```
***********************************************************************************
******************************;
** Program Name   : adxb.sas
                           **;
** Date Created   : 01Jun2021
                           **;
** Programmer Name: (b) (4)
                           **;
** Purpose        : Create adxb dataset
                           **;
** Input data     : xb adsl adc19ef
                           **;
** Output data    : adxb.sas7bdat
                           **;
***********************************************************************************
******************************;

%let
oprot=/Volumes/app/cdars/prod/sites/cdars4/prjC459/nda2_unblinded_esub/bla_esub_sdt
m_seq/saseng/cdisc3_0;
%let
prot=/Volumes/app/cdars/prod/sites/cdars4/prjC459/nda2_unblinded_esub/bla_esub_seq_
adam/saseng/cdisc3_0;
libname dataprot "&oprot./data" access=readonly;
libname datvprot "&prot./data_vai";

proc printto print="&prot./analysis/esub/output/adxb.rpt"
        log="&prot./analysis/esub/logs/adxb.log" new;
run;

***********************************************************************************
*******;
* Clean *;
***********************************************************************************
*******;

proc delete data=work._all_;
run;

/******Bring in SDTM XB dataset****/

proc sort data= dataprot.xb out=xb;
        by usubjid xbdtc visitnum visit xbtestcd XBSTRESC;
run;

data xb;
        set xb;
        Length PARAMCD $8. PARAM PARCAT1 $200. PARAMN AVAL AVISITN 8. AVALC $20.
AVISIT $80. ADT 8. ;
        Format ADT date9.;
        PARAM=strip(XBTEST);
        PARAMCD=strip(XBTESTCD);
        PARCAT1=strip(XBCAT);

        IF upcase(XBTESTCD)="WGSCLINE" then PARAMN=1;
        AVAL=.;
        AVALC=strip(XBSTRESC);
        AVISITN=VISITNUM;
        AVISIT=strip(VISIT);
        ADT = input(xbdtc, ?? yymmdd10.);
        Keep Usubjid PARAMCD PARAM PARCAT1 PARAMN AVAL AVALC AVISITN AVISIT ADT
VISIT VISITNUM;
run;

/*********Bring in required Variabled from ADSL dataset*****/

proc sort data=datvprot.adsl(keep=studyid usubjid siteid subjid agegr1n agegr1 sex
race ethnic country armcd arm actarmcd actarm randdt trtsdt trtedt trt01a trt01an
trt01p trt01pn
        vax101dt vax102dt dvstdt unblnddt randfl evaleffl aai1effl aai2effl
mulenrfl phase phasen hivfl) out=adsl;
        by usubjid;
run;

/****Bring in ADC19EF dataset to create required Flags***/

proc sort data=datvprot.adc19ef out=adc19ef;
        by usubjid;
run;

data eff1(keep=usubjid PDP27FL) eff2(keep=usubjid FC19D27 AVISIT) eff3(keep=usubjid
FSEVPD27 AVISIT) eff4(keep=usubjid FSEVCD27 AVISIT) eff5(keep=usubjid FSEVPD1
AVISIT) eff6(keep=usubjid FSEVCD1 AVISIT);
        set adc19ef;
        by usubjid;
        if first.usubjid then output eff1;
        if paramcd in ("C19ONST") and FILOCRFL= "Y" and AVALC="POS" and ILD27FL="Y"
and ((not missing(DVSTDT) and adt <= DVSTDT) or missing(DVSTDT)) and PDRMUPFL = "N"
and EVALEFFL='Y' then do;
                FC19D27="Y";
                output eff2;
        end;
        if paramcd in ("SEVCONST") and FILOCRFL= "Y" and AVALC="POS" and
ILD27FL="Y" and ((not missing(DVSTDT) and adt <= DVSTDT) or missing(DVSTDT)) and
PDRMUPFL = "N" and EVALEFFL='Y' then do;
                FSEVPD27="Y";
                output eff3;
        end;
        if paramcd in ("CDCSONST") and FILOCRFL= "Y" and AVALC="POS" and
ILD27FL="Y" and ((not missing(DVSTDT) and adt <= DVSTDT) or missing(DVSTDT)) and
```

```
CDRMUPFL = "N" and EVALEFFL='Y' then do;
                FSEVCD27="Y";
                output eff4;
        end;
        if paramcd in ("SEVCONST") and FILOCRFL= "Y" and AVALC="POS" and adt >=
VAX101DT and PDRMUPFL = "N" and AAI1EFFL='Y' then do;
                FSEVPD1="Y";
                output eff5;
        end;
        if paramcd in ("CDCSONST") and FILOCRFL= "Y" and AVALC="POS" and adt >=
VAX101DT and CDRMUPFL = "N" and AAI1EFFL='Y' then do;
                FSEVCD1="Y";
                output eff6;
        end;
run;

/***get swab dates and visit from adc19ef dataset for first Occurence****/

proc sort data=adc19ef(where=(paramcd in ("RTCOV2NS" "SARSCOV2") and avalc="POS"))
out=posswab;
        by usubjid avisit;
run;

data eff21(keep=usubjid FC19D27 AVISIT visit adt);
        merge eff2(in=a) posswab;
        by usubjid avisit;
        if a;
run;
proc sort nodupkey;
        by usubjid visit adt;
run;

data eff31(keep=usubjid FSEVPD27 AVISIT visit adt);
        merge eff3(in=a) posswab;
        by usubjid avisit;
        if a;
run;
proc sort nodupkey;
        by usubjid visit adt;
run;

data eff41(keep=usubjid FSEVCD27 AVISIT visit adt);
        merge eff4(in=a) posswab;
        by usubjid avisit;
        if a;
run;
proc sort nodupkey;
        by usubjid visit adt;
run;
```

```
data eff51(keep=usubjid FSEVPD1 AVISIT visit adt);
        merge eff5(in=a) posswab;
        by usubjid avisit;
        if a;
run;
proc sort nodupkey;
        by usubjid visit adt;
run;

data eff61(keep=usubjid FSEVCD1 AVISIT visit adt);
        merge eff6(in=a) posswab;
        by usubjid avisit;
        if a;
run;
proc sort nodupkey;
        by usubjid visit adt;
run;

data feff(drop=avisit);
        merge eff21 eff31 eff41 eff51 eff61;
        by usubjid visit adt;
run;

/***Derive State from Pfizer Corporate Registry Database (based on Investigator
ID)****/

libname PODSPROD oracle path="podsprod.pfizer.com" schema="PODSDAL"
user="pods_cdars" password="Pfizer#8143" dbmax_text= 32767;
proc sql noprint;
        create table person_address
        as select distinct a.state_province_county, a.country_name,
a.address_id,a.city_town_village
        from PODSPROD.pods_ods_address_v a
        where delete_flag='N' ;
quit;

proc sql noprint;
        create table person_address1
        as select a.state_province_county, a.address_id, b.country_iso_code as
country, a.city_town_village as city
        from person_address as a,PODSPROD.pods_ods_country_v b
        where compress(upcase(a.country_name))=compress(upcase(b.country_name)) and
obsolete_country_flg = 'N' ;
quit;

proc sql noprint;
        create table center
        as select ci.person_id, ci.study_id, sa.study_alias, ci.study_site_number
as trialno, p.person_full_name as invname, ci.address_id
        from podsprod.pods_ods_contact_info_v ci, podsprod.pods_ods_person_v p,
podsprod.pods_ods_study_alias_v sa
        where ci.person_id = p.person_id and ci.study_id = sa.study_id and
ci.contact_role='PRINCIPAL INVESTIGATOR' and ci.contact_status='ACTIVE' and
ci.primary_contact='Y' and ci.delete_flag = 'N'
                        and p.delete_flag = 'N' and sa.alias_type = 'PROTOCOL ID'
and sa.delete_flag = 'N' and (sa.study_alias="C4591001" or ci.study_id="C4591001" )
;
quit;

data center1;
        attrib invid length = $7 label="Investigator Identifier" invnam length =
$200 label="Investigator Name" siteid length = $12 studyid length = $20 ;
        set center;
        invid= left(put(person_id,7.));
        studyid=left(study_alias);
        siteid=left(trialno);
        invnam=left(invname);
        keep studyid study_id siteid invid invnam address_id;
run;

proc sql noprint;
        create table country
        as select a.invid, a.invnam, a.studyid, a.siteid, b.country, b.city,
b.state_province_county as state
        from center1 as a, person_address1 as b
        where a.address_id =b.address_id and (a.studyid="C4591001" or
a.study_id="C4591001") ;
quit;

proc sort data=adsl;
        by studyid siteid usubjid;
run;

proc sort data=country out=state(keep=studyid siteid state);
        by studyid siteid ;
run;

data adsl;
        merge adsl(in=a) state;
        by studyid siteid;
        if a;
run;
proc sort data=adsl;
        by usubjid;
run;

/*******merge all datasets to create ADXB dataset****/

data adxb_;
        merge adsl(in=a) xb(in=b) eff1;
        by usubjid;
        if a and b;
        length ADY 8.;
        if trtsdt ^= . and adt ^= . then do;
                ady = adt - trtsdt;
                if adt >= trtsdt then ady = ady + 1;
        end;
run;
proc sort ;
        by usubjid visit adt;
run;

data adxb_1(drop=visit visitnum);
        merge adxb_(in=a) feff;
        by usubjid visit adt;
        if a;
        if FC19D27="" then FC19D27="N";
        if FSEVPD27="" then FSEVPD27="N";
        if FSEVCD27="" then FSEVCD27="N";
        if FSEVPD1="" then FSEVPD1="N";
        if FSEVCD1="" then FSEVCD1="N";
run;

/********Expand all the USA state Abbreviations as per Specifications****/

data adxb_2;
        set adxb_1;
        if upcase(country)="USA" then do;
                if State="AK" then state="ALASKA";
                else if State="AL" then state="ALABAMA";
                else if State="AR" then state="ARKANSAS";
                else if State="AZ" then state="ARIZONA";
                else if State="CA" then state="CALIFORNIA";
                else if State="CO" then state="COLORADO";
                else if State="CT" then state="CONNECTICUT";
                else if State="DE" then state="DELAWARE";
                else if State="FL" then state="FLORIDA";
                else if State="GA" then state="GEORGIA";
                else if State="HI" then state="HAWAII";
                else if State="IA" then state="IOWA";
                else if State="ID" then state="IDAHO";
                else if State="IL" then state="ILLINOIS";
                else if State="IN" then state="INDIANA";
                else if State="KS" then state="KANSAS";
                else if State="KY" then state="KENTUCKY";
                else if State="LA" then state="LOUISIANA";
                else if State="MA" then state="MASSACHUSETTS";
                else if State="MD" then state="MARYLAND";
                else if State="ME" then state="MAINE";
                else if State="MI" then state="MICHIGAN";
                else if State="MN" then state="MINNESOTA";
                else if State="MO" then state="MISSOURI";
                else if State="MS" then state="MISSISSIPPI";
                else if State="MT" then state="MONTANA";
                else if State="NC" then state="NORTH CAROLINA";
                else if State="ND" then state="NORTH DAKOTA";
                else if State="NE" then state="NEBRASKA";
                else if State="NH" then state="NEW HAMPSHIRE";
                else if State="NJ" then state="NEW JERSEY";
                else if State="NM" then state="NEW MEXICO";
                else if State="NV" then state="NEVADA";
                else if State="NY" then state="NEW YORK";
                else if State="OH" then state="OHIO";
                else if State="OK" then state="OKLAHOMA";
                else if State="OR" then state="OREGON";
                else if State="PA" then state="PENNSYLVANIA";
                else if State="RI" then state="RHODE ISLAND";
                else if State="SC" then state="SOUTH CAROLINA";
                else if State="SD" then state="SOUTH DAKOTA";
                else if State="TN" then state="TENNESSEE";
                else if State="TX" then state="TEXAS";
                else if State="UT" then state="UTAH";
                else if State="VA" then state="VIRGINIA";
                else if State="VT" then state="VERMONT";
                else if State="WA" then state="WASHINGTON";
                else if State="WI" then state="WISCONSIN";
                else if State="WV" then state="WEST VIRGINIA";
                else if State="WY" then state="WYOMING";
        end;
run;

/******Assign Labels****/

data adxb;
        retain STUDYID USUBJID SITEID SUBJID AGEGR1N AGEGR1 SEX RACE ETHNIC COUNTRY
STATE ARMCD ARM ACTARMCD ACTARM RANDDT TRTSDT TRTEDT TRT01A TRT01AN TRT01P TRT01PN
                        VAX101DT VAX102DT DVSTDT UNBLNDDT RANDFL EVALEFFL AAI1EFFL
AAI2EFFL MULENRFL PHASE PHASEN HIVFL PARAMN PARAMCD PARAM PARCAT1 AVAL AVALC
AVISITN AVISIT ADT ADY PDP27FL FC19D27 FSEVPD27 FSEVCD27 FSEVPD1 FSEVCD1;
        set adxb_2;
        label PARAMN = "Parameter (N)" PARAMCD = "Parameter Code" PARAM =
"Parameter" PARCAT1 = "Parameter Category 1" AVAL = "Analysis Value" AVALC =
"Analysis Value (C)"
                        AVISITN = "Analysis Visit (N)" AVISIT = "Analysis Visit"
ADT = "Analysis Date" ADY = "Analysis Relative Day" FC19D27 = "First COVID Onset 7D
Post D2 (PD)"
                        FSEVPD27 = "First Sev COVID Onset 7D Post D2 (PD)" FSEVCD27
= "First Sev COVID Onset 7D Post D2 (CDC)" FSEVPD1 = "First Sev COVID Onset Post D1
(PD)"
                        FSEVCD1 = "First Sev COVID Onset Post D1 (CDC)" STATE =
"State" ;
run;

/***output the dataset ****/

proc sort data = adxb out = datvprot.adxb(label = "Sequencing Analysis Dataset");
        by studyid usubjid paramn adt ;
run;

proc printto;
run;
```