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

\*\* PROGRAM: TAE.SAS

\*\*

\*\* CREATED: NOVEMBER 2016

\*\*

\*\* PURPOSE: CREATE TABLE 7.2.1 - ADVERSE EVENTS BY BODY SYSTEM

\*\* TABLE 7.2.2 - ADVERSE EVENTS BY BODY SYSTEM RELATED

\*\*

\*\* PROGRAMMER: A.CHANG

\*\*

\*\* INPUT: ADAE, ADSL DATA

\*\*

\*\* OUTPUT: TABLE 7.2.1, TABLE 7.2.2

\*\*

\*\* PROTOCOL: PROD-124

\*\*

\*\* MODIFIED: DATE BY NOTE

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

\*\*

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

%include msetup ;

%let printme = 1 ;

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

\*\* BRING IN ANALYSIS DATA \*\*;

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

**%macro** tae (num=) ;

data adae ;

merge adslib.adae (in=inae where=(aestdtm > trtsdtm))

adslib.adsl ;

\*\* TAKE ONLY AE REPORTED AFTER PROD ADMINISTERED \*\*;

by usubjid ;

if &num = **1** then if inae ;

else if &num = **2** then if inae and aerel ne 'UNRELATED' ;

run ;

\*\*-- EXPAND ADAE DATASET TO A NORMALIZED STRUCTURE (OUTPUT RECORDS FOR EACH CATEGORY) --\*\*;

data adae ;

set adae ;

length aetext $**400** ;

\*\* OUTPUT FOR PREFERRED TERM \*\*;

seq = **3** ;

aetext = trim(left(propcase(aedecod))) ;

output ;

\*\* OUTPUT FOR BODY SYSTEM ROWS \*\*;

seq = **2** ;

aetext = trim(left(propcase(aebodsys))) ;

output ;

\*\* OUTPUT FOR ANY AE ROW \*\*;

seq = **1** ;

aebodsys = 'AAAA' ;

aetext = 'Subjects with Adverse Event(s)^{super d}' ;

output ;

run ;

\*\*----- OUTPUT FOR TOTALS COLUMN -----\*\*;

data adae ;

set adae ;

output ; \*\* OUTPUT ONCE FOR EACH SUBJECT \*\*;

trt01pn = **0** ; \*\* RE-ASSIGN TOTAL, THEN OUTPUT AGAIN \*\*;

trt01p = 'All Subjects' ;

output ;

run ;

\*\*----- POPULATION COUNTS -----\*\*;

data popcnt ;

set adslib.adsl (where=(saffl eq 'Y')) ;

output ; \*\* OUTPUT ONCE FOR EACH SUBJECT \*\*;

trt01pn = **0** ; \*\* RE-ASSIGN TOTAL, THEN OUTPUT AGAIN \*\*;

trt01p = 'Total' ;

output ;

run ;

proc sort data=popcnt out=popcnt nodupkey ;

by trt01pn subjid ;

run ;

proc freq data=popcnt noprint ;

tables trt01pn\*trt01p /out=popcnt (drop=percent) ;

run ;

\*\*----- ASSIGN POPULATION COUNT INTO MACRO VARIABLES -----\*\*;

data \_null\_ ;

set popcnt ;

call symput('popcnt'||compress(put(trt01pn,**8.**)),compress(put(count,**8.**))) ;

call symput('trt'||compress(put(trt01pn,**8.**)),trim(left(trt01p))) ;

run ;

proc print data=popcnt ;

title "CHECK DENOMINATORS" ;

where &printme ;

run ;

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

\*\* FREQUENCY COUNTS \*\*;

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

\*\*-- DEDUPE BY SUBJECT/EVENTS --\*\*;

proc sort data=adae out=subjects nodupkey ;

by usubjid trt01pn aebodsys seq aetext ;

run ;

proc freq data=subjects noprint ;

tables trt01pn\*aebodsys\*seq\*aetext /out=subjects (drop=percent) ;

run ;

proc print data=subjects ;

format aebodsys aetext $20. ;

where &printme ;

run ;

data subjects ;

merge subjects

popcnt (rename=(count=popcnt)) ;

by trt01pn ;

length col $**200** ;

if n(count,popcnt) eq **2** then

do ;

if seq = **1** then col = compress(put(count,**8.**)) ;

else col = compress(put(count,**8.**))||'

('||compress(put(count/popcnt\***100**,**8.1**))||')' ;

end ;

run ;

proc sort data=subjects out=subjects ;

by aebodsys seq aetext ;

run ;

proc transpose data=subjects out=adae (drop=\_name\_) ;

by aebodsys seq aetext ;

id trt01pn ;

var col ;

run ;

data adae ;

set adae (rename=(\_0=col0 \_1=col1 \_2=col2 )) ;

if aebodsys = '' then delete ;

run ;

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

\*\* FORMAT DATA FOR REPORT \*\*;

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

data adae ;

set adae ;

if col0 eq ' ' then col0 = '0' ;

if col1 eq ' ' then col1 = '0' ;

if col2 eq ' ' then col2 = '0' ;

if seq eq **3** then aetext = "^R'\li360 '" || aetext ;

run ;

proc print data=adae ;

title "CHECK COMBINED STATS" ;

var aebodsys seq aetext col0 col1 col2 ;

where &printme ;

run ;

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

\*\* CREATE MACRO TITLE AND FOOTNOTE A \*\*;

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

data \_null\_ ;

if &num = **1** then

do ;

call symput('title','Adverse Events^{super a} by Body System') ;

call symput('footnote'

, '^{super a} Includes all adverse events reported after start of

PROD administration.') ;

end ;

else if &num = **2** then

do ;

call symput('title','Adverse Events by Body System Related^{super a}') ;

call symput('footnote'

, '^{super a} Includes all adverse events reported after start of

PROD administration that are considered definitely, probably, or

possibly related to PROD.') ;

end ;

run ;

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

\*\* CREATE REPORT \*\*;

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

\*\*----- RTF SETUP -----\*\*;

options nodate nonumber orientation=landscape missing=' ';

ods listing close ;

ods escapechar='^' ;

ods rtf style=TStyleRTF file="&opath.\T7-2-&num..rtf" ;

\*\*----- TITLES/FOOTNOTES -----\*\*;

title1 j=left "CM Pharmaceuticals, Inc." j=right 'Page ^{pageof}' ;

title2 j=left "Protocol PROD-124" j=right "&sysdate9" ;

title3 j=center "Table 7.2.&num" ;

title5 j=center "&title" ;

title6 j=left "Study Population: Safety" ;

footnote1 "^{style [outputwidth=100% bordertopcolor=black bordertopwidth=1pt]}" ;

footnote2 h=**10**pt j=left "&footnote" ;

footnote3 h=**10**pt j=left "^{super b} Subjects who had more than one event within a body system

were counted once." ;

footnote4 h=**10**pt j=left "^{super c} Subjects who had more than one event assigned to the same

preferred term were counted once." ;

footnote5 h=**10**pt j=left "^{super d} Subjects who had more than one event were counted once." ;

footnote6 ;

footnote7 j=left "Data Source: ADAE, ADSL" j=right "Program: tae.sas" ;

\*\*----- REPORT DEFINITION -----\*\*;

proc report data=adae missing nowindows center split='|' style(report)=[outputwidth=**9.0**in] ;

column aebodsys seq aetext col0-col2 ;

define aebodsys /order noprint ;

define seq /order noprint ;

define aetext /"^R'\ql\li360\fi-360 MedDRA Body System'^{super b}"

"Preferred Term^{super c}" style(header)=[just=left] style(column)=[just=left

cellwidth=**3.00**in] ;

define col0 /"&trt0|(N=&popcnt0)|n (%)" style(column)=[just=center cellwidth=**1.2**in] ;

define col1 /"&trt1|(N=&popcnt1)|n (%)" style(column)=[just=center cellwidth=**1.2**in] ;

define col2 /"&trt2|(N=&popcnt2)|n (%)" style(column)=[just=center cellwidth=**1.2**in] ;

compute before aebodsys ;

line ' ' ;

endcomp ;

run ;

\*\*----- CLOSE RTF AND RESET TITLES/FOOTNOTES -----\*\*;

ods rtf close ;

ods listing ;

options date number ;

title ;

footnote ;

**%mend** tae ;

%***tae***(num=**1**)

%***tae***(num=**2**)

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

\*\* PROGRAM: TDM.SAS

\*\*

\*\* CREATED: NOVEMBER 2016

\*\*

\*\* PURPOSE: CREATE TABLE 2 - DEMOGRAPHIC AND BASELINE CHARACTERISTICS

\*\*

\*\* PROGRAMMER: A.CHANG

\*\*

\*\* INPUT: ADSL DATA

\*\*

\*\* OUTPUT: TABLE 2

\*\*

\*\* PROTOCOL: PROD-124

\*\*

\*\* MODIFIED: DATE BY NOTE

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

\*\*

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

%include msetup ;

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

\*\* BRING IN ANALYSIS DATA \*\*;

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

**data** adsl ;

set adslib.adsl (where=(saffl eq 'Y')) ;

run ;

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

\*\* EXPAND ADSL DATASET TO A NORMALIZED STRUCTURE (OUTPUT RECORDS FOR EACH CATEGORY) \*\*;

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

**data** adsl ;

set adsl ;

length parcat1 param $**200** ;

parcat1n = **1** ;

parcat1 = 'Age (yrs)' ;

if age gt **.z** then aval = age ;

else aval = **.** ;

if aval gt **.z** then output ;

parcat1n = **2** ;

parcat1 = 'Sex n (%)' ;

paramn = sexgr1n ;

param = sexgr1 ;

aval = **.** ;

output ;

parcat1n = **3** ;

parcat1 = 'Race n (%)' ;

paramn = racegr1n ;

param = racegr1 ;

aval = **.** ;

output ;

parcat1n = **4** ;

parcat1 = 'Weight (kg)' ;

if weight gt **.z** then aval = weight ;

else aval = **.** ;

if aval gt **.z** then output ;

parcat1n = **5** ;

parcat1 = 'Height (cm)' ;

if height gt **.z** then aval = height ;

else aval = **.** ;

if aval gt **.z** then output ;

run ;

\*\*-- OUTPUT FOR TOTALS COLUMN --\*\*;

**data** adsl ;

set adsl ;

output ; \*\* OUTPUT ONCE FOR EACH SUBJECT \*\*;

trt01pn = **3** ; \*\* RE-ASSIGN FOR SAFETY POPULATION, THEN OUTPUT AGAIN \*\*;

trt01p = 'All Subjects' ;

output ;

if fasfl = 'Y' then

do ;

trt01pn = **4** ; \*\* RE-ASSIGN FOR EVALUABLE POPULATION, THEN OUTPUT AGAIN \*\*;

trt01p = 'All' ;

output ;

end ;

run ;

\*\*-- POPULATION COUNTS --\*\*;

**proc** **sort** data=adsl out=popcnt nodupkey ;

by trt01pn subjid ;

**run** ;

**proc** **freq** data=popcnt noprint ;

tables trt01pn\*trt01p /out=popcnt (drop=percent) ;

**run** ;

\*\*-- ASSIGN POPULATION COUNT INTO MACRO VARIABLES --\*\*;

**data** \_null\_ ;

set popcnt ;

call symput('popcnt'||compress(put(trt01pn,**8.**)),compress(put(count,**8.**))) ;

call symput('trt'||compress(put(trt01pn,**8.**)),trim(left(trt01p))) ;

run ;

title "CHECK DENOMINATORS" ;

**proc** **print** data=popcnt ;

where &printme ;

**run** ;

title ;

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

\*\* FREQUENCY COUNTS \*\*;

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

**proc** **freq** data=adsl (where=(aval=**.**)) noprint ;

tables trt01pn\*parcat1n\*parcat1\*paramn\*param /out=freqs (drop=percent) ;

**run** ;

\*\*-- BRING IN DENOMINATORS --\*\*;

**data** freqs ;

merge freqs popcnt (rename=(count=popcnt)) ;

by trt01pn ;

length col $**200** ;

if n(count,popcnt) eq **2** then col = compress(put(count,**8.**))||'('

||compress(put(count/popcnt\***100**,**8.1**))||')' ;

run ;

title "CHECK FREQUENCY DATA" ;

**proc** **print** data=freqs ;

where &printme ;

**run** ;

title ;

\*\*-- CREATE TABLE SHELL --\*\*;

**data** shell (drop = i j) ;

length parcat1 param col $**200** trt01p $**20** ;

count = **0** ;

col = '0' ;

do i = **1** to **4** by **1** ;

trt01pn = i ;

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

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

else if trt01pn = **3** then trt01p = 'All Subjects' ;

else if trt01pn = **4** then trt01p = 'All' ;

parcat1n = **2** ;

parcat1 = 'Sex n (%)' ;

do j=**1** to **2** by **1** ;

paramn = j ;

if paramn = **1** then param = 'Male' ;

else if paramn = **2** then param = 'Female' ;

output ;

end ;

parcat1n = **3** ;

parcat1 = 'Race n (%)' ;

do j=**1** to **5** by **1** ;

paramn = j ;

if paramn = **1** then param = 'Caucasian' ;

else if paramn = **2** then param = 'Black' ;

else if paramn = **3** then param = 'Hispanic' ;

else if paramn = **4** then param = 'Asian' ;

else if paramn = **5** then param = 'Other' ;

output ;

end ;

end ;

run ;

**proc** **sort** data=shell ;

by trt01pn parcat1n parcat1 paramn param ;

**run** ;

\*\*-- MERGE FREQUENCY DATA WITH TABLE SHELL --\*\*;

**data** freqs ;

set freqs

shell ;

by trt01pn parcat1n parcat1 paramn param ;

if first.paramn ;

run ;

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

\*\* SUMMARY STATS \*\*;

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

**proc** **sort** data=adsl (where=(aval ne **.**))

out=summary ;

by trt01pn trt01p parcat1n parcat1;

**run** ;

**proc** **means** data=summary noprint ;

by trt01pn trt01p parcat1n parcat1 ;

var aval ;

output out=summary n=n mean=mean std=std median=median min=min max=max ;

**run** ;

\*\*-- FORMAT SUMMARY STATS TO CHARACTER VARIABLES --\*\*;

**data** summary (keep= trt01pn trt01p parcat1n parcat1 paramn param col) ;

set summary ;

length param col $**200** ;

paramn = **1** ;

param = 'n' ;

col = compress(put(n,**8.**)) ;

output ;

paramn = **2** ;

param = 'Mean' ;

col = compress(put(mean,**8.1**)) ;

output ;

paramn = **3** ;

param = 'SD' ;

col = compress(put(std,**8.1**)) ;

output ;

paramn = **4** ;

param = 'Median' ;

col = compress(put(median,**8.1**)) ;

output ;

paramn = **5** ;

param = 'Range (Min, Max)' ;

col = '(' || compress(put(min,**8.**)) || ', ' || compress(put(max,**8.**)) || ')' ;

output ;

run ;

title "CHECK SUMMARY STATS" ;

**proc** **print** data=summary ;

where &printme ;

**run** ;

title ;

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

\*\* COMBINE SUMMARY STATS AND FREQUENCY COUNTS \*\*;

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

**data** stats ;

set freqs summary ;

run ;

title "CHECK DATA PRIOR TO TRANSPOSE" ;

**proc** **print** data=stats ;

var parcat1n parcat1 paramn param trt01pn col ;

where &printme ;

**run** ;

title ;

\*\*----- TRANSPOSE DATA -----\*\*;

**proc** **sort** data=stats out=stats ;

by parcat1n paramn ;

**run** ;

**proc** **transpose** data=stats out=stats (drop=\_name\_) ;

by parcat1n parcat1 paramn param ;

id trt01pn ;

var col ;

**run** ;

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

\*\* FORMAT DATA FOR REPORT \*\*;

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

**data** stats ;

set stats (rename= (\_1=col3 \_2=col4 \_3=col1 \_4=col2)) ;

if parcat1n in (**1**:**4**) then pagevar = **1** ;

else if parcat1n = **5** then pagevar = **2** ;

run ;

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

\*\* CREATE REPORT \*\*;

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

\*\*----- RTF SETUP -----\*\*;

options nodate nonumber orientation=landscape missing=' ';

ods listing close ;

ods escapechar='^' ;

ods rtf style=TStyleRTF file="&opath.\T2.rtf" ;

\*\*----- TITLES/FOOTNOTES -----\*\*;

title1 j=left "CM Pharmaceuticals, Inc." j=right 'Page ^{pageof}' ;

title2 j=left "Protocol PROD-124" j=right "&sysdate9" ;

title3 j=center "Table 2" ;

title5 j=center "Demographic and Baseline Characteristics" ;

title7 j=left "Study Population: Safety" ;

footnote1 "^{style [outputwidth=100% bordertopcolor=black bordertopwidth=1pt]}" ;

footnote2 j=left "Data Source: ADSL" j=right "Program: tdm.sas" ;

\*\*----- REPORT DEFINITION -----\*\*;

**proc** **report** data=stats missing nowindows center split='|' style(report)=[outputwidth=**9.0**in] ;

column pagevar parcat1n parcat1 paramn param col1 ('^S={borderbottomcolor=black

borderbottomwidth=2} Evaluable Population' col2-col4) ;

define pagevar /order noprint ;

define parcat1n /order noprint ;

define parcat1 /order noprint ;

define paramn /order noprint ;

define param /"^R'\ql Characteristic'" style(column)=[leftmargin=**0.15**in just=left

cellwidth=**2.00**in] ;

define col1 /"&trt3|(N=&popcnt3)" style(column)=[just=center cellwidth=**1.5**in] ;

define col2 /"&trt4|(N=&popcnt4)" style(column)=[just=center cellwidth=**1.5**in] ;

define col3 /"&trt1|(N=&popcnt1)" style(column)=[just=center cellwidth=**1.5**in] ;

define col4 /"&trt2|(N=&popcnt2)" style(column)=[just=center cellwidth=**1.5**in] ;

break after pagevar / page ;

compute before parcat1 ;

length text $**400** ;

text = trim(left(parcat1)) ;

line @**2** text $400. ;

endcomp ;

compute before parcat1n ;

line ' ' ;

endcomp ;

**run** ;

\*\*----- CLOSE RTF AND RESET TITLES/FOOTNOTES -----\*\*;

ods rtf close ;

ods listing ;

options date number ;

title ;

footnote ;

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

\*\* PROGRAM: TVS.SAS

\*\*

\*\* CREATED: NOVEMBER 2016

\*\*

\*\* PURPOSE: CREATE TABLES 9.1 - VITAL SIGNS - SYSTOLIC BLOOD PRESSURE (MM HG)

\*\* 9.2 - VITAL SIGNS - DIASTOLIC BLOOD PRESSURE (MM HG)

\*\* 9.3 - VITAL SIGNS - HEART RATE (BEATS/MIN)

\*\* 9.4 - VITAL SIGNS - OXYGEN SATURATION (%)

\*\*

\*\* PROGRAMMER: A.CHANG

\*\*

\*\* INPUT: ADVS, ADSL DATA

\*\*

\*\* OUTPUT: TABLES 9.1-9.4

\*\*

\*\* PROTOCOL: PROD-124

\*\*

\*\* MODIFIED: DATE BY NOTE

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

\*\*

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

%include msetup ;

**%macro** tvs (num=) ;

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

\*\* BRING IN ANALYSIS DATA \*\*;

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

data vs ;

set adslib.advs ;

\*\*-- SET PARCAT1N --\*\*;

if vtpt = 'Screening' then parcat1n = **1** ;

else if vtpt = '-5 min' then parcat1n = **2** ;

else if vtpt = '0 min' then parcat1n = **3** ;

else if vtpt = 'Baseline^{super b}' then parcat1n = **4** ;

else if vtpt = '+5 min' then parcat1n = **5** ;

else if vtpt = '+10 min' then parcat1n = **6** ;

else if vtpt = '+15 min' then parcat1n = **7** ;

else if vtpt = '+20 min' then parcat1n = **8** ;

else if vtpt = '+25 min' then parcat1n = **9** ;

else if vtpt = '+30 min' then parcat1n = **10** ;

else if vtpt = '+35 min' then parcat1n = **11** ;

else if vtpt = '+40 min' then parcat1n = **12** ;

else if vtpt = '+45 min' then parcat1n = **13** ;

else if vtpt = '+105 min' then parcat1n = **14** ;

else if vtpt = 'Study Day 2' then parcat1n = **15** ;

\*\*-- SUBSET BASED ON TABLE NUMBER --\*\*;

if &num = **1** then if vstestcd in ('SYSBP') ;

else if &num = **2** then if vstestcd in ('DIABP') ;

else if &num = **3** then if vstestcd in ('PULSE') ;

else if &num = **4** then if vstestcd in ('OXYSAT') ;

run ;

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

\*\* REMOVE DATA OUTSIDE PLANNED TIMEPOINTS \*\*;

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

\*\* NOTE: REMOVED DATA WITH DATA OUTSIDE PLANNED TIMEPOINTS \*\*;

\*\* (PATID = 118 HAD EXTRA DATA AT 50 MIN, \*\*;

\*\* PATID = 125 HAD EXTRA DATA AT 50 MIN, 55 MIN) \*\*;

data vs dropped ;

set vs ;

if vstpt in ('SCREENING & BASELINE','-5 min','0 min','5 min','10 min','15 min'

,'20 min','25 min','30 min','35 min','40 min','45 min'

,'1 HOUR AFTER INFUSION PERIOD #2','STUDY DAY 2') then output vs ;

else output dropped ;

run ;

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

\*\* BASELINE AND TABLE SHELL \*\*;

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

data vsbl ;

set vs (where=(vsblfl='Y')) ;

run ;

\*\*-- SUMMARY STATS FOR BASELINE --\*\*;

proc sort data=vsbl (where=(vsstresn ne **.**))

out=vsbl ;

by trt01pn trt01p parcat1n vtpt ;

run ;

proc means data=vsbl noprint ;

by trt01pn trt01p parcat1n vtpt ;

var vsstresn ;

output out=vsbl n=n mean=mean std=std median=median min=min max=max ;

run ;

\*\*-- CREATE TABLE SHELL --\*\*;

data shell ;

set vsbl (drop= \_type\_ \_freq\_) ;

do i=**1** to **16** by **1** ;

if i in (**1**:**4**) then parcat1n = i ;

else if i = **5** then

do ;

parcat1n = **4.1** ;

n = **.** ;

end ;

else if i in (**6**:**16**) then parcat1n = i-**1** ;

if parcat1n = **1** then vtpt = 'Screening' ;

else if parcat1n = **2** then vtpt = '-5 min' ;

else if parcat1n = **3** then vtpt = '0 min' ;

else if parcat1n = **4** then vtpt = 'Baseline^{super b}' ;

else if parcat1n = **5** then vtpt = '+5 min' ;

else if parcat1n = **6** then vtpt = '+10 min' ;

else if parcat1n = **7** then vtpt = '+15 min' ;

else if parcat1n = **8** then vtpt = '+20 min' ;

else if parcat1n = **9** then vtpt = '+25 min' ;

else if parcat1n = **10** then vtpt = '+30 min' ;

else if parcat1n = **11** then vtpt = '+35 min' ;

else if parcat1n = **12** then vtpt = '+40 min' ;

else if parcat1n = **13** then vtpt = '+45 min' ;

else if parcat1n = **14** then vtpt = '+105 min' ;

else if parcat1n = **15** then vtpt = 'Study Day 2' ;

else vtpt = '' ;

output ;

end ;

run ;

\*\*-- BLANK LINE BETWEEN SCREENING & BASELINE AND SCHEDULED TIMEPOINTS --\*\*;

data shell ;

set shell ;

if parcat1n = **4.1** then

do ;

mean=**.** ; std=**.** ; median=**.** ; min=**.** ; max=**.** ;

end ;

run ;

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

\*\* SCHEDULED TIMEPOINT \*\*;

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

data vsst ;

set vs (where= (parcat1n in (**5**:**15**))) ;

run ;

\*\*-- SUMMARY STATS FOR SCHEDULED TIMEPOINT --\*\*;

proc sort data=vsst (where=(vsstresn ne **.**))

out=vsst ;

by trt01pn trt01p parcat1n vtpt ;

run ;

proc means data=vsst noprint ;

by trt01pn trt01p parcat1n vtpt ;

var vsstresn ;

output out=vsst n=n mean=mean std=std median=median min=min max=max ;

run ;

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

\*\* CHANGE FROM BASELINE \*\*;

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

data vschg ;

set vs (where= (parcat1n in (**5**:**15**))) ;

run ;

\*\*-- SUMMARY STATS FOR CHANGE FROM BASELINE --\*\*;

proc sort data=vschg (where=(vschgbl ne **.**))

out=vschg ;

by trt01pn trt01p parcat1n vtpt ;

run ;

proc means data=vschg noprint ;

by trt01pn trt01p parcat1n vtpt ;

var vschgbl ;

output out=vschg n=n mean=mean std=std median=median min=min max=max ;

run ;

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

\*\* MERGE TOGETHER \*\*;

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

data summary ;

merge shell

vsst (drop= \_type\_ \_freq\_

rename=(n=n1 mean=mean1 std=std1 median=median1 min=min1 max=max1))

vschg (drop= \_type\_ \_freq\_

rename=(n=n2 mean=mean2 std=std2 median=median2 min=min2 max=max2)) ;

by trt01pn trt01p parcat1n vtpt ;

\*\*-- COHORT AND SCHEDULED TIMEPOINT --\*\*;

col1 = compress(put(trt01pn,**8.**)) ;

col2 = vtpt ;

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

if parcat1n in (**1**:**4**) then n0 = n ;

else if parcat1n in (**5**:**15**) then n0 = n1 ;

col3 = compress(put(n0,**2.**)) ;

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

col4 = compress(put(mean,**5.1**)) ;

col5 = compress(put(std,**5.1**)) ;

col6 = compress(put(median,**5.**)) ;

if n(min,max)=**2** then col7 = '(' || compress(put(min,**8.**)) || ', '

|| compress(put(max,**8.**)) || ')' ;

\*\*-- SCHEDULED TIMEPOINT --\*\*;

col8 = compress(put(mean1,**5.1**)) ;

col9 = compress(put(std1,**5.1**)) ;

col10 = compress(put(median1,**5.**)) ;

if n(min1,max1)=**2** then col11 = '(' || compress(put(min1,**8.**)) || ', '

|| compress(put(max1,**8.**)) || ')' ;

\*\*-- CHANGE FROM BASELINE --\*\*;

col12 = compress(put(mean2,**5.1**)) ;

col13 = compress(put(std2,**5.1**)) ;

col14 = compress(put(median2,**5.**)) ;

if n(min2,max2)=**2** then col15 = '(' || compress(put(min2,**8.**)) || ', '

|| compress(put(max2,**8.**)) || ')' ;

\*\*-- EMPTY COLUMN FOR TABLE DISPLAY --\*\*;

\_empty = ' ' ;

run ;

proc sort data = summary ;

by trt01pn parcat1n ;

run ;

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

\*\* FORMAT DATA FOR REPORT \*\*;

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

\*\*-- ASSIGN TABLE TITLE MACRO --\*\*;

data \_null\_ ;

if &num = **1** then call symput('title'

,'Vital Signs - Systolic Blood Pressure (mm Hg)') ;

else if &num = **2** then call symput('title'

,'Vital Signs - Diastolic Blood Pressure (mm Hg)') ;

else if &num = **3** then call symput('title','Vital Signs - Heart Rate (beats/min)') ;

else if &num = **4** then call symput('title','Vital Signs - Oxygen Saturation (%)') ;

run ;

\*\*-- ASSIGN SAFFL POPULATION COUNT MACRO --\*\*;

proc freq data=adslib.adsl (where=(saffl='Y')) noprint ;

tables studyid /out=popcnt (drop=percent) ;

run ;

data \_null\_ ;

set popcnt ;

call symput('popcnt',compress(put(count,**8.**))) ;

run ;

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

\*\* CREATE REPORT \*\*;

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

\*\*----- RTF SETUP -----\*\*;

options nodate nonumber orientation=landscape missing=' ';

ods listing close ;

ods escapechar='^' ;

ods rtf style=TStyleRTF file="&opath.\T9-&num..rtf" ;

\*\*----- TITLES/FOOTNOTES -----\*\*;

title1 j=left "CM Pharmaceuticals, Inc." j=right 'Page ^{pageof}' ;

title2 j=left "Protocol PROD-124" j=right "&sysdate9" ;

title3 j=center "Table 9.&num" ;

title5 j=center "&title" ;

title7 j=left "Study Population: Safety (N = &popcnt)" ;

footnote1 "^{style [outputwidth=100% bordertopcolor=black bordertopwidth=1pt]}" ;

footnote2 h=**10**pt j=left "^{super a} Cohort 1 Sequence: Bag in Infusion 1

/ Bottle in Infusion 2"

", Cohort 2 Sequence: Bottle in Infusion 1 / Bag in Infusion 2." ;

footnote3 h=**10**pt j=left "^{super b} Baseline is the last value prior to the start of PROD." ;

footnote4 h=**10**pt j=left "Note: Summary statistics at each scheduled timepoint include those

subjects with a baseline value and a value at the scheduled timepoint. " ;

footnote6 j=left "Data Source: ADVS, ADSL" j=right "Program: tvs.sas" ;

\*\*----- REPORT DEFINITION -----\*\*;

proc report data=summary missing nowindows center split='|' style(report)=[outputwidth=**9.0**in] ;

column col1-col3 ('^S={borderbottomcolor=black borderbottomwidth=2} Baseline^{super b}'

col4-col7) \_empty

('^S={borderbottomcolor=black borderbottomwidth=2} Scheduled Timepoint' col8-col11 )

\_empty ('^S={borderbottomcolor=black borderbottomwidth=2} Change from Baseline' col12-col15) ;

define \_empty / display " " style(column)=[just=center cellwidth=**0.1**in] ;

define col1 / order "Cohort^{super a}" style(column)=[just=center cellwidth=**0.4**in] ;

define col2 /"Scheduled|Timepoint" style(column)=[just=center cellwidth=**0.6**in] ;

define col3 /"N" style(column)=[just=center cellwidth=**0.4**in] ;

define col4 /"Mean" style(column)=[just=center cellwidth=**0.4**in] ;

define col5 /"SD" style(column)=[just=center cellwidth=**0.4**in] ;

define col6 /"Median" style(column)=[just=center cellwidth=**0.4**in] ;

define col7 /"(Min, Max)" style(column)=[just=center cellwidth=**0.6**in] ;

define col8 /"Mean" style(column)=[just=center cellwidth=**0.4**in] ;

define col9 /"SD" style(column)=[just=center cellwidth=**0.4**in] ;

define col10 /"Median" style(column)=[just=center cellwidth=**0.4**in] ;

define col11 /"(Min, Max)" style(column)=[just=center cellwidth=**0.6**in] ;

define col12 /"Mean" style(column)=[just=center cellwidth=**0.4**in] ;

define col13 /"SD" style(column)=[just=center cellwidth=**0.4**in] ;

define col14 /"Median" style(column)=[just=center cellwidth=**0.4**in] ;

define col15 /"(Min, Max)" style(column)=[just=center cellwidth=**0.6**in] ;

break after col1 / page ;

compute before col1 ;

line '' ;

endcomp ;

run ;

\*\*----- CLOSE RTF AND RESET TITLES/FOOTNOTES -----\*\*;

ods rtf close ;

ods listing ;

options date number ;

title ;

footnote ;

**%mend** tvs ;

%***tvs***(num=**1**)

%***tvs***(num=**2**)

%***tvs***(num=**3**)

%***tvs***(num=**4**)