```
*******************************************************************************
**************;
*
*           CLIENT:  ModernaTX, Inc.
*         PROTOCOL:  mRNA-1273-P201
*
*          PURPOSE: Create analysis dataset adarp7d
*
*      INPUT FILES: SDTM domains
*     OUTPUT FILES: ADARP7D.sas7bdat
*
*      USAGE NOTES:
*
*******************************************************************************
**************;
*  Copyright 2020 Pharmaceutical Product Development, Inc.
*  All Rights Reserved.
*******************************************************************************
**************;

options noquotelenmax;
%include "madam.sas";

**Assign global macro variable DSETNAME to reflect the name of the final ADaM
dataset**;
%global DSETNAME;
%let dsetname = adarp7d;
%let adslvar=tr01sdt tr01sdtm tr01edt tr01edtm dose2dt dos2dtm;* rfxstdtc be moved
from adsl;

/*proc contents data=adb.adsl;run;*/

** PARAMCD>>PARAM FORMAT;
proc format;
        value $param
        'RASHOCC'='Rash Occurrence'
        "LYMPH" = "Axillary Swelling or Tenderness"
        "PAIN" = "Pain"
        "ERYTHDIA" = "Erythema Longest Diameter (mm)"
        "SWELLDIA" = "Swelling Longest Diameter (mm)"
        "HEADACHE" = "Headache"
        "FATIGUE" = "Fatigue"
        "MYALGIA" = "Myalgia"
        "ARTHRALG" = "Arthralgia"
        "NAUSEA" = "Nausea/Vomiting"
        "CHILLS" = "Chills"
/*      "MEDATTEN" = 'Receive Medical Attention?' */
        "FEVER" = "Fever (C)"
  "MEDTAKP"='Prevent Pain or Fever from Occurring'
  "MEDTAKT"='Treat Pain or Fever already Occurred'
```

```
  "MEDTAK" ='Medication Taken Today for Pain or Fever';
run;

%macro supp_merge_test(domain);
        %if %sysfunc(exist(trans.supp&domain.))=1 %then %do;

%revsupp(libin=Trans,libout=WORK,ds=&domain.,supp=trans.supp&domain.,outds=&domain.
_all);
        %end;
        %else %do;
                data &domain._all;
                        set trans.&domain.;
                run;
        %end;
%mend supp_merge_test;
%supp_merge_test(face);
%supp_merge_test(faae);

**Merge supplemental data onto parent domain**;
%revsupp(libin=trans,libout=work,ds=vs,supp=suppvs,outds=vs_all);

DATA FACE_ALL;
        SET FACE_ALL;

        **derive param and paramcd;
        if faobj = 'Rash' and fatestcd ne 'SEV' then paramcd = 'RASHOCC';
        else if faobj='Underarm Gland Swelling or Tenderness' then PARAMCD =
'LYMPH';
        else if faobj in ('Pyrexia' 'Fever') then PARAMCD = 'FEVER';
        else if faobj='Pain' then PARAMCD = 'PAIN';
        else if faobj='Nausea/Vomiting' then PARAMCD = 'NAUSEA';
        else if faobj='Erythema' then PARAMCD = 'ERYTHDIA';
        else if faobj='Swelling' then PARAMCD = 'SWELLDIA';
        else if faobj='Headache' then PARAMCD = 'HEADACHE';
        else if faobj='Fatigue' then PARAMCD = 'FATIGUE';
        else if faobj='Myalgia' then PARAMCD = 'MYALGIA';
        else if faobj='Arthralgia' then PARAMCD = 'ARTHRALG';
        else if faobj='Chills' then PARAMCD = 'CHILLS';
/*      else if faobj='Receive Medical Attention' then PARAMCD = 'MEDATTEN'; */


        if PARAMCD in ('LYMPH' 'PAIN' 'ERYTHDIA' 'SWELLDIA' 'HEADACHE' 'FATIGUE'
                'MYALGIA' 'ARTHRALG' 'NAUSEA' 'CHILLS' 'MEDATTEN' 'FEVER'
'RASHOCC') then PARAM = put(PARAMCD,$param.);
        else PARAM = faobj;
  domain_orig='FACE';
RUN;
DATA FAAE_ALL;
        SET FAAE_ALL;
```

```
        **derive param and paramcd;
                if faobj = 'Rash' and fatestcd ne 'SEV' then paramcd = 'RASHOCC';
        else if faobj='Underarm Gland Swelling or Tenderness' then PARAMCD =
'LYMPH';
        else if faobj in ('Pyrexia' 'Fever') then PARAMCD = 'FEVER';
        else if faobj='Pain' then PARAMCD = 'PAIN';
        else if faobj='Nausea/Vomiting' then PARAMCD = 'NAUSEA';
        else if faobj='Erythema' then PARAMCD = 'ERYTHDIA';
        else if faobj='Swelling' then PARAMCD = 'SWELLDIA';
        else if faobj='Headache' then PARAMCD = 'HEADACHE';
        else if faobj='Fatigue' then PARAMCD = 'FATIGUE';
        else if faobj='Myalgia' then PARAMCD = 'MYALGIA';
        else if faobj='Arthralgia' then PARAMCD = 'ARTHRALG';
        else if faobj='Chills' then PARAMCD = 'CHILLS';
/*      else if faobj='Receive Medical Attention' then PARAMCD = 'MEDATTEN'; */


        if PARAMCD in ('LYMPH' 'PAIN' 'ERYTHDIA' 'SWELLDIA' 'HEADACHE' 'FATIGUE'
                'MYALGIA' 'ARTHRALG' 'NAUSEA' 'CHILLS' 'MEDATTEN' 'FEVER'
'RASHOCC') then PARAM = put(PARAMCD,$param.);
        else PARAM = faobj;
  domain_orig='FAAE';
RUN;


**which subjid need to keep meet conditions (Only keep subjects who has records in
FAAE or (VS WHERE VSCAT=REACTOGENICITY and VSTPTNUM>7)) ;

**keep the subjects with FEVER in both FAAE/VS**;
data VS_FV_subjid;
        set vs_all(where=(^(^(vstptnum>7 and vscat='REACTOGENICITY') or vsstat =
"NOT DONE")  and ^missing(VSTPTREF)));
        if vstestcd='TEMP';
        paramcd='FEVER';
        FATPTREF=VSTPTREF;
        keep usubjid subjid PARAMCD FAtptref;
run;
data FAAE_fv_subjid;
        set faae_all(where=(^((lowcase(faobj) in ('erythema' 'swelling') and
fatest='Occurrence Indicator' AND FAORRES='Y') or fastat = "NOT DONE")  and
^missing(faTPTREF)));
        if paramcd='FEVER';
        VSTESTCD='TEMP';
        VSTPTREF=FATPTREF;
        keep usubjid VSTESTCD vstptref;
run;

data VS_subjid;
  length VSTPTREF $200.;
        set vs_all(where=(^(^(vstptnum>7 and vscat='REACTOGENICITY') or vsstat =
```

```
"NOT DONE")   and ^missing(vsTPTREF)))
                FAAE_FV_SUBJID;
        keep usubjid subjid VSTESTCD vstptref;
run;
data FAAE_subjid;
        set faae_all(where=(^((lowcase(faobj) in ('erythema' 'swelling') and
fatest='Occurrence Indicator' AND FAORRES='Y') or fastat = "NOT DONE")  and
^missing(faTPTREF)))
                vs_fv_subjid;
        keep usubjid subjid FATESTCD fatest faobj FAORRES PARAMCD FATPTREF;
run;

proc sql noprint;
        create table _FAAE_subjid as
        select distinct usubjid,PARAMCD,FATPTREF
        from FAAE_subjid
        order by usubjid,PARAMCD,FATPTREF;
quit;
proc sql noprint;
        create table _VS_subjid as
        select distinct usubjid,VSTESTCD,vstptref
        from VS_subjid
        order by usubjid,VSTESTCD,vstptref;
quit;
PROC SORT DATA=vs_all;by usubjid  VSTESTCD vstptref;run;
PROC SORT DATA=face_all;by usubjid  PARAMCD FATPTREF;run;
PROC SORT DATA=faae_all;by usubjid  PARAMCD FATPTREF;run;


data VS1;
  length VSTPTREF $200.;
        merge vs_all(in=a) _VS_subjid(in=b);
        by usubjid  VSTESTCD vstptref;
        if a and b;
run;
*paramcd-MEDTAKP MEDTAKT MEDTAK;
data vs2;
  length VSTPTREF $200. vstestcd $8. atptref $50. ;
  set vs_all(where=(VSTPTREF is not null));
  if index(upcase(VSTPTREF),'DOSE 1') > 0 and index(upcase(VSTPTREF), 'OL') = 0
then ATPTREF = 'Vaccination 1';
  else if index(upcase(VSTPTREF),'DOSE 2') > 0 and index(upcase(VSTPTREF), 'OL') =
0 then ATPTREF = 'Vaccination 2';
  else if index(upcase(VSTPTREF), 'DOSE 1 OL')> 0 then ATPTREF  = 'OL Vaccination
1';
  else if index(upcase(VSTPTREF), 'DOSE 2 OL')> 0 then ATPTREF  = 'OL Vaccination
2';
  *if input(scan(vstpt,-1,' '),??best.)<8;
  if MEDTAKP='Y' then do;
    vstestcd='MEDTAKP';
```

```
    output;
  end;
  if MEDTAKT='Y' then do;
    vstestcd='MEDTAKT';
    output;
  end;
  if MEDTAK ='Y' then do;
    vstestcd='MEDTAK';
    output;
  end;
run;
**filter subject max(vstptnum)<=7**;
proc sql ;
  create table vs_subjects as
  select distinct usubjid
  from vs_all
  where vstptnum>7 and vscat='REACTOGENICITY' and VSTPTREF is not null;
quit;
proc sql undo_policy=none;
  create table vs2 as
  select *
  from vs2
  where usubjid in (select usubjid from vs_subjects);
quit;
proc sort data=vs2 dupout=dup nodupkey;by usubjid atptref vstpt vstestcd;run;

data FACE1;
        merge face_all(in=a) _FAAE_subjid(in=b);
        by usubjid  PARAMCD FATPTREF;
        if a and b;
run;

data FAAE1;
        merge faae_all(in=a) _FAAE_subjid(in=b);
        by usubjid  PARAMCD FATPTREF;
        if a and b;
run;

** set source data together;
data combine(DROP=PARAM PARAMCD);
  length vstestcd $8.;
        set vs1(in=vs where=(vscat='REACTOGENICITY'  and vsstat ^= "NOT DONE"))
      vs2(drop=atptref)
                FACE1(in=face WHERE=( ^((lowcase(faobj) in ('erythema' 'swelling')
and fatest='Occurrence Indicator' AND FAORRES='Y') or fastat = "NOT DONE" or
UPCASE(facat) ^= "REACTOGENICITY")))
                FAAE1(in=faae where=( ^((lowcase(faobj) in ('erythema' 'swelling')
and fatest='Occurrence Indicator' AND FAORRES='Y') or fastat = "NOT DONE" or
UPCASE(facat) ^= "REACTOGENICITY")));
run;
```

```
**KEEP NEED SUBJID;
**add adsl variable and choose subjid in adsl;
proc sort tagsort data=combine; by usubjid ; run;

data combine_adsl;
        merge combine(in=a) adb.adsl(in=b keep=USUBJID  &adslvar.);
        by usubjid ;
        if a and b;
run;


data combine_final;
        length PARCAT1 atptgr1 atptref $50 AVALC atpt $100 PARAMCD $8 _war param
$200 ;
        set combine_adsl;

        * Analysis Datetime;
        if domain='VS' then fadtc = vsdtc;
        %ISO2SAS(isodate=fadtc, datec=_fadtc, daten=ADT, timec=_atm);
        if ^missing(_atm) and ^missing(ADT) then ADTM =
input(compress(put(ADT,date9.)||':'||_atm),datetime20.);
        if ^missing(_atm) then ATM = input(_atm,time5.);

        if domain="FA" then ady=fady;
        else if domain="VS" then ady=vsdy;

        * Most Recent Dose Reference;
        %*xxdoserf(xx=FA,adt=ADT,adtm=ADTM,rddy=RDDY);

        if domain = "FA" then do;
                if index(upcase(fatptref),'DOSE 1') > 0 and index(upcase(fatptref),
'OL') = 0 then ATPTREF = 'Vaccination 1';
                else if index(upcase(fatptref),'DOSE 2') > 0 and
index(upcase(fatptref), 'OL') = 0 then ATPTREF = 'Vaccination 2';
        else if index(upcase(fatptref), 'DOSE 1 OL')> 0 then ATPTREF  = 'OL
Vaccination 1';
        else if index(upcase(fatptref), 'DOSE 2 OL')> 0 then ATPTREF  = 'OL
Vaccination 2';
        end;
        if domain = "VS" then do;
                if index(upcase(vstptref),'DOSE 1') > 0 and index(upcase(vstptref),
'OL') = 0 then ATPTREF = 'Vaccination 1';
                else if index(upcase(vstptref),'DOSE 2') > 0 and
index(upcase(vstptref), 'OL') = 0 then ATPTREF = 'Vaccination 2';
        else if index(upcase(vstptref), 'DOSE 1 OL')> 0 then ATPTREF  = 'OL
Vaccination 1';
        else if index(upcase(vstptref), 'DOSE 2 OL')> 0 then ATPTREF  = 'OL
```

```
Vaccination 2';
        end;

        ** parameter category and numerica format;
        if domain="FA" AND find(fascat,'ADMINIST','i')>0 THEN PARCAT1 =
"LOCAL";*UPCASE(FASCAT)="ADMINISTRATION SITE";
        ELSE IF domain="FA" THEN PARCAT1 = FASCAT;
        ELSE PARCAT1 = vsscat;

        ** atpt atptn;
        if domain="FA"  THEN atpt = strip(fatpt);
        ELSE atpt = strip(vstpt);

        *eval;
/*      length eval $30.;*/
/*      if domain='VS' then eval='';*/
/*      else eval=cats(faeval);*/
/*      faeval=eval;*/


        if fatpt='DAY 1' and FAEVAL='INVESTIGATOR' then atpt='DAY 1, AFTER
VACCINATION (AT HOME)';

        if ATPT = 'DAY 1, 30 MINS AFTER VACCINATION (AT STUDY CLINIC)' then ATPTN =
1.05;
        else if atpt = "DAY 1, 1 HOUR AFTER VACCINATION (AT STUDY CLINIC)" then
atptn = 1.1;
        else if atpt = "DAY 1, AFTER VACCINATION (AT HOME)" then atptn = 1.2;
        else atptn =input(compress(atpt,'DAY '),??BEST.);

        IF PARCAT1="LOCAL" THEN PARCAT1N = 1;
        ELSE IF PARCAT1="SYSTEMIC" THEN PARCAT1N = 2;


        **derive param and paramcd;
  if vstestcd in ('MEDTAKP' 'MEDTAKT' 'MEDTAK') then PARAMCD=vstestcd;
        else if vstestcd in ('TEMP' 'VSALL') or faobj in ('Pyrexia' 'Fever') then
PARAMCD = 'FEVER';
        else if faobj = 'Rash' and fatestcd ne 'SEV' then PARAMCD = 'RASHOCC';
        else if faobj='Underarm Gland Swelling or Tenderness' and fatestcd='SEV'
then PARAMCD = 'LYMPH';
        else if faobj='Pain' then PARAMCD = 'PAIN';
        else if faobj='Nausea/Vomiting' and fatestcd='SEV' then PARAMCD = 'NAUSEA';
        else if faobj='Erythema' and find(upcase(fatest),'OCCURRENCE') and
upcase(faorres) in ('N' 'NO') then PARAMCD = 'ERYTHDIA';
        else if faobj='Swelling' and find(upcase(fatest),'OCCURRENCE') and
upcase(faorres) in ('N' 'NO') then PARAMCD = 'SWELLDIA';
        else if faobj='Erythema' and find(upcase(fatest),'INTENSITY') then PARAMCD
= 'ERYTHDIA';
        else if faobj='Swelling' and find(upcase(fatest),'INTENSITY') then PARAMCD
```

```
= 'SWELLDIA';
        else if faobj='Erythema' and lowcase(fatest)='longest diameter' then
PARAMCD = 'ERYTHDIA';
        else if faobj='Swelling' and lowcase(fatest)='longest diameter' then
PARAMCD = 'SWELLDIA';
        else if faobj='Headache' then PARAMCD = 'HEADACHE';
        else if faobj='Fatigue' then PARAMCD = 'FATIGUE';
        else if faobj='Myalgia' then PARAMCD = 'MYALGIA';
        else if faobj='Arthralgia' then PARAMCD = 'ARTHRALG';
        else if faobj='Chills' then PARAMCD = 'CHILLS';
/*      else if faobj='Receive Medical Attention?' then PARAMCD = 'MEDATTEN'; */
        else if ^missing(faobj) or ^missing(fatestcd) or ^missing(vstestcd) then
do;
                _war = 'Put Aler'||'t_A: PARAMCD needs udapte. ';
                put _war SUBJID= FATESTCD= VSTESTCD= FAOBJ=;
        end;

        if PARAMCD in ('MEDTAKP' 'MEDTAKT' 'MEDTAK' 'LYMPH' 'PAIN' 'ERYTHDIA'
'SWELLDIA' 'HEADACHE' 'FATIGUE'
                'MYALGIA' 'ARTHRALG' 'NAUSEA' 'CHILLS' 'MEDATTEN' 'FEVER'
'RASHOCC') then PARAM = put(PARAMCD,$param.);
        else PARAM = faobj;
  if missing(paramcd)=0;

        * AVALC, AVAL;
        if upcase(faorres) in ('N' 'NO') and FATEST='Occurrence Indicator' and
PARAMCD in ('ERYTHDIA' 'SWELLDIA') then do; AVAL = 0; AVALC = fastresc; end;
        else if domain='VS' then do; AVAL = vsstresn; AVALC = vsstresc; if paramcd
in ('MEDTAKP' 'MEDTAKT' 'MEDTAK') then do;aval=.;avalc='Y';end; end;
        else do; AVAL = fastresn; AVALC = fastresc; end;
  if find(avalc,'GRADE','i') then avalc=tranwrd(avalc,'GRADE GRADE ','GRADE ');


        * Analysis Toxicity Grade;
        if PARAMCD in ('ERYTHDIA' 'SWELLDIA') then do;
                if .<AVAL<25 or find(fastresc,'GRADE 0') then ATOXGRN = 0;
                else if 25<=AVAL<=50 or find(fastresc,'GRADE 1') then ATOXGRN = 1;
                else if 50<AVAL<=100 or find(fastresc,'GRADE 2') then ATOXGRN = 2;
                else if AVAL>100 or find(fastresc,'GRADE 3') then ATOXGRN = 3;
                else if find(fastresc,'GRADE 4') then ATOXGRN = 4;
        end;
        else if PARAMCD='FEVER' then do;
                if .<AVAL<38.0 or find(fastresc,'GRADE 0') then ATOXGRN = 0;
                else if 38.0<=AVAL<=38.4 or find(fastresc,'GRADE 1') then ATOXGRN =
1;
                else if 38.4<AVAL<=38.9 or find(fastresc,'GRADE 2') then ATOXGRN =
2;
                else if 38.9<AVAL<=40.0 or find(fastresc,'GRADE 3') then ATOXGRN =
3;
                else if AVAL>40.0 or find(fastresc,'GRADE 4') then ATOXGRN = 4;
```

```
        end;
        else if PARAMCD in ('PAIN' 'LYMPH' 'HEADACHE') then do;
                if upcase(AVALC)='NONE' or find(fastresc,'GRADE 0') then ATOXGRN =
0;
                else if upcase(AVALC)='DOES NOT INTERFERE WITH ACTIVITY' or
find(fastresc,'GRADE 1') then ATOXGRN = 1;
                else if upcase(AVALC)='NO INTERFERENCE WITH ACTIVITY' or
find(fastresc,'GRADE 1') then ATOXGRN = 1;
                else if index(upcase(AVALC),'REPEATED') or find(fastresc,'GRADE 2')
then ATOXGRN = 2;
                else if upcase(AVALC)='ANY USE OF PRESCRIPTION RELIEVER OR PREVENTS
DAILY ACTIVITY' or find(fastresc,'GRADE 3') then ATOXGRN = 3;
                else if upcase(AVALC)='ANY USE OF PRESCRIPTION PAIN RELIEVER OR
PREVENTS DAILY ACTIVITY' or find(fastresc,'GRADE 3') then ATOXGRN = 3;
                else if find(fastresc,'GRADE 4') then ATOXGRN = 4;
                else if ^missing(AVALC) then do;
                        _war = 'Put Aler'||'t_A: ATOXGR needs udapte. ';
                        put _war SUBJID= PARAMCD= PARAM=;
                        put _war AVALC=;
                end;
        end;
        else if PARAMCD in ('FATIGUE' 'MYALGIA' 'ARTHRALG') then do;
                if upcase(AVALC)='NONE' or find(fastresc,'GRADE 0') then ATOXGRN =
0;
                else if upcase(AVALC)='NO INTERFERENCE WITH ACTIVITY' or
find(fastresc,'GRADE 1') then ATOXGRN = 1;
                else if upcase(AVALC)='SOME INTERFERENCE WITH ACTIVITY' or
find(fastresc,'GRADE 2') then ATOXGRN = 2;
                else if upcase(AVALC)='SIGNIFICANT; PREVENTS DAILY ACTIVITY' or
find(fastresc,'GRADE 3') then ATOXGRN = 3;
                else if find(fastresc,'GRADE 4') then ATOXGRN = 4;
                else if ^missing(AVALC) then do;
                        _war = 'Put Aler'||'t_A: ATOXGR needs udapte. ';
                        put _war SUBJID= PARAMCD= PARAM=;
                        put _war AVALC=;
                end;
        end;
        else if PARAMCD='NAUSEA' then do;
                if upcase(AVALC)='NONE' or find(fastresc,'GRADE 0') then ATOXGRN =
0;
                else if upcase(AVALC)='NO INTERFERENCE WITH ACTIVITY OR 1-2
EPISODES/24 HOURS' or find(fastresc,'GRADE 1') then ATOXGRN = 1;
                else if upcase(AVALC)='SOME INTERFERENCE WITH ACTIVITY OR >2
EPISODES/24 HOURS' or find(fastresc,'GRADE 2') then ATOXGRN = 2;
                else if upcase(AVALC)='PREVENTS DAILY ACTIVITY, REQUIRES OUTPATIENT
IV HYDRATION' or find(fastresc,'GRADE 3') then ATOXGRN = 3;
                else if find(fastresc,'GRADE 4') then ATOXGRN = 4;
                else if ^missing(AVALC) then do;
                        _war = 'Put Aler'||'t_A: ATOXGR needs udapte. ';
                        put _war SUBJID= PARAMCD= PARAM=;
```

```
                        put _war AVALC=;
                end;
        end;
        else if PARAMCD='CHILLS' then do;
                if upcase(AVALC)='NONE' or find(fastresc,'GRADE 0') then ATOXGRN =
0;
                else if upcase(AVALC)='NO INTERFERENCE WITH ACTIVITY' or
find(fastresc,'GRADE 1') then ATOXGRN = 1;
                else if upcase(AVALC)='SOME INTERFERENCE WITH ACTIVITY NOT
REQUIRING MEDICAL ATTENTION' or find(fastresc,'GRADE 2') then ATOXGRN = 2;
                else if upcase(AVALC)='PREVENTS DAILY ACTIVITY AND REQUIRES MEDICAL
ATTENTION' or find(fastresc,'GRADE 3') then ATOXGRN = 3;
                else if find(fastresc,'GRADE 4') then ATOXGRN = 4;
                else if ^missing(AVALC) then do;
                        _war = 'Put Aler'||'t_A: ATOXGR needs udapte. ';
                        put _war SUBJID= PARAMCD= PARAM=;
                        put _war AVALC=;
                end;
        end;

        if ^missing(ATOXGRN) then ATOXGR = 'Grade '||strip(put(ATOXGRN,best.));

        ** Surce Data, Sequence, Variable;
        length srcvar $50. SRCDOM $20.;
        if domain='VS' then do; SRCDOM = DOMAIN; SRCSEQ = VSSEQ; SRCVAR =
'VSSTRESN'; if paramcd in ('MEDTAKP' 'MEDTAKT' 'MEDTAK') then SRCVAR =
cats("SUPPVS.QVAL where QNAM='",paramcd,"'");end;
        else do; SRCDOM = DOMAIN_ORIG; SRCSEQ = FASEQ; SRCVAR = 'FASTRESC'; end;

        if PARAMCD in ('PAIN' 'ERYTHDIA' 'SWELLDIA' 'LYMPH') and ATOXGRN>0 then
anl01fl='Y';

        _atpgr=scan(atpt,1,',');
        if _atpgr="DAY 1" then do;atptgr1='DAY 1';atptgr1n=1;end;
        else do;atptgr1=atpt;_atptgr1n=compress(scan(atpt,2,'
'),'.','dk');atptgr1n=input(_atptgr1n,best12.);end;

        if atptgr1n~=. then _atpta=1;
run;

***ANL02FL***;
proc sort /*tagsort*/ data=combine_final;
        by USUBJID PARAMCD descending anl01fl ATPTREF descending _atpta ATPTGR1N
adt atm;
run;
data anl02_;
        set combine_final;
        by USUBJID PARAMCD descending anl01fl ATPTREF descending _atpta ATPTGR1N;
        if first.ATPTREF and atptgr1n>=8 and _atpta=1 and anl01fl='Y' then
anl02fl='Y';
```

```
        if anl02fl='Y';
        keep USUBJID PARAMCD anl01fl ATPTREF anl02fl;
run;
data combine_final;
        merge combine_final anl02_;
        by USUBJID PARAMCD descending anl01fl ATPTREF;

        if faeval in ('STUDY SUBJECT', 'CAREGIVER') then _faeval=1;
        if faeval='' then _faeval=2;
        if faeval='INVESTIGATOR' then _faeval=3;
        if atoxgrn~=. or (paramcd='RASHOCC' and avalc ne '') then _agfl=1;
run;
***ANL03FL***;
proc sort tagsort data=combine_final ;
        by usubjid param atptref atpt _agfl atoxgrn descending _faeval ;
run;
data an03_1;
        set combine_final;
        by usubjid param atptref atpt _agfl atoxgrn  descending _faeval;
        if last.atpt and paramcd ne 'RASHOCC'  and _agfl=1 then ANL03FL='Y';
run;

proc sort tagsort data=an03_1;
        by usubjid param atptref atpt _agfl  avalc  descending _faeval ;
run;
data an03;
        set an03_1;
        by usubjid param atptref atpt _agfl  avalc  descending _faeval;
        if last.atpt and paramcd ='RASHOCC'  and _agfl=1 then ANL03FL='Y';
run;


%macro is_varexist(ds,var );

%local dsid rc ;

%let dsid = %sysfunc(open(&ds));

%if (&dsid) %then %do;
  %if %sysfunc(varnum(&dsid,&var)) %then 1;
  %else 0 ;
  %let rc = %sysfunc(close(&dsid));
%end;
%else 0;

%mend is_varexist;


data adsl;
        set adb.adsl;
```

```
        where dose2fl='N';
        atptref = "Vaccination 2";
        dropfl= "Y";
        keep usubjid  dropfl atptref;
run;
proc sql noprint;
        create table an03_ as
        select a.*,b.dropfl
        from an03 as a
        left join adsl as b
        on a.usubjid=b.usubjid  and a.atptref=b.atptref;
quit;

data an03(drop=dropfl);
        set an03_;
        where dropfl ^= "Y";
run;

proc sort /*tagsort*/ data=an03;
        by USUBJID PARCAT1 PARAMCD ATPTREF adt atm srcseq;
run;

data an04;
        set an03;
        by USUBJID PARCAT1 PARAMCD ATPTREF adt atm srcseq;
        ASEQ + 1;
        if first.usubjid then ASEQ = 1;
        drop _:;
run;

proc sql noprint;
create table __anl04fl as
select distinct USUBJID,  PARAMCD, ATPTREF, min(ATPTGR1N) as _min_atpt,
max(ATPTGR1N) as _max_atpt, 'Y' as ANL04FL
from an04
where ATOXGRN>0 or (paramcd='RASHOCC' and avalc='Y')
group by USUBJID, PARAMCD, ATPTREF
having .<_min_atpt<8 and _max_atpt>=8
order by USUBJID,  PARAMCD, ATPTREF
;
quit;

proc sort data=an04; by  USUBJID  PARAMCD ATPTREF; run;

data an05;
    merge an04 __anl04fl;
    by USUBJID  PARAMCD ATPTREF;
    obs_x+1;
run;
```

```
*-add variables anl05fl anl06fl-added by (b) (6) at 2021-02-25;
data an05_00;
  set an05;
  where parcat1='LOCAL';
run;
*anl05fl;
proc sql ;
  create table an05_01 as
  select distinct usubjid, atptref, paramcd, 'Y' as anl05fl, count(distinct atptn)
as cnt_atptn
  from an05_00
  where atptn in (7, 8) and (atoxgrn>0 or (paramcd='RASHOCC' and avalc='Y'))
  group by usubjid, atptref, paramcd
  having cnt_atptn=2;
quit;
proc sql ;
  create table an05_02 as
  select a.*, b.anl05fl
  from an05_00 as a
  left join an05_01 as b
  on a.usubjid=b.usubjid and a.atptref=b.atptref and a.paramcd=b.paramcd;
quit;
*anl06fl;
data an05_03;
  set an05_02;
  if missing(anl05fl)=1 then anl06fl='Y';
run;
*4. After the ADB is completed, remove records where ATPTN>7 and ATOXGRN<1.;
proc sql ;
  create table an06 as
  select a.*, b.anl05fl, b.anl06fl
  from an05 as a
  left join an05_03 as b
  on a.obs_x=b.obs_x
  having (/*atptn>7 and */atoxgrn<1)=0 or paramcd in ('MEDTAKP' 'MEDTAKT' 'MEDTAK')
or (PARAMCD='RASHOCC' and AVALC='Y');
quit;
*Then find the subject with max(ATPTN)<=7 per ATPTREF per PARAM, remove all these
records.;
proc sql ;
  create table an06_del as
  select distinct usubjid, atptref, paramcd, max(atptn) as atptn_max, 1 as del_flag
  from an06
  group by usubjid, atptref, paramcd
  having atptn_max<=7;
quit;
proc sql ;
  create table an07 as
  select a.*, b.del_flag
  from an06 as a
```

```
  left join an06_del as b
  on a.usubjid=b.usubjid and a.atptref=b.atptref and a.paramcd=b.paramcd
  having del_flag~=1;
quit;

data &DSETNAME;
        set an07;
        if missing(subjid) then
subjid=strip(scan(usubjid,4,'-'))||strip(scan(usubjid,5,'-'));
run;
proc sort /*tagsort*/ data=&DSETNAME(drop=aseq);
        by USUBJID PARCAT1 PARAMCD ATPTREF atptn adt atm srcseq;
run;

data &DSETNAME;
        set &DSETNAME;
        by USUBJID PARCAT1 PARAMCD ATPTREF atptn adt atm srcseq;
        ASEQ + 1;
        if first.usubjid then ASEQ = 1;
        drop _:;
run;

*only keep the usubjid exist in adb.adsl;
%*only_keep_usubjid_from_adsl(update_dataset=&DSETNAME.);

**Generate final dataset by updating certain attributes. Optionally merge common
variables & create sequence variable as needed**;
%trta(dsname=&DSETNAME);


%adam_dataset_update(
        ds=&DSETNAME,
        libin=work,
        libout=output,
        adsllib=adb,
        addcomvar=Y,
        addseq=,
        dropinfmt=Y,
        mapspecfile=&ADAM_SPEC,
        maploc=&ADAM_SPEC_LOC,
        debug=Y
        );

**Generate Value Level Metadata values for the ad<xx> dataset**;
%* dod_vlm(
        type=ADAM,
        selmems=&DSETNAME,
        excmems=,
        specloc=&ADAM_SPEC_LOC,
        specname=&ADAM_SPEC,
```

```
        debug=N
        );

**Generate Enhanced Controlled Terminology for the ad<xx> dataset**;
%* dod_enhcd_ct(
        type=ADAM,
        selmems=&DSETNAME,
        excmems=,
        specloc=&ADAM_SPEC_LOC,
        specname=&ADAM_SPEC,
        map_ct_nm=,
        map_ct_loc=,
        debug=N
        );
```