\*\*-------------------------------------------------------------------------------------------\*\*

\*\* PROGRAM: D1ADSL.SAS

\*\*

\*\* CREATED: NOVEMBER 2016

\*\*

\*\* PURPOSE: CREATE ANALYSIS DATASET ADSL

\*\*

\*\* PROGRAMMER: A.CHANG

\*\*

\*\* INPUT: SDTM DATA

\*\*

\*\* OUTPUT: ADSLIB.ADSL

\*\*

\*\* PROTOCOL: PROD-124

\*\*

\*\* MODIFIED: DATE BY NOTE

\*\* ---------- ---------- -----------------------------------------------------

\*\*

\*\*-------------------------------------------------------------------------------------------\*\*;

%include msetup;

%let printme = 1 ;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* BRING IN DM SDTM DATA \*\*;

\*\*-------------------------------------------------------------------------------\*\*;

**data** adsl ;

merge sdtmlib.dm (in=indm)

sdtmlib.suppdm ;

by usubjid studyid ;

if indm ;

run ;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* DERIVED VARIABLES FOR TABLE/LISTINGS DISPLAY \*\*;

\*\*-------------------------------------------------------------------------------\*\*;

**data** adsl ;

set adsl ;

length trt01p $**20** ;

if armcd eq '1' then trt01pn = **1** ;

else if armcd eq '2' then trt01pn = **2** ;

if trt01pn eq **1** then trt01p = 'Cohort 1' ;

else if trt01pn eq **2** then trt01p = 'Cohort 2' ;

length sexgr1 $**6** ;

if sex eq 'M' then

do ;

sexgr1n = **1** ;

sexgr1 = 'Male' ;

end ;

else if sex eq 'F' then

do ;

sexgr1n = **2** ;

sexgr1 = 'Female' ;

end ;

else put "WARN" "ING: UNEXPECTED SEX " sex= ;

length racegr1 $**30** ;

racegr1 = propcase(race) ;

if race eq 'CAUCASIAN' then racegr1n = **1** ;

else if race eq 'BLACK' then racegr1n = **2** ;

else if race eq 'HISPANIC' then racegr1n = **3** ;

else if race eq 'ASIAN' then racegr1n = **4** ;

else if race eq 'OTHER' then

do ;

racegr1n = **5** ;

race = 'OTHER, ' || trim(qval) ;

end ;

else put "WARN" "ING: UNEXPECTED RACE " race= ;

run ;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* BRING IN TRTSDT AND TRTEDT \*\*;

\*\*-------------------------------------------------------------------------------\*\*;

**proc** **sort** data=rawlib.vital

out=vs (keep= inv\_no patid proto vsdt);

by patid vsdt ;

**run** ;

**data** vs (keep= usubjid vsdt) ;

set vs ;

by patid vsdt ;

length &pat $**16** ;

&pat = compress(proto) || '-' || '0' || put(inv\_no,**2.**) || '-' || put(patid,**3.**) ;

if first.patid ;

run ;

**data** adsl ;

merge adsl (in=insl)

vs (rename=(vsdt=trtsdt));

by usubjid ;

trtedt = trtsdt ;

format trtsdt trtedt date9. ;

if insl ;

run ;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* BRING IN PROD ADMINISTRATION TIME \*\*;

\*\*-------------------------------------------------------------------------------\*\*;

**proc** **sort** data=sdtmlib.vs (where=(vstpt = '0 min')) nodupkey

out=vs (keep= usubjid vsdtc) ;

by usubjid ;

**run** ;

**data** adsl ;

merge adsl (in=insl)

vs (rename=(vsdtc=trtsdtmc));

by usubjid ;

if trtsdtmc ne '' then

trtsdtm = dhms(input(scan(trtsdtmc,**1**,'T'),YYMMDD10.)

, input(scan(scan(trtsdtmc,**2**,'T'),**1**,':'),**2.**)

, input(scan(scan(trtsdtmc,**2**,'T'),**2**,':'),**2.**)

, **0**) ;

else trtsdtm = **.** ;

format trtsdtm datetime16. ;

if insl ;

run ;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* BRING IN BASELINE WEIGHT AND HEIGHT (ONLY COLLECTED ONCE) \*\*;

\*\*-------------------------------------------------------------------------------\*\*;

**data** weight ;

set sdtmlib.vs (where=(vstestcd='WEIGHT' and vsblfl='Y')) ;

run ;

**data** height ;

set sdtmlib.vs (where=(vstestcd='HEIGHT' and vsblfl='Y')) ;

run ;

**data** adsl ;

merge adsl (in=insl)

weight (keep=usubjid vsstresn rename=(vsstresn=weight))

height (keep=usubjid vsstresn rename=(vsstresn=height)) ;

by usubjid ;

if insl ;

run ;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* CREATE SAFETY POPULATION FLAG \*\*;

\*\*-------------------------------------------------------------------------------\*\*;

**data** ex ;

set rawlib.exposure (where=(proddose ne **.**)) ;

length &pat $**16** ;

&pat = compress(proto) || '-' || '0' || put(inv\_no,**2.**) || '-' || put(patid,**3.**) ;

run ;

**proc** **sort** data=ex nodupkey ;

by usubjid ;

**run** ;

title 'CHECK EX DATA' ;

**proc** **print** data=ex ;

var usubjid proddose ;

where &printme ;

**run** ;

title ;

**data** adsl (drop= proddose) ;

merge adsl (in=insl)

ex (keep=usubjid proddose) ;

by usubjid ;

if proddose ne **.** then saffl = 'Y' ;

else saffl = 'N' ;

if insl ;

run ;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* EVALUABLE POPULATION \*\*;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* NOTE: POPULATION THAT DID NOT HAVE ANY OF THE BELOW PROTOCOL DEVIATIONS \*\*;

\*\* reascd=1 : Entrance Criteria not met \*\*;

\*\* reascd=2 : PROD not administered fully \*\*;

\*\* reascd=3 : Images not obtained \*\*;

\*\* NOTE: USED FASFL (FULL ANALYSIS SET POPULATION FLAG) PER ADAMIG v1.1 \*\*;

**data** ts ;

set rawlib.summary (where=(reascd not in (**1**,**2**,**3**))) ;

length &pat $**16** ;

&pat = compress(proto) || '-' || '0' || put(inv\_no,**2.**) || '-' || put(patid,**3.**) ;

run ;

**proc** **sort** data=ts nodupkey ;

by usubjid ;

**run** ;

**data** adsl ;

merge adsl (in=insl)

ts (in=ints keep= usubjid) ;

by usubjid ;

if ints then fasfl = 'Y' ;

else fasfl = 'N' ;

if insl ;

run ;

title 'CHECK FASFL' ;

**proc** **print** data=adsl ;

var usubjid fasfl ;

where &printme ;

**run** ;

title ;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* OUTPUT FINAL DATA \*\*;

\*\*-------------------------------------------------------------------------------\*\*;

options replace ;

**proc** **sql**;

create table adslib.adsl (label='Subject Level Analysis Dataset') as

select studyid

, usubjid

, subjid

, country

, siteid

, rfstdtc

, trtsdt label='Date of First Exposure to Treatment '

, trtedt label='Date of Last Exposure to Treatment '

, trtsdtm label='Datetime of First Exposure to Treatment '

, brthdtc

, age

, ageu

, sex

, sexgr1 label='Pooled Sex Group 1 '

, sexgr1n label='Pooled Sex Group 1 (N) '

, race

, racegr1 label='Pooled Race Group 1 '

, racegr1n label='Pooled Race Group 1 (N) '

, arm

, armcd

, trt01p label='Planned Treatment for Period 01 '

, trt01pn label='Planned Treatment for Period 01 (N) '

, weight label='Baseline Weight (kg) '

, height label='Baseline Height (cm) '

, saffl label='Safety Population Flag '

, fasfl label='Full Analysis Set Population Flag '

from adsl

order by usubjid ;

**quit**;

options noreplace;

title "CHECK ANALYSIS DATA" ;

**proc** **print** data=adslib.adsl ;

where &printme ;

**run** ;

title ;

\*\*-------------------------------------------------------------------------------------------\*\*

\*\* PROGRAM: D2ADAE.SAS

\*\*

\*\* CREATED: NOVEMBER 2016

\*\*

\*\* PURPOSE: CREATE ANALYSIS DATASET ADAE

\*\*

\*\* PROGRAMMER: A.CHANG

\*\*

\*\* INPUT: SDTM AE DATA, ADSL

\*\*

\*\* OUTPUT: ADSLIB.ADAE

\*\*

\*\* PROTOCOL: PROD-124

\*\*

\*\* MODIFIED: DATE BY NOTE

\*\* ---------- ---------- -----------------------------------------------------

\*\*

\*\*-------------------------------------------------------------------------------------------\*\*;

%include msetup;

%let printme = 1 ;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* BRING IN AE AND DM SDTM DATA \*\*;

\*\*-------------------------------------------------------------------------------\*\*;

**data** adae ;

merge sdtmlib.ae (in=inae drop= domain studyid)

adslib.adsl ;

by usubjid ;

if inae ;

run ;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* DERIVED VARIABLES FOR TABLE/LISTINGS \*\*;

\*\*-------------------------------------------------------------------------------\*\*;

**data** adae ;

set adae ;

\*\*-- CONVERT CHARACTER DATE/TIME TO NUMERIC --\*\*;

if aestdtc ne '' then

aestdtm = dhms(input(scan(aestdtc,**1**,'T'),YYMMDD10.)

, input(scan(scan(aestdtc,**2**,'T'),**1**,':'),**2.**)

, input(scan(scan(aestdtc,**2**,'T'),**2**,':'),**2.**)

, **0**) ;

else aestdtm = **.** ;

if aeendtc ne '' then

aendtm = dhms(input(scan(aeendtc,**1**,'T'),YYMMDD10.)

, input(scan(scan(aeendtc,**2**,'T'),**1**,':'),**2.**)

, input(scan(scan(aeendtc,**2**,'T'),**2**,':'),**2.**)

, **0**) ;

else aendtm = **.** ;

format aestdtm aendtm datetime16. ;

\*\*-- CALCULATE AE DURATION --\*\*;

adurn = aendtm - aestdtm ;

aduru = 'SECONDS' ;

\*\*-- ASEV AND AREL --\*\*;

asev = propcase(aesev) ;

arel = propcase(aerel) ;

run ;

title 'CHECK DATE/TIME CONVERSION AND DURATIONS' ;

**proc** **print** data=adae ;

var aestdtc aestdtm aeendtc aendtm adurn ;

where &printme ;

**run** ;

title ;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* OUTPUT FINAL DATA \*\*;

\*\*-------------------------------------------------------------------------------\*\*;

options replace ;

**proc** **sql**;

create table adslib.adae (label='Adverse Event Analysis Dataset ') as

select studyid

, usubjid

, subjid

, siteid

, aeseq

, aeterm

, aedecod

, aebodsys

, aestdtc

, aestdtm label='Analysis Start Date/Time '

, aeendtc

, aendtm label='Analysis End Date/Time '

, trtsdtm

, adurn label='AE Duration (N) '

, aduru label='AE Duration Units '

, aesevn

, aesev

, asev label='Analysis Severity/Intensity '

, aereln

, aerel

, arel label='Analysis Causality '

, aeacn

, aeacnoth

, aeser

from adae

order by usubjid ;

**quit**;

options noreplace;

title "CHECK ANALYSIS DATA" ;

**proc** **print** data=adslib.adae ;

where &printme ;

**run** ;

title ;

\*\*-------------------------------------------------------------------------------------------\*\*

\*\* PROGRAM: D2ADVS.SAS

\*\*

\*\* CREATED: NOVEMBER 2016

\*\*

\*\* PURPOSE: CREATE ANALYSIS DATASET ADVS

\*\*

\*\* PROGRAMMER: A.CHANG

\*\*

\*\* INPUT: SDTM VS DATA, ADSL

\*\*

\*\* OUTPUT: ADSLIB.ADVS

\*\*

\*\* PROTOCOL: PROD-124

\*\*

\*\* MODIFIED: DATE BY NOTE

\*\* ---------- ---------- -----------------------------------------------------

\*\*

\*\*-------------------------------------------------------------------------------------------\*\*;

%include msetup;

%let printme = 1 ;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* BRING IN SDTM DATA \*\*;

\*\*-------------------------------------------------------------------------------\*\*;

**data** advs ;

merge sdtmlib.vs (in=invs drop= studyid)

adslib.adsl ;

by usubjid ;

if invs ;

run ;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* DERIVED VARIABLES FOR TABLE/LISTINGS DISPLAY \*\*;

\*\*-------------------------------------------------------------------------------\*\*;

**data** advs ;

set advs ;

length vtpt $**40** ;

if vstpt = 'SCREENING & BASELINE' then vtpt = 'Screening' ;

else if vstpt = '-5 min' or vstpt = '0 min' then vtpt = vstpt ;

else if vstpt = '1 HOUR AFTER INFUSION PERIOD #2' then vtpt = '+105 min' ;

else if vstpt = 'STUDY DAY 2' then vtpt = 'Study Day 2' ;

else vtpt = '+' || vstpt ;

length vstresc $**10** ;

vstresc = trim(left(put(vsstresn,**5.1**))) ;

run ;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* CALCULATE CHANGE IN BASELINE AND ASSIGN CLINICALLY SIGNIFICANT FLAG \*\*;

\*\*-------------------------------------------------------------------------------\*\*;

\*\*-- ADD BASELINE VALUE --\*\*;

**proc** **sort** data=advs (where=(vsblfl='Y'))

out=bl (keep = usubjid vstestcd vsstresn rename=(vsstresn=vsbl)) ;

by usubjid vstestcd ;

**run** ;

**proc** **sort** data=advs ;

by usubjid vstestcd ;

**run** ;

**data** advs ;

merge advs (in=invs)

bl ;

by usubjid vstestcd ;

if invs ;

run ;

**proc** **sort** data=advs ;

by usubjid vsseq ;

**run** ;

\*\*-- CALCULATE CHANGE IN BASELINE --\*\*;

**data** advs ;

set advs ;

length vschgblc $**10** ;

if vtpt in ('Screening','-5 min','0 min') then

do ;

vschgbl = **.** ;

vschgblc = 'n/a' ;

end ;

else

do ;

vschgbl = vsstresn - vsbl ;

vschgblc = trim(left(put(vschgbl,**5.1**))) ;

end ;

run ;

\*\*-- ASSIGN CLINICAL IMPORTANCE FLAG --\*\*;

**data** advs ;

set advs ;

if vschgblc ne 'n/a' then

do ;

if vstestcd='SYSBP' then

do ;

if vsstresn gt **200** and vschgbl ge **20** then clinsig='I' ;

else if vsstresn lt **90** and vschgbl le -**20** then clinsig='D' ;

end ;

else if vstestcd='DIABP' then

do ;

if vsstresn gt **120** and vschgbl ge **10** then clinsig='I' ;

else if vsstresn lt **60** and vschgbl le -**10** then clinsig='D' ;

end ;

else if vstestcd = 'OXYSAT' then

do ;

if vsstresn lt **90** and vschgbl le -**5** then clinsig='D' ;

end ;

else if vstestcd='PULSE' then

do ;

if vsstresn gt **120** and vschgbl ge **10** then clinsig='I' ;

else if vsstresn lt **45** and vschgbl le -**10** then clinsig='D' ;

end ;

end ;

if clinsig in ('I','D') then vschgblc = trim(left(put(vschgbl,**5.1**))) || clinsig ;

run ;

\*\*-------------------------------------------------------------------------------\*\*;

\*\* OUTPUT FINAL DATA \*\*;

\*\*-------------------------------------------------------------------------------\*\*;

options replace ;

**proc** **sql**;

create table adslib.advs (label='Vital Signs Analysis Dataset') as

select studyid

, usubjid

, subjid

, siteid

, country

, age

, sex

, race

, arm

, armcd

, trt01p

, trt01pn

, saffl

, fasfl

, visitnum

, visit

, vsdtc

, vsseq

, vstpt

, vtpt label='Analysis Planned Time Point Name '

, vstestcd

, vstest

, vsorres

, vsorresu

, vsstresc

, vsstresn

, vstresc label='Analysis Character Result/Finding '

, vsstresu

, vsblfl

, vsbl label='Baseline Value '

, vschgbl label='Analysis Change from Baseline (N) '

, vschgblc label='Analysis Change from Baseline '

, clinsig label='Clinically Significant Flag '

from advs

order by usubjid, vsseq ;

**quit**;

options noreplace;

title "CHECK ANALYSIS DATA" ;

**proc** **print** data=adslib.advs ;

where &printme ;

**run** ;

title ;