVER_Surveillance_DD_&shortdeliv..xlsx" replace
    RANGE="&intab.$A&startrow.:&endcol.9999";
    getnames=YES;
run;

data &tabnm;
    length tabname $ 300 variable_name $ 32 status $ 7;
    set temp;
Ln 1, Col 1
```

SCENARIO 2 – DD FROM PRIOR DELIVERY + CHANGES



	A	B	C	D	G	H	I
1	Variable Position	Status	Data Component	Variable Name	Variable Length	Variable Format	Value Label
2	1	Prior	Admin	study_id	15	$	
3	2	Prior	Admin	study	8	STUDY_F	1 =RECOVER 2 =HEROS
4	3	Prior	Admin	study_site	8	STUDY_SITEF	1 =BSWH 2 =KPNW 3 =St. Luke's 4 =University of AZ/Tucson 5 =University of UT 6 =University of Miami
5	4	Prior	Admin	der_study_part	8	DER_STUDY_PART	0 =Not enrolled 1 =Fully enrolled, no withdrawal 2 =Fully enrolled, withdrew during stu
6	5	Prior	Admin	mmwrweek	8	BEST	
7	6	Prior	Admin	MMWRYear	8	BEST	

SCENARIO 2 – CONTENTS ON INCOMING FILE

01surveillancecontents.sas - Notepad

```
*****************************************************************************************;
*** Exploratory Contents on created files                                             ***;
*****************************************************************************************;

%macro pos(dsnum,dsin,tit2);

ODS OUTPUT attributes=attributes position=position;

PROC CONTENTS DATA=in.&dsin VARNUM;
title2 "Contents of &tit2 for &delivdate.";
RUN;

ODS OUTPUT CLOSE;

data position&dsnum;
    length variable $ 32;
    set position (drop=member);
run;

proc export data = position&dsnum dbms = excel
    outfile = ".\&tit2._Contents_&delivdate..xlsx" replace;
run;

data curr.raw_contents_&delivdate.;
    set position&dsnum;
run;

proc print data=curr.raw_contents_&delivdate. (obs=5) noobs;
run;

%mend;
```

Ln 66, Col 1

surveil_lab_clean_exp_Content...

Char

	A	B	C	D	E	F
1	variable	Num	Type	Len	Format	Label
2	weekly1_specimen_id	1	Char	10		Weekly 1: Study-specific specimen ID which appears on an
3	weekly2_specimen_id	2	Char	10		Weekly 2: Study-specific specimen ID which appears on an
4	weekly3_specimen_id	3	Char	10		Weekly 3: Study-specific specimen ID which appears on an
5	wetswab1_specimen_id	4	Char	10		Wet Swab 1: Study-specific specimen ID which appears on
6	wetswab2_specimen_id	5	Char	10		Wet Swab 2: Study-specific specimen ID which appears on
7	wetswab3_specimen_id	6	Char	10		Wet Swab 3: Study-specific specimen ID which appears on
8	wetswab4_specimen_id	7	Char	10		Wet Swab 4: Study-specific specimen ID which appears on
9	wetswab5_specimen_id	8	Char	10		Wet Swab 5: Study-specific specimen ID which appears on
10	study_id	9	Char	15	$15.	Unique participant ID assigned at screening: 7 characters (
11	study	10	Num	8	STUDY_F.	[derived variable] Name of Study
12	study_site	11	Num	8	STUDY_SITEF.	Study site
13	der_study_part	12	Num	8	DER_STUDY_PART.	Study participation status
14	MMWRWEEK	13	Num	8	BEST12.	MMWR week number that this row of data was collected i
15	MMWRYear	14	Num	8	BEST12.	Year of MMWR week number
16	startweek	15	Num	8	MMDDYY10.	Date of First Day in MMWRWEEK
17	endweek	16	Num	8	MMDDYY10.	Date of Last Day in MMWRWEEK

POSITION1

Scenario 2 – Import prior DD

02surveillancereorder.sas - Notepad

File Edit Format View Help

```
*************************************************************************;
** import tabs in spreadsheet for keep statements                      **;
*************************************************************************;

%macro imptabs(tabn=1,tabnm=Codebook,intab=Codebook,startrow=1,endcol=I);

proc import dbms=xlsx out = temp
    datafile = ".\QC\RECOVER_Surveillance_DD_&shortdeliv..xlsx" replace;
    RANGE="&intab.$A&startrow.:&endcol.9999";
    getnames=YES;
run;

data &tabnm;
    length tabname $ 300 variable_name $ 32 status $ 7;
    set temp;
    tabname = "&intab";
    if status='Updat' then status='Updated';
    if variable_name='' then delete;
run;

proc print data=&tabnm (obs=5) noobs;
format _character_ $40.;
title2 "Test Print of &tabnm";
run;

proc contents data=&tabnm varnum;
title2 "Contents of &tabnm";
run;

proc sort data=&tabnm.;
    by variable_position;
```

Ln 48, Col 14

02surveillancereorder.sas - Notepad

File Edit Format View Help

```
*************************************************************************;
** now create include files for status, label and data component **;
** using edited DD from prior delivery                                 **;
*************************************************************************;

data _null_;
    length status_statement $ 100;
    file '.\INCLUDE_status.txt' lrecl=100;
    set curr.Codebook_&shortdeliv.;
    word1="IF name='";
    word2="' THEN status='";
    word3="';";
    status_statement=cats(word1,variable_name,word2,status,word3);
    put status_statement;
run;

data _null_;
    length data_component_statement $ 300;
    file '.\INCLUDE_data_component.txt' lrecl=300;
    set curr.Codebook_&shortdeliv.;
    word1="IF name='";
    word2="' THEN data_component='";
    word3="';";
    data_component_statement=cats(word1,variable_name,word2,data_component
    put data_component_statement;
run;

data _null_;
    length label_statement $ 1000;
    file '.\INCLUDE_label.txt' lrecl=1000;
    set curr.Codebook_&shortdeliv.;
```

Ln 1, Col 1

Scenario 2 – Import prior DD

02surveillancereorder.sas - Notepad

File Edit Format View Help

```
*******************************************************************;
** Set up a list to reorder the combined file                    **;
*******************************************************************;

PROC SQL;
SELECT VARIABLE_NAME
INTO :COMBKEEP SEPARATED BY ' '
FROM curr.Codebook_&shortdeliv.;
quit;

%put &combkeep;

*Create dataset with shortened dataset name for codebook macro;
data cb out.surv_formatted_&delivdate.  ;
retain &combkeep ;
set in.surveil_lab_clean_exp_&inputfiledate;
    keep &combkeep.;


    %include '.\INCLUDE_label.txt';
run;

*Read out reordered dataset with formats stripped;
data out.RECOVER_Surveillance_&delivdate.;
retain &combkeep.;
set cb ;
informat _all_ ;
format _all_ ;
run;

*************************************************************************
```

Ln 153, Col 1

02surveillancereorder.sas - Notepad

File Edit Format View Help

```
proc export data = position dbms = excel
    outfile = ".\Formatted_Surveillance_Contents_&delivdate..xlsx" replace
run;

proc export data = position (where=(flag_label_length=1)) dbms = excel
    outfile = ".\TooLongLabelLengths&delivdate..xlsx" replace;
run;

ODS OUTPUT attributes=attributes position=position;

PROC CONTENTS DATA=out.RECOVER_Surveillance_&delivdate. VARNUM;
title2 "Contents of Unformatted Surveillance File for &delivdate.";
RUN;

ODS OUTPUT CLOSE;

proc print data=position (obs=5) noobs;
title2 "Test print contents of deliverable Surveillance file for &shortdel
run;

data position;
    length variable $ 32;
    set position (drop=member);
    length_label=length(label);
run;

proc export data = position dbms = excel
    outfile = ".\Unformatted_Surveillance_Contents_&delivdate..xlsx" repla
run;
```

Ln 201, Col 1

Scenario 2 – Quality assurance (comparisons)

05CompareDocumentationWIP.sas - Notepad

File Edit Format View Help

```
** massage data sets to trim down variables to compare **;
*********************************************************;

data prior;
    length value_label $ 500;
    set curr.codebook_&shortdeliv. (keep=variable_name status variable_position
                                    data_component variable_description
                                    variable_type variable_length
                                    variable_format value_label);
informat _all_;
format _all_;

    value_label=TRANSLATE(value_label,' ','09'x); /* replace tabs with a single space */
    value_label=translate(value_label,' ','0A'x); /* replace CR with a single space */
    value_label=translate(value_label,' ','0D'x); /* replace LF with a single space */
    value_label=compbl(value_label);
    value_label=left(value_label);

run;

proc sort data=prior;
    by variable_position;
run;

data new;
    length data_component $ 63 value_label $ 500;
    set curr.metadata_full (keep=name status varnum data_component
                                label vtype length format value_label
                            rename=(name=variable_name varnum=variable_position
                                label=variable_description vtype=variable_type
                                length=variable_length format=variable_format));
```

Ln 45, Col 1

05CompareDocumentationWIP.sas - Notepad

File Edit Format View Help

```
proc sort data=master_formats (keep=fmtname) out=master_names nodupkey;
    by fmtname;
run;

proc sort data=surv_formats (keep=fmtname) out=surv_names nodupkey;
    by fmtname;
run;

data inboth;
    merge master_names (in=a) surv_names (in=b);
    by fmtname;
    if a and b;
run;

data sub_master (keep=fmtname start end hlo label type);
    length start end $ 32 fmtname $ 33;
    merge inboth (in=a) master_formats (in=b);
    by fmtname;
    if a and b;
    if start ne '';
    start=left(start);
    end=left(end);
run;

data sub_surv (keep=fmtname start end hlo label type);
    length start end $ 32 fmtname $ 33;
    merge inboth (in=a) surv_formats (in=b);
    by fmtname;
    if a and b;
    if start ne '';
    start=left(start);
    end=left(end);
```

Ln 97, Col 1

Scenario 2 – Quality assurance (comparisons)

05CompareDocumentationWIP.sas - Notepad

File Edit Format View Help

```
*************************************************************;
** PROC COMPARE of prior metadata with new metadata     **;
*************************************************************;
ods trace on;

ods output comparedatasets=comparedatasets
comparevariables=comparevariables comparesummary=comparesummary;

proc compare base=prior
             compare=new ;
    id variable_position;
title1 'Comparison of major variables';
run;

ods output close;

proc print data=comparedatasets noobs;
title1 'Comparedatasets';
run;

proc print data=comparevariables noobs;
title1 'Comparevariables';
run;

proc print data=comparesummary noobs;
title1 'Comparesummary';
run;

ods trace off;
```

Ln 122, Col 1

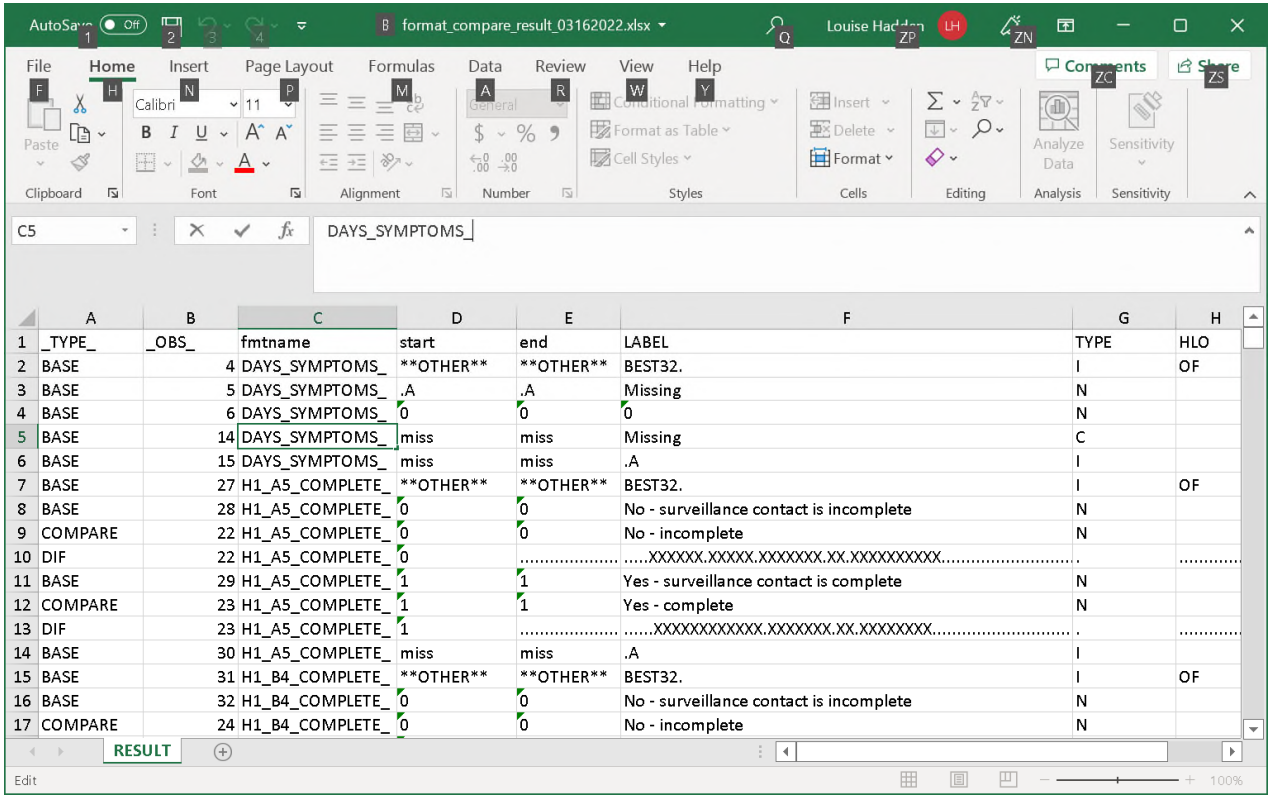
05CompareDocumentationWIP.lst - Notepad

File Edit Format View Help

```
Variables Summary

Number of Variables in Common: 9.
Number of Variables with Differing Attributes: 3.
Number of ID Variables: 1.


Listing of Common Variables with Differing Attributes

Variable              Dataset     Type  Length  Label

Variable_Position     WORK.PRIOR  Num        8  Variable Position
                      WORK.NEW    Num        8  Variable Number
Variable_Description  WORK.PRIOR  Char     255  Variable Description
                      WORK.NEW    Char     255  Variable Label
Variable_Format       WORK.PRIOR  Char      31  Variable Format
                      WORK.NEW    Char      32  Variable Format


Observation Summary

Observation      Base  Compare  ID

First Obs           1        1  Variable_Position=1
Last  Obs        1042     1042  Variable_Position=1042

Number of Observations in Common: 1042.
Total Number of Observations Read from WORK.PRIOR: 1042.
Total Number of Observations Read from WORK.NEW: 1042.

Number of Observations with Some Compared Variables Unequal: 0.
```

Ln 1, Col 1

SCENARIO 2 – QUALITY ASSURANCE (COMPARISONS)



	A	B	C	D	E	F	G	H
1	_TYPE_	_OBS_	fmtname	start	end	LABEL	TYPE	HLO
2	BASE	4	DAYS_SYMPTOMS_	**OTHER**	**OTHER**	BEST32.	I	OF
3	BASE	5	DAYS_SYMPTOMS_	.A	.A	Missing	N	
4	BASE	6	DAYS_SYMPTOMS_	0	0	0	N	
5	BASE	14	DAYS_SYMPTOMS_	miss	miss	Missing	C	
6	BASE	15	DAYS_SYMPTOMS_	miss	miss	.A	I	
7	BASE	27	H1_A5_COMPLETE_	**OTHER**	**OTHER**	BEST32.	I	OF
8	BASE	28	H1_A5_COMPLETE_	0	0	No - surveillance contact is incomplete	N	
9	COMPARE	22	H1_A5_COMPLETE_	0	0	No - incomplete	N	
10	DIF	22	H1_A5_COMPLETE_	0	.....	.....XXXXXX.XXXXX.XXXXXXX.XX.XXXXXXXXXX.....	.	.....
11	BASE	29	H1_A5_COMPLETE_	1	1	Yes - surveillance contact is complete	N	
12	COMPARE	23	H1_A5_COMPLETE_	1	1	Yes - complete	N	
13	DIF	23	H1_A5_COMPLETE_	1	.....	.....XXXXXXXXXXXX.XXXXXXX.XX.XXXXXXXX.....	.	.....
14	BASE	30	H1_A5_COMPLETE_	miss	miss	.A	I	
15	BASE	31	H1_B4_COMPLETE_	**OTHER**	**OTHER**	BEST32.	I	OF
16	BASE	32	H1_B4_COMPLETE_	0	0	No - surveillance contact is incomplete	N	
17	COMPARE	24	H1_B4_COMPLETE_	0	0	No - incomplete	N	

CONCLUSION

It is clear that using metadata to drive processing and quality assurance can be incredibly helpful and comprises a valuable addition to one's SAS toolbox. I hope you'll have some fun iterations with functions with your metadata as well!

ACKNOWLEDGEMENTS

This type of complex programming and processing is a team sport. I could not have created and implemented these techniques on my own.

A very heartfelt thank you to team members of my projects at Abt.

Contact Information

Louise S. Hadden
Louise_hadden@abtassoc.com

www.linkedin.com/in/louisehadden
https://communities.sas.com/t5/user/viewprofilepage/user-id/12504

Code samples provided on request.