

```

/*****
*****
*
*       This code creates all of the CoIIN measures.
*       These measures are due in Co-Lab by the 8th of the month for the previous
months data.
*       The Co-Lab site is located at
https://mcehcoiin.community.nichq.org/user/login?destination=/feed
*       You must have an account set up by the AMCHP folks before you can access the
website. Or, the
*       measures can be forwarded to Rebecca Landers and she can enter the data into
Co-Lab.
*
*****
*****/

/*****
*
*       Get the data.
*
*****/
libname E pcfiles path='your data warehouse location goes here'; run;

/*      proc datasets library= E memtype=data; run;      */

data E
      (keep = SAMPLE_TYPE PARTY_ID DCN GENDER DOB AGE_AT_TEST_MONTHS
            CITY STATE ZIP_PLUS_4 JURISDICTION COUNTY COUNTY_FIPS_CODE
            COLLECT_DT TEST_TYPE SAMPLE_TYPE TEST_RESULT
            LAB_NAME PROV_NAME PROV_CITY PROV_COUNTY PAYMENT_SOURCE yt
cd mt yb
            Screen Specimen Confirmed Hierarchy);

      set E.ENVSURV_BLOOD_TESTS /*      Table split July 1 each year.  Until then it
hold current and previous years testing.      */
            E.ENVSURV_BLOOD_TESTS_2017;

      yt=year(datepart(COLLECT_DT));
      cd=datepart(COLLECT_DT);
      mt=month(datepart(COLLECT_DT));
      yb=year(datepart(DOB));

      /*      Screen is sorted and lagged later to check for screen testing. */
if SAMPLE_TYPE in ('CAPILLARY', 'UNKNOWN') then Screen = 1;
      else Screen = 2;

      /*      Specimen is used to find highest test, per kido, per unit of time.
*/
if SAMPLE_TYPE = 'VENOUS' then Specimen = 1;
      else if SAMPLE_TYPE = 'CAPILLARY' then Specimen = 2;

```

```

        else Specimen = 3;

        /*      Confirmed is used to flag venous tests, and non-venous EBL pairs.
*/
        if SAMPLE_TYPE = 'VENOUS' then Confirmed = 1;
            else Confirmed = 9; /* All non-venous are defaulted to 9, and get
re-evaluated later.      */
/*
        EPHT Hierarchy rules:
        1 - Venous >= 5
        2 - Two non-venous, both >= 5, within 84 days of each other.  Second tests
in the pair confirms the first.
        3 - Venous < 5
        4 - Cap >= 5, not confirmed
        5 - Cap < 5
*/
        if SAMPLE_TYPE = 'VENOUS' and TEST_RESULT >= 5 then Hierarchy = 1;
            else if SAMPLE_TYPE = 'VENOUS' and TEST_RESULT < 5 then Hierarchy =
3;
            else Hierarchy = 9; /* All non-venous are defaulted to 9 and get
re-evaluated later.      */

        where STATE = 'MO'
            and TEST_TYPE = 'LEAD - BLOOD'
            and SAMPLE_TYPE in ('VENOUS' 'CAPILLARY' 'UNKNOWN')
            and TEST_RESULT > -1
            /*      Bringing back extra months to confirm as many tests as
possible.      */
            and (-1 < AGE_AT_TEST_MONTHS <= 84)
            /*      Back years provide historical perspective.      */
            and datepart(COLLECT_DT) >= mdy(9,1,2017);

run;

libname E clear; run;

/*****
*
*      Lagging the file.
*
*****/
proc sort data = E out = Es; by PARTY_ID cd Screen TEST_RESULT; run;

data Lag1;
    set Es;
    by PARTY_ID;
    array x(*) cd_prev;
                                cd_prev = lag1(cd);
                                Screen_prev = lag1(Screen);

    if first.PARTY_ID then cnt = 1;

```

```

        do i = cnt to dim(x);
        x(i) = .;
            end;
    cnt + 1;

    format ScreenTest $3.;

    sdbt = cd - cd_prev; label sdbt = 'Screening Days';

    if 1 <= cd - cd_prev <= 365 then ScreenTest = 'no';
        else ScreenTest = 'yes';
run;

/* CDC Lead Branch step to get one test (the highest test), per kido, per day.
*/
proc sort data = Lag1 out = Lag1s; by PARTY_ID cd Specimen descending TEST_RESULT;
run;
proc sort data = Lag1s nodupkey out = Lag1sd dupout = Lag1sdout; by PARTY_ID cd;
run;

/* Get the TEST_REUSLTs on the same row to find if both are >= 5. */
data Lag2 (drop = cnt i);
    set Lag1sd (drop = cd_prev cnt i);

    tr = TEST_RESULT;

    by PARTY_ID;
    array x(*) cd_prev;
        cd_prev = lag1(cd);
        tr_prev = lag1(tr);

    if first.PARTY_ID then cnt = 1;
        do i = cnt to dim(x);
            x(i) = .;
                end;
    cnt + 1;

    /* Is this a Capillary pair? */
    pdbt = cd - cd_prev; label pdbt = 'Pair Days Between Tests';

    if Screen_prev = 1 and Screen = 1 and 0 < cd - cd_prev <= 84 then Pair =
'yes';
        else Pair = 'no';

    if Screen = 1 and tr_prev >= 5 and tr >= 5 and Pair = 'yes' then Confirmed =
1;
        else if Screen = 1 then Confirmed = 0;
            else Confirmed = Confirmed;

    if Specimen in (2, 3) and Confirmed = 1 then Hierarchy = 2;

```

```

        else if Specimen in (2 3) and tr >= 5 then Hierarchy = 4;
        else if Specimen in (2 3) and tr < 5 then Hierarchy = 5;
        else Hierarchy = Hierarchy;
run;

proc freq data = lag2 noprint; tables Specimen*Pair*Confirmed*Hierarchy / out =
Lag2chk; run;

/*      De-duplicate for highest result by Hierarchy, per kido, per year, per month.
*/
proc sort data = Lag2 out = Lag2s; by PARTY_ID yt mt Hierarchy descending
TEST_RESULT;
        where AGE_AT_TEST_MONTHS < 72; run;
proc sort data = Lag2s nodupkey out = Lag2sd dupout = Lag2sdout ; by PARTY_ID yt mt;
run;

/*****
*
*              CoIIN Calculations
*
*****/

/*      _N = numerator,  _D = Denominator,  _P = Percent,  _C = Count      */

/* 01, P1, and P2 */
proc sql;
create table O1_P2 as
select yt as Year, mt as Month,

        /*      01: Elevated Blood Lead Levels */
        sum(case when tr >= 5 and Confirmed = 1 then 1 else 0 end) as O1_N,
        count(*) as O1_D,
        sum(case when tr >= 5 and Confirmed = 1 then 1 else 0 end) / count(*)
        as O1_P format percent8.2,

        /*      P1: Blood Lead Level Screening Tests      */
        sum(case when ScreenTest = 'yes' then 1 else 0 end) as P1_C,

        /*      P2: Confirmatory Tests for Elevated Blood Lead Levels */
        sum(case when tr >= 5 and Confirmed = 1 then 1 else 0 end) as P2_N,
        sum(case when tr >= 5 then 1 else 0 end) as P2_D,
        sum(case when tr >= 5 and Confirmed = 1 then 1 else 0 end) /
        sum(case when tr >= 5 then 1 else 0 end) as P2_P format percent8.2

from Lag2sd
group by yt, mt
order by yt, mt;
quit;

/*      P3      */

```

```

proc sql;
  create table P3_N as
  select yt as Year, mt as Month, count(distinct PROV_NAME) as CDCCnt
  from Lag2sd
  where tr >= 5 and Confirmed = 1 and
  /*      CDC published recomenations      */
  /*      1 = Capillary, 2 = Venous, F = current test, FFlag1 =
previous test. */

  (
  (Screen_prev = 1 and ( 5 <= tr_prev < 10) and Screen = 2 and pdbt
<= 92) or
  (Screen_prev = 1 and (10 <= tr_prev < 45) and Screen = 2 and pdbt
<= 31) or
  (Screen_prev = 1 and (45 <= tr_prev < 59) and Screen = 2 and pdbt
<= 3) or
  (Screen_prev = 1 and (60 <= tr_prev < 70) and Screen = 2 and pdbt
<= 2) or
  (Screen_prev = 1 and Screen = 2 and tr_prev >= 70 and pdbt < 1) or

  /* Venous      */
  (Screen_prev = 2 and (5 <= tr_prev < 10) and Screen = 2 and tr >=
tr_prev and pdbt <= 92) or
  (Screen_prev = 2 and (5 <= tr_prev < 10) and Screen = 2 and tr <
tr_prev and pdbt <= 276) or

  (Screen_prev = 2 and (10 <= tr_prev < 20) and Screen = 2 and tr >=
tr_prev and pdbt <= 92) or
  (Screen_prev = 2 and (10 <= tr_prev < 20) and Screen = 2 and tr <
tr_prev and pdbt <= 184) or

  (Screen_prev = 2 and (20 <= tr_prev < 25) and Screen = 2 and pdbt <=
92) or

  (Screen_prev = 2 and (45 <= tr_prev < 70) and Screen = 2 and pdbt <=
21) or
  (Screen_prev = 2 and tr_prev >= 70 and Screen = 2 and pdbt <= 7))

  group by yt, mt
  order by yt, mt;
quit;

```

```

proc sql;
  create table P3_D as
  select yt as Year, mt as Month, count(distinct PROV_NAME) as ProvCnt
  from Lag2sd
  where tr >= 5 and Confirmed = 1
  group by yt, mt
  order by yt, mt;

```

```

quit;

proc sql;
    create table P3 as
    select  A.Year, A.Month, A.CDCCnt as P3_N,
           B.ProvCnt as P3_D, A.CDCCnt/B.ProvCnt as P3_P format
percent8.2
    from P3_N as A join P3_D as B
    on A.Year = B.Year and A.Month = B.Month;
quit;

/*****
*****
*
*     P4: Care in a Medical Home
*     At the end of each month Paula Darr will send an Excel worksheet via email
with the data required
*     for this measure.
*
*     Use DCN to match Paula's data to the ENVSURV confirmed EBL counts.
*
*****
*****/

data Tests (keep = DCN Year Month);
set Lag2sd (rename = (yt=Year mt=Month));
where  tr >= 5
      and Confirmed = 1;
run;

/* 'Care' refers to the < 72 months file Paula sent. */
proc sort data = Care (drop = Age); by DCN Year Month; run;
proc sort data = Tests; by DCN Year Month; run;

data Ct Only Tonly;
merge  Care  (in = C)
       Tests (in = T);

by DCN Year Month;

fromC = C;
fromT = T;

if fromC = 1 and fromT = 0 then output COnly;
   else if fromC = 0 and fromT = 1 then output TOnly;
   else output Ct;

run;

proc sql;

```

```
create table P4 as
select Year, Month, count(*) as P4
from Ct
group by Year, Month
order by Year, Month;
quit;
```

```
/*      P5: Lead Home Abatement and Hazard Repair
This number can only be provided by the Lead Licensing program.
Tony started providing numbers in March, 2019. */
```

```
proc sql;
create table P5 (
    Year num format 4.,
    Month num format 2.,
    P5_C num format 3.);
```

```
insert into P5
    set Year = 2019,
        Month= 2,
        P5_C = 0
    set Year = 2019,
        Month = 3,
        P5_C = 7;
```

```
quit;
```

```
/*      P6: Provider Education (Per Sharon Odom the CLPPP program has not started
doing this yet.)      */
```

```
proc sql;
create table P6 (
    Year num format 4.,
    Month num format 2.,
    P6_C char(15) /*smallint*/
);
```

```
insert into P6
    set Year = 2019,
        Month = 2,
        P6_C = 'Not Started'
    set Year = 2019,
        Month = 3,
        P6_C = 'Not Started';
```

```
quit;
```

```
/*      OP1: Case Management
From 4/9/2019 internal CoIIN meeting, we decided that denominator would =
```

```
state confirmed EBLs,  
    and the numerator would = state confirmed EBLs - ( Grey Aera (less JASPER) 5  
<= confirmed EBLs < 10. */
```

```
proc sql;  
create table OP1 as  
select A.Year, A.Month, B.OP1_D - A.OP1_Factor as OP1_N, B.OP1_D, (B.OP1_D -  
A.OP1_Factor) / B.OP1_D as OP1_P format percent8.2  
from  
  
(      select  yt as Year,  
                mt as Month,  
                sum(case when 5 <= TEST_RESULT < 10 and Confirmed = 1 then 1  
else 0 end) as OP1_Factor  
        from Lag2sd  
        where JURISDICTION in ('ST LOUIS', 'ST LOUIS CITY', 'JEFFERSON', 'KANSAS  
CITY', 'GREENE')  
        group by Year, Month      ) as A  
  
        left join  
  
(      select  yt as Year,  
                mt as Month,  
                sum(case when TEST_RESULT >= 5 and Confirmed = 1 then 1 else  
0 end) as OP1_D  
        from Lag2sd  
        group by Year, Month      ) as B  
  
        on A.Year=B.Year and A.Month=B.Month;  
quit;
```

```
/*  
*  
*      Bring the individual calculations into one CoIIN table  
*  
*/
```

```
data CoIIN;  
merge O1_P2 P3 P4 P5 P6 OP1;  
by Year Month;  
run;
```

```
filename PutHere "location where you want to save the table +  
\CoIIN_&SYSDATE..xlsx";
```

```
proc export data = CoIIN  
    dbms = xlsx  
    outfile = PutHere replace;  
    sheet = 'Summary Table';  
run;
```