

```

*****
** Program Name   : adae-s092-cut1-ped6.sas                **
** Date Created  : 15Nov2021                               **
** Programmer Name : (b) (4), (b)                          **
** Purpose       : Create adae-s092-cut1-ped6              **
** Input data    : adae adsl                               **
** Output data   : adae-s092-cut1-ped6.html               **
*****
options mprint mlogic symbolgen mprint symbolgen mlogic nocenter missing=" ";
**Setup the environment**
%let
bprot=/Volumes/app/cdars/prod/sites/cdars4/prjC459/nda2_unblinded_esub/sbla1215_esub_adam/saseng/cdisc3_0/;
%let prot=/Volumes/app/cdars/prod/sites/cdars4/prjC459/nda2_unblinded_esub/sbla1215_esub_adam/saseng/cdisc3_0/;
%let codename=adae-s092-cut1-ped6;

libname datvprot "&bprot.data_vai" access=readonly;
%let outlog=&prot./analysis/eSUB/logs/&codename..log;
%let outtable=&prot./analysis/eSUB/output/&codename..html;

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

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="Y" and AGEGR4N=1 and HIVFL ne "Y" and trt02an=8 and trt01an=9 and VAX201DT>. and
X1CSRDT>;
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 agegr4n=1 and (vphasen >=5 and vphasen ne 99)
and .<VAX201DT<=ASTDT<=X1CSRDT and HIVFL ne "Y" and trt02an=8 and trt01an=9;
run;

data g_adsl_dsin;
  set g_adsl_dsin;

  if TRT02AN in (8) then
    do;
      newtrtn=1;
      newtrt=coalescec("BNT162b2 (30 (*ESC*){unicode 03BC}g)", TRT012A);
      output;
    end;

  if TRT02AN in (9) then
    do;
      newtrtn=2;
      newtrt=coalescec("Placebo", TRT02A);
      output;
    end;
run;

data g_a_dsin;
  set g_a_dsin;

  if TRT02AN in (8) then
    do;
      newtrtn=1;
      newtrt=coalescec("BNT162b2 (30 (*ESC*){unicode 03BC}g)", TRT02A);
      output;
    end;

  if TRT02AN in (9) then
    do;
      newtrtn=2;
      newtrt=coalescec("Placebo", TRT02A);
      output;
    end;
run;

proc format;
  value catlbl
  1="Any event"
  2="Any serious adverse event"
  3="Severe"
  4="Related(*ESC*){super f}"
  5="Life-threatening"
  6="Any nonserious adverse event"
  7="Any adverse event leading to withdrawal"
  8="Death";
  value scatlbl
  101, 102, 106, 107="Related(*ESC*){super f}"
  201, 202, 206, 207="Severe"
  301, 302, 306, 307="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 _basetemplate(compress=no);
    length _varname $8 _cvalue $35 _direct $20 _vrlabel $200 _rwlabel
        _colabel $800 _datatyp $5 _module $8 _pr_lbl $ 200;
    array _c _character_;
    delete;
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=NEWTRT;

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

proc sql;
    create table trtbign as select distinct _trt, newtrt, compress(put(count(*),
        5.)) as bign, coalesce(sum(FPX1CUT)/(365.25*100), 0) as tenum from (select
        distinct USUBJID, _trt, newtrt, FPX1CUT 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 (aeser in ('N'));
  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 (aeser in (' ' 'N'))
    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 (aeser in (' ' 'N'))
    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 (aeser in (' ' 'N'))
    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(aeacn)='DRUG WITHDRAWN' or aesubjdc='Y');
    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(aeacn)='DRUG WITHDRAWN' or aesubjdc='Y')
        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(aeacn)='DRUG WITHDRAWN' or aesubjdc='Y')
        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(aeacn)='DRUG WITHDRAWN' or aesubjdc='Y')
    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 tab8;
  set _data1;
  where analysis_subset='Y' and (upcase(AEOUT)="FATAL");
  by _trt usubjid aeterm descending ATOXGRN;

  if last.usubjid then
    do;
      catvar=8;
      output;
    end;

  if last.ATOXGRN then
    do;
      _catvar=8;
      output;
    end;
run;

data rel_tab8;
  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=8+100;
      output;
    end;

```

```

if last.ATOXGRN then
  do;
    _catvar=8+100;
    output;
  end;
run;

data sev_tab8;
  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=8+200;
      output;
    end;

  if last.ATOXGRN then
    do;
      _catvar=8+200;
      output;
    end;
run;

data lif_tab8;
  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=8+300;
      output;
    end;

  if last.ATOXGRN then
    do;
      _catvar=8+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 sev_tab7 rel_tab7 lif_tab7 tab8;

  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;
catvar=107;
output;
catvar=207;
output;
catvar=307;
output;
catvar=8;
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.)); */
      _rlabel="{nbspace 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, 107, 207, 307) then

```

```

        _fixvar=4;
    else if catvar in (8) then
        _fixvar=5;
    _cvalue=strip(put(count, best.));
run;

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

data rep2;
    set rep2;
    length _cpct _cvalue2 $40. CNP_CI $100.;
    newtrt=strip(newtrt)||" (N(*ESC*){super a}="||strip(bign)||", TE(*ESC*){super b}="||strip(put(tenum,
        8.1)) || ")";

    percent = count / bign * 100;
    if percent > 0 then do;
        if round(percent, 0.1) GE 0.1 then _cpct = "(*ESC*){nbspace 1}("||strip(put(percent,5.1))||)";
        else _cpct = "(*ESC*){nbspace 1}(0.0)";
        _cvalue = trim(_cvalue)||_cpct;
    end;

    _cvalue2=strip(put((count/TENUM), 8.1));

    if count ne 0 then
        lcl=(cinv(0.05/2, 2*count))/(2*TENUM);
    else
        lcl=0;
    ucl=(cinv(1-0.05/2, 2*(count+1)))/(2*TENUM);
    CNP_CI=strip("(" || strip(put(lcl, 8.1))
        || ",(*ESC*){nbspace 1}" || strip(put(ucl, 8.1)) || ")");
    _dummy=1;

proc sort;
    by _cat _fixvar catvar _rlabel _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 "Incidence Rates of at Least 1 Adverse Event From Dose 3 to Data Cutoff Date (02SEP2021) (*ESC*){unicode
2013} ";
title2 "Open-Label Follow-up Period (*ESC*){unicode 2013}";
title3 "Subjects Who Originally Received Placebo and Then Received BNT162b2 After Unblinding (*ESC*){unicode
2013}";
title4 "Phase 2/3 Subjects 12 Through 15 Years of Age (*ESC*){unicode 2013} Safety Population";
footnote1 "Note: Dose 3 = First dose of BNT162b2 (30 (*ESC*){unicode 03BC}g).";
footnote2 "a.~{nbspace 5}N = number of subjects in the specified group. This value is the denominator for the
FDA-CBER-2022-5812-0071624

```

percentage calculations.";

footnote3 "b.~{nbspspace 5}TE = total exposure time in 100 person-years across all subjects in the specified group. Exposure time for a subject is the time from Dose 3 to data cutoff date. This value is the denominator for the incidence rate calculation.";

footnote4 "%nrbquote(c.~{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.)";

footnote5 "d.~{nbspspace 5}Incidence rate (IR) is calculated as number of subjects reporting the event/total exposure time in 100 person-years (PY) across all subjects in the specified group.";

footnote6 "e.~{nbspspace 5}2-sided CI based on Poisson distribution.";

footnote7 "f.~{nbspspace 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
    _rwlable ("~S={just=center}Vaccine Group (as Administered)~{line}" newtrt,
      (_cvalue _cvalue2 cnp_ci _dummy) );
  define _cat / group noprint;
  define _fixvar / group order=internal noprint;
  define catvar / group order=internal noprint;
  define _rwlable / 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 c} (%)"
    style(column)={leftmargin=12px} style(header)={just=center} center;
  define _cvalue2 / display nozero
    "IR(*ESC*){super d}"
    style(column)={leftmargin=12px} style(header)={just=center} center;
  define cnp_ci / display nozero "(95% CI(*ESC*){super e})"
    style(column)={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;
```