```
/*************************************************************************************/
/***  This program calculates fatal crashes and fatalities for FHWA's 3 focus areas    ***/
/***  Each of the 3 focus areas was redefined in 2014, which led to "new" being added  ***/
/***  in front of file names.  See FARS manual for complete list of variable           ***/
/***  definitions and categories:                                                      ***/
/***  http://www-nrd.nhtsa.dot.gov/CMSWeb/listpublications.aspx?Id=J&ShowBy=DocType    ***/
/*************************************************************************************/

/*************************************************************************************/
/***  Various comments inserted below are intended to assist with later programming    ***/
/***  and to help with understanding certain lines of the current program.             ***/
/*************************************************************************************/

/*** Rwd is Roadway Departure crashes   ***/
/*** Int is Intersection crashes        ***/
/*** Ped is Pedestrian and Bicycle type crashes ***/

%macro iter;

/*** Macro statement runs multiple years efficiently.  Choose begin and end years for A and B***/

%DO i=&A %TO &B;
/*%do i = 2010 %to 2012;*/

/*** Change years as needed for future programs, such as 2011 to 2013 ***/
/*** Note that many FARS variables changed category definitions in 2010 ***/
/************************/
/*New Definition, start */
/************************/
/*** New definitions below for 1- Roadway Departure crashes, then 2- intersection crashes, and then 3-
Pedestrian-Bicycle crashes ***/
/*** Three separate sections are placed consecutively, each producing usable data files for
appropriate crash types.  No other meaningful disaggregations emerged from the remaining crash data.
For this reason, only three crash types have been defined. ***/
/*****************************/
/*new RwD Definition, start */
/*****************************/
```

```
PROC SORT DATA=fars&i..vevent (KEEP=ST_CASE VEH_NO VEVENTNUM SOE) OUT=VSOE; /*** VSOE is a working
file for vehicle sequence of events ***/
  BY ST_CASE VEH_NO VEVENTNUM;
RUN;

PROC TRANSPOSE DATA=fars&i..vevent OUT=TRANS_VSOE (DROP=_NAME_ _LABEL_) PREFIX=SEQ;
  BY ST_CASE VEH_NO;
  ID VEVENTNUM;
  VAR SOE;
  FORMAT SOE;
RUN;

DATA NewUS_VSEQ1;
      SET trans_vsoe (KEEP=ST_CASE VEH_NO SEQ1);
    IF
       seq1 IN (17,19:43,46,52,53,57,59,63,64,65,67,68,69,71); /* See FARS manual for attribute
definitions, some attributes added in 2010 and 2012 */
RUN;

DATA preNewpre_USDEP10;  /*** Merge working vehicle event files with FARS accident file   ***/
      MERGE fars&i..ACCIDENT (KEEP=year ST_CASE state RELJCT1 RELJCT2 harm_ev)
          NewUS_VSEQ1 (KEEP=ST_CASE VEH_NO SEQ1 in=A);
    BY ST_CASE;
    IF A;

      newrwd_ind = 1;
      rwd_seq1 = seq1;
RUN;

data Newpre_USDEP10;  /*** Merging vehicle file variables with selected accident and vehicle sequence
of events variables***/
      merge preNewpre_USDEP10 (in=a)
              fars&i..Vehicle (keep=st_case veh_no m_harm);
      by st_case veh_no;
      if a;
      rwd_mharm = m_harm;
run;
```

```
proc sort data=Newpre_USDEP10 NODUPKEY out=NewUSDEP_acc (keep=year st_case state newrwd_ind rwd_seq1
rwd_mharm);
      by year st_case;
run;

/*** Merging wth person file ***/
data NewUSDEP_per (keep=year st_case state veh_no per_no newrwd_ind rwd_seq1 rwd_mharm rwd_fatalinj
rwd_ainj rwd_fatalainj);
      merge NewUSDEP_acc
              fars&i..person;
      by st_case;
      if newrwd_ind = 1;

            if inj_sev = 4 then rwd_fatalinj = 1;  /*** fatal injury ***/
run;




/**** Files created below, for fatally injured, at person and vehicle and crash levels***/

proc sort data = NewUSDEP_per;
      by year st_case veh_no per_no;
run;

/*** person file for fatal injuries ***/

proc sort data = NewUSDEP_per out = NewUSDEP_fatalper (where=(rwd_fatalinj = 1));
      by year st_case veh_no per_no;
run;

/*** vehicle file for fatal injuries ***/

proc sort data = NewUSDEP_fatalper NODUPKEY out = NewUSDEP_fatalveh;
      by year st_case veh_no;
run;

/*** crash file for fatal injuries ***/
```

```
proc sort data = NewUSDEP_fatalveh NODUPKEY out = NewUSDEP_fatalacc (keep=year st_case state veh_no
per_no newrwd_ind rwd_seq1 rwd_mharm rwd_fatalinj);
      by year st_case;
run;

/****************************/
/*new RwD Definition, finish*/
/****************************/
/***********************************/
/*new INTERSECTION Definition, start */
/***********************************/
data Newint;
      set fars&i..accident (KEEP=year ST_CASE state RELJCT1 RELJCT2 harm_ev);
      if (reljct2 in(2,3,4,8)); /*** See FARS manual for attributes  of reljct2  ***/
      newint_ind = 1;
run;

proc sort data=Newint NODUPKEY out=Newint_acc (keep=year st_case state newint_ind);
      by year st_case;
run;

data Newint_per (keep=year st_case state veh_no per_no newint_ind int_fatalinj int_ainj
int_fatalainj);
      merge Newint_acc
              fars&i..person;
      by st_case;
      if newint_ind = 1;

            if inj_sev = 4 then int_fatalinj = 1;  /*** fatal injury ***/
run;

/**** Files created below, for fatally injured, at person and vehicle and crash levels***/


/*** person file for fatal injuries ***/

proc sort data = Newint_per out = Newint_fatalper (where=(int_fatalinj = 1));
```

```
      by year st_case veh_no per_no;
run;

/*** vehicle file for fatal injuries ***/

proc sort data = Newint_fatalper NODUPKEY out = Newint_fatalveh;
      by year st_case veh_no;
run;

/*** crash file for fatal injuries ***/

proc sort data = Newint_fatalveh NODUPKEY out = Newint_fatalacc (keep=year st_case state veh_no per_no
newint_ind int_fatalinj);
      by year st_case;
run;
/***************************/
/*new INT Definition, finish*/
/***************************/
/***************************/
/*new PED Definition, start */
/***************************/
data pedetal_per (keep=year st_case state per_typ per_no veh_no pedetal_ind ped_fatalinj ped_ainj
ped_fatalainj);
      merge fars&i..person (KEEP=ST_CASE PER_TYP PER_NO VEH_NO INJ_SEV)
              fars&i..accident (KEEP=YEAR ST_CASE STATE);

/*** merge of person and crash files for needed pedestrian and pedalcyclist information  ***/

      if (per_typ in(5,6,7,8)); /*** FARS categories for pedestrian and bicycle...see FARS manual for
category definitions ***/
      by st_case;
      pedetal_ind = 1;


             if inj_sev = 4 then ped_fatalinj = 1; /*** fatal injuries  ***/
run;

/**** Files created below, for fatally injured, at person and vehicle and crash levels***/
```

```
/*** person file for fatal injuries ***/

proc sort data = pedetal_per out = pedetal_fatalper (where=(ped_fatalinj = 1));
     by year st_case veh_no per_no;
run;

/*** vehicle file for fatal injuries ***/

proc sort data = pedetal_fatalper NODUPKEY out = pedetal_fatalveh;
     by year st_case veh_no;
run;

/*** crash file for fatal injuries ***/

proc sort data = pedetal_fatalveh NODUPKEY out = pedetal_fatalacc (keep=year st_case state per_typ
per_no veh_no pedetal_ind ped_fatalinj);
     by year st_case;
run;
/*****************************/
/*new PED Definition, finish*/
/*****************************/

%end;
%mend iter;
%iter;   /*** end of macro ***/
```