

```
*****,  
** Program Name : adae-s091-d1-cut-ped-saf.sas **;  
** Date Created : 22Mar2021 **,  
** Programmer Name : (b) (4), (b) **,  
** Purpose : Create adae-s091-d1-cut-ped-saf **,  
** Input data : adae **,  
** Output data : adae-s091-d1-cut-ped-saf.html **,  
*****,
```

```
options mprint mlogic symbolgen mprint symbolgen mlogic nocenter missing=" ";
```

```
**Setup the environment**;
```

```
%let prot=/Volumes/app/cdars/prod/sites/cdars4/prjC459/nda2_unblinded_esub/euaext_esub_adam/saseng/cdisc3_0;  
libname datvprot "&prot./data_vai" access=readonly;
```

```
%let codename=adae-s091-d1-cut-ped-saf;  
%let outlog=&prot./analysis/esub/logs/&codename..log;  
%let outtable=&prot./analysis/esub/output/&codename..html;
```

```
proc printto log="&outlog." new;  
run;
```

```
%let cutoff=13MAR2021;
```

```
data g_adsl_dsin;  
set datvprot.adsl;
```

```
if trt01an=8 and agegr4n=1 then  
trtarn=1;  
else if trt01an=8 and agegr4n=2 then  
trtarn=2;  
else if trt01an=9 and agegr4n=1 then  
trtarn=3;  
else if trt01an=9 and agegr4n=2 then  
trtarn=4;  
trtar=trt01a;  
where SAFFL eq 'Y' and AGEGR1N=1 and HIVFL ne 'Y' and MULENRFL ne "Y";  
run;
```

```
data g_a_dsin;  
set datvprot.adae;
```

```
if trt01an=8 and agegr4n=1 then  
trtarn=1;  
else if trt01an=8 and agegr4n=2 then  
trtarn=2;  
else if trt01an=9 and agegr4n=1 then  
trtarn=3;  
else if trt01an=9 and agegr4n=2 then  
trtarn=4;  
trtar=trt01a;  
analysis_subset='Y';  
where AECAT='ADVERSE EVENT' and saffl="Y" and VPHASEN in (1, 2, 3, 99)  
and (astdt ne . and astdt<="&cutoff."d) and (UNBLNDDT=. or UNBLNDDT > ASTDT);
```

```

run;

data g_adsl_dsin;
  set g_adsl_dsin;

  if TRT01AN in (8) then
    do;
      newtrtn=1;
      newtrt=coalescec("BNT162b2 (30 (*ESC*){unicode 03BC}g)", TRT01A);
      output;
    end;

  if TRT01AN in (9) then
    do;
      newtrtn=2;
      newtrt=coalescec("Placebo", TRT01A);
      output;
    end;
run;

data g_a_dsin;
  set g_a_dsin;

  if TRT01AN in (8) then
    do;
      newtrtn=1;
      newtrt=coalescec("BNT162b2 (30 (*ESC*){unicode 03BC}g)", TRT01A);
      output;
    end;

  if TRT01AN in (9) then
    do;
      newtrtn=2;
      newtrt=coalescec("Placebo", TRT01A);
      output;
    end;
run;

proc format;
  value catlbl 1="Any event" 2="Any serious adverse event" 3="Severe"
  4="Related(*ESC*){super c}" 5="Life-threatening"
  6="Any adverse event leading to withdrawal" 7="Death";
  value scatlbl 101, 102, 106="Related(*ESC*){super c}" 201, 202, 206="Severe"
  301, 302, 306="Life-threatening";
run;

proc sort data=g_adsl_dsin out=_ds1;
  by usubjid newtrtn;
run;

proc sort data=g_a_dsin out=_ds2;
  by usubjid newtrtn;
run;

```

```

data final;
  merge _ds1(in=d1) _ds2(in=d2);
  by usubjid newtrtn;

  if d1;

proc sort;
  by newtrtn usubjid;
run;

data final_;
  set final;
  by newtrtn usubjid;
  _uniqid=_n_;
run;

data _data1;
  set final_;
  where (NEWTRTN is not missing);

proc sort;
  by NEWTRTN USUBJID;
run;

data _data1;
  retain _trt 0;
  length _str $200;
  _datasrt=1;
  set _data1 end=eof;
  by NEWTRTN USUBJID;
  drop _str;
  _str=' ';
  _lastby=1;
  _dummyby=0;

  if first.NEWTRTN then
    do;

      if not missing(NEWTRTN) then
        do;
          _trt=_trt + 1;
          end;
          _str=NEWTRTN;

          if _trt > 0 then
            call symput('_trtlb'||compress(put(_trt, 4.)), trim(left(_str)));
          end;
        do;

proc sql;
  create table trtbign as select distinct _trt, newtrtn, compress(put(count(*),
  5.)) as bign from (select distinct USUBJID, _trt, newtrtn from _data1 where
  NEWTRTN is not missing) group by _trt;
quit;

```

```

proc sort data=_data1 out=_bydat1(keep=_datasrt _dummyby) nodupkey;
  by _datasrt;
run;

data _bydat1;
  set _bydat1 end=eof;
  by _datasrt;
  retain _preby 0;
  drop _preby;
  _byvar1=0;

  if eof then
    do;
      call symput("_preby1", compress(put(_byvar1, 4.)));

      if 0=0 then
        output;
    end;
run;

data _bydat1;
  set _bydat1;
  by _datasrt;
  length _bycol _byindnt $50 _bylast $10;
  _bycol=" ";
  _byindnt=" ";
  _bylast=" ";

proc sort;
  by _datasrt;
run;

proc sort data=_data1;
  by _trt usubjid aeterm descending ATOXGRN;
run;

data tab1;
  set _data1;
  where analysis_subset='Y' and aeterm ne " ";
  by _trt usubjid aeterm descending ATOXGRN;

  if last.usubjid then
    do;
      catvar=1;
      output;
    end;

  if last.ATOXGRN then
    do;
      _catvar=1;
      output;
    end;
run;

```

```
data rel_tab1;
  set _data1;
  by _trt usubjid aeterm descending ATOXGRN;
  where analysis_subset='Y' and aeterm ne " and upcase(AREL)='RELATED';
```

```
  if last.usubjid then
    do;
      catvar=1+100;
      output;
    end;
```

```
  if last.ATOXGRN then
    do;
      _catvar=1+100;
      output;
    end;
```

```
run;
```

```
data sev_tab1;
  set _data1;
  by _trt usubjid aeterm descending ATOXGRN;
  where analysis_subset='Y' and aeterm ne " and ATOXGRN=3;
```

```
  if last.usubjid then
    do;
      catvar=1+200;
      output;
    end;
```

```
  if last.ATOXGRN then
    do;
      _catvar=1+200;
      output;
    end;
```

```
run;
```

```
data lif_tab1;
  set _data1;
  by _trt usubjid aeterm descending ATOXGRN;
  where analysis_subset='Y' and aeterm ne " and (atoxgr="GRADE 4");
```

```
  if last.usubjid then
    do;
      catvar=1+300;
      output;
    end;
```

```
  if last.ATOXGRN then
    do;
      _catvar=1+300;
      output;
    end;
```

```
run;
```

```

data tab2;
  set _data1;
  where analysis_subset='Y' and aeser='Y';
  by _trt usubjid aeterm descending ATOXGRN;

  if last.usubjid then
    do;
      catvar=2;
      output;
    end;

  if last.ATOXGRN then
    do;
      _catvar=2;
      output;
    end;
run;

data rel_tab2;
  set _data1;
  by _trt usubjid aeterm descending ATOXGRN;
  where analysis_subset='Y' and aeser='Y' and upcase(AREL)='RELATED';

  if last.usubjid then
    do;
      catvar=2+100;
      output;
    end;

  if last.ATOXGRN then
    do;
      _catvar=2+100;
      output;
    end;
run;

data sev_tab2;
  set _data1;
  by _trt usubjid aeterm descending ATOXGRN;
  where analysis_subset='Y' and aeser='Y' and ATOXGRN=3;

  if last.usubjid then
    do;
      catvar=2+200;
      output;
    end;

  if last.ATOXGRN then
    do;
      _catvar=2+200;
      output;
    end;
run;

```

```

data lif_tab2;
  set _data1;
  by _trt usubjid aeterm descending ATOXGRN;
  where analysis_subset='Y' and aeser='Y' and (atoxgr="GRADE 4");

  if last.usubjid then
    do;
      catvar=2+300;
      output;
    end;

  if last.ATOXGRN then
    do;
      _catvar=2+300;
      output;
    end;
run;

data tab6;
  set _data1;
  where analysis_subset='Y' and (upcase(aeacn)='DRUG WITHDRAWN' or aesubjdc='Y');
  by _trt usubjid aeterm descending ATOXGRN;

  if last.usubjid then
    do;
      catvar=6;
      output;
    end;

  if last.ATOXGRN then
    do;
      _catvar=6;
      output;
    end;
run;

data rel_tab6;
  set _data1;
  by _trt usubjid aeterm descending ATOXGRN;
  where analysis_subset='Y' and (upcase(aeacn)='DRUG WITHDRAWN' or aesubjdc='Y')
    and upcase(AREL)='RELATED';

  if last.usubjid then
    do;
      catvar=6+100;
      output;
    end;

  if last.ATOXGRN then
    do;
      _catvar=6+100;
      output;
    end;

```

```

run;

data sev_tab6;
  set _data1;
  by _trt usubjid aeterm descending ATOXGRN;
  where analysis_subset='Y' and (upcase(aeacn)='DRUG WITHDRAWN' or aesubjdc='Y')
    and ATOXGRN=3;

  if last.usubjid then
    do;
      catvar=6+200;
      output;
    end;

  if last.ATOXGRN then
    do;
      _catvar=6+200;
      output;
    end;
run;

```

```

data lif_tab6;
  set _data1;
  by _trt usubjid aeterm descending ATOXGRN;
  where analysis_subset='Y' and (upcase(aeacn)='DRUG WITHDRAWN' or aesubjdc='Y')
    and (atoxgr="GRADE 4");

  if last.usubjid then
    do;
      catvar=6+300;
      output;
    end;

  if last.ATOXGRN then
    do;
      _catvar=6+300;
      output;
    end;
run;

```

```

data tab7;
  set _data1;
  where analysis_subset='Y' and (upcase(AEOUT)="FATAL");
  by _trt usubjid aeterm descending ATOXGRN;

  if last.usubjid then
    do;
      catvar=7;
      output;
    end;

  if last.ATOXGRN then
    do;
      _catvar=7;

```



```

    output;
end;
run;

data rel_tab7;
  set _data1;
  by _trt usubjid aeterm descending ATOXGRN;
  where analysis_subset='Y' and (upcase(AEOUT)="FATAL") and
    upcase(AREL)="RELATED";

  if last.usubjid then
    do;
      catvar=7+100;
      output;
    end;

  if last.ATOXGRN then
    do;
      _catvar=7+100;
      output;
    end;
run;

data sev_tab7;
  set _data1;
  by _trt usubjid aeterm descending ATOXGRN;
  where analysis_subset='Y' and (upcase(AEOUT)="FATAL") and ATOXGRN=3;

  if last.usubjid then
    do;
      catvar=7+200;
      output;
    end;

  if last.ATOXGRN then
    do;
      _catvar=7+200;
      output;
    end;
run;

data lif_tab7;
  set _data1;
  by _trt usubjid aeterm descending ATOXGRN;
  where analysis_subset='Y' and (upcase(AEOUT)="FATAL") and (atoxgr="GRADE 4");

  if last.usubjid then
    do;
      catvar=7+300;
      output;
    end;

  if last.ATOXGRN then
    do;

```

```

        _catvar=7+300;
        output;
    end;
run;

data _data1;
    set _data1(in=a) tab1 sev_tab1 rel_tab1 lif_tab1 tab2 sev_tab2 rel_tab2
        lif_tab2 tab6 sev_tab6 rel_tab6 lif_tab6 tab7;

    if a then
        do;
            catvar=0;
            _catvar=0;
        end;

    if _catvar ne . then
        catvar=.;
run;

proc format cntlout=tmpfmt;
    select catlbl;
run;

data _anall;
    length CATVAR 8;
    set _data1;
    where same and CATVAR is not missing;
    _blcksrt=0;
    _cnt=1;
    _cat=1;

    if _trt <=0 then
        delete;
    output;
run;

proc sort data=_anall;
    by _datasrt _blcksrt CATVAR _trt _cat;
run;

proc sort data=_anall out=_temp91 nodupkey;
    by _datasrt _blcksrt _cat CATVAR _trt USUBJID;
run;

proc freq data=_temp91;
    format CATVAR;
    tables _datasrt*_blcksrt*_cat * CATVAR * _trt / sparse norow nocol nopercnt
        out=_pct1(drop=percent);
run;

Data temp;
    catvar=1;
    output;
    catvar=101;

```

```

output;
catvar=201;
output;
catvar=301;
output;
catvar=2;
output;
catvar=102;
output;
catvar=202;
output;
catvar=302;
output;
catvar=6;
output;
catvar=106;
output;
catvar=206;
output;
catvar=306;
output;
catvar=7;
output;
run;

proc sql;
  create table temp2 as select distinct a._datasrt , a._blcksrt, a._cat, a._trt,
    b.* from _pct1 as a left join temp as b on 1;
quit;

proc sql;
  create table _pct2 as select a.*, coalesce(b.count, 0) as count from temp2 as
    a left join _pct1 as b on a._datasrt=b._datasrt and a._blcksrt=b._blcksrt and
    a._cat=b._cat and a._trt=b._trt and a.catvar=b.catvar;
quit;

data rep1;
  set _pct2;
  length _rlabel $200. _cvalue $50.;

  if catvar<100 then
    do;
      _rlabel=strip(put(CATVAR, catlbl.));

      /* grp=1; */
    end;
  else
    do;
      _rlabel=repeat(byte(160), 2)|| strip(put(CATVAR, scatlbl.));

      /* grp=2; */
    end;

  if catvar in (1, 101, 201, 301) then

```

```

_fixvar=1;
else if catvar in (2, 102, 202, 302) then
_fixvar=2;
else if catvar in (6, 106, 206, 306) then
_fixvar=3;
else if catvar in (7) then
_fixvar=4;
_cvalue=strip(put(count, best.));
run;

proc sql;
create table rep2 as select a.*, b.bign, b.newtrt from rep1 as a left join
trtbign as b on a._trt=b._trt;
quit;

data rep2;
set rep2;
length _cpct $40.;

if bign>0 then
do;
percent=count / bign * 100;

if percent > 0 then
do;

if round(percent, 0.1) GE 0.1 then
_cpct="(*ESC*){nbspspace 1}{||strip(put(percent, 5.1))||}";
else
_cpct="(*ESC*){nbspspace 1}{(0.0)";
_cvalue=trim(_cvalue)||_cpct;
end;
end;
newtrt=strip(newtrt)||'| (N(*ESC*){super a}="||strip(bign)||)";
_dummy=1;

proc sort;
by _cat _fixvar catvar _rwlable _trt newtrt _cvalue;
run;

```

```

options topmargin=0.75in bottommargin=0.75in leftmargin=0.75in
rightmargin=0.75in;

```

```

;
options orientation=LANDSCAPE papersize="LETTER";
ods escapechar="~";
option nobyline;
title1 "Number (%) of Subjects Reporting at Least 1 Adverse Event From Dose 1 Through Cutoff Date (13MAR2021),
Subjects 12 Through 15 Years of Age (*ESC*){unicode 2013} Safety Population";
footnote1 "Note: Adverse events that occurred on the day of or after subjects were unblinded are excluded from this
summary.";
footnote2 "%nrbquote(a.~{nbspspace 5}N = number of subjects in the specified group. This value is the denominator for
the percentage calculations.)";
footnote3 "%nrbquote(b.~{nbspspace 5}n = Number of subjects reporting at least 1 occurrence of the specified event
category. For "any

```

```
event,  
" n = number of subjects reporting at least 1 occurrence of any event.);"  
footnote4 "%nrbrquote(c.~{nbsp 5}Assessed by the investigator as related to investigational product.);"  
ods html file="&outtable.";
```

```
proc report data=rep2 nowd list missing contents="" split="|";  
  column _cat _fixvar catvar  
  _rwlabel ("~S={just=center}Vaccine Group (as Administered)~{line}" newtrt,  
  (_cvalue _dummy) );  
  define _cat / group noprint;  
  define _fixvar / group order=internal noprint;  
  define catvar / group order=internal noprint;  
  define _rwlabel / group "Adverse Event" order=data style(column)={just=left  
  width=65mm} style(header)={just=left} left;  
  define newtrt / across nozero "" style(column)={width=35mm leftmargin=12px}  
  style(header)={just=center} center;  
  define _cvalue / display nozero "n(*ESC*){super b} (%)"  
  style(column)={width=35mm leftmargin=12px} style(header)={just=center} center;  
  define _dummy / sum noprint;  
  compute before _cat;  
    line @1 " ~n";  
  endcomp;  
  compute after _cat;  
    line " ~n";  
  endcomp;  
run;  
  
ods html close;  
  
proc printto;  
run;
```