```
%************************************************************* ;
%************ Reshape Eye Exam Dataset ************** ;
%************************************************************* ;
/************************************************
Purpose: transport patient-level dataset into eye-level dataset
record in import dataset should be in patient-level with eye-level variables (e.g. sisurgery_rul/sisurgery_lul) except variable subjid,
each patient-level record will be transport into 2 eye-level records for OD/OS, variable=eye is added to indicate OS/OD in new dataset
eye-level variables (for OS/OD) within patient-level dataset into patient-level variable
(eg: turn variable sisurgery_rul/sisurgery_lul in original dataset into sisurgery_ul in output datasets)
************************************************/
/*
parameter
1. indata: input dataset, eg, WORK.PRIMARY_END_W4
2. memname: same as input dataset, eg, "PRIMARY_END_W4"
3. outdata: output dataset, eg, PRIMARY_END_W4_*/

* Rename Macro - rename old variable name as new variable name;
%macro proc_rename1(oldvarlist, newvarlist);
 %let k=1;
 %let old = %scan(&oldvarlist, &k);
 %let new = %scan(&newvarlist, &k);
   %do %while(("&old" NE "") & ("&new" NE ""));
    rename &old = &new;
          %let k = %eval(&k + 1);
    %let old = %scan(&oldvarlist, &k);
```

```
    %let new = %scan(&newvarlist, &k);
 %end;
%mend;

%macro proc_reshape(indata,memname,outdata);
* extract variable names;
proc sql noprint;
        create table temp_varnames as
        select name
        from dictionary.columns
         where libname='WORK' and memname= &memname;
quit;


*extract left/right eye related variable;
data temp_right_var temp_left_var ;
        set temp_varnames;
        if substr(name,length(name)-1,2) ="_r" then do;
        right = 1;
        main = substr(name,1,length(name)-2);
        end;
        else if substr(name,length(name)-1,2) ="_l" then do;
        right = 0;
        main = substr(name,1,length(name)-2);
        end;
        else if substr(name,length(name)-3,4) = "_rul" then do;
        right = 1;
```

```
main = substr(name,1,length(name)-3)||'_'||substr(name,length(name)-1,2);
end;
else if substr(name,length(name)-3,4) = "_rll" then do;
right = 1;
main = substr(name,1,length(name)-4)||'_'||substr(name,length(name)-1,2);
end;
else if substr(name,length(name)-3,4) = "_lul" then do;
right = 0;
main = substr(name,1,length(name)-4)||'_'||substr(name,length(name)-1,2);
end;
else if substr(name,length(name)-3,4) = "_lll" then do;
right = 0;
main = substr(name,1,length(name)-4)||'_'||substr(name,length(name)-1,2);
end;
else if substr(name,length(name)-4,2) = "r_" then do;
right = 1;
main = substr(name,1,length(name)-5)||substr(name,length(name),2);
end;
else if substr(name,length(name)-4,2) = "l_" then do;
right = 0 ;
main = substr(name,1,length(name)-5)||substr(name,length(name),2);
end;
if right = 1 then output  temp_right_var;
else if right = 0 then output temp_left_var;


proc sql noprint;
```

```
        select name into : OD_names separated by ', '
        from temp_right_var;

        select name into : OS_names separated by ', '
        from temp_left_var;

        select name into : OD_names2 separated by ' '
        from temp_right_var;

        select name into : OS_names2 separated by ' '
        from temp_left_var;

        select main into : new_names separated by ' '
        from temp_left_var;
quit;

 %put &OD_names;
 %put &OS_names;
 %put &OD_names2;
 %put &OS_names2;
 %put &new_names;


* extract OS/OD variables into OS/OD dataset;
proc sql;
        create table temp_OD as
        select subjid,redcap_event_name,
```

```
        &OD_names,"OD" as eye
        from  &indata ;

        create table temp_OS as
        select subjid,redcap_event_name,
        &OS_names,"OS" as eye
        from  &indata ;
quit;

 * Rename OS/OD variables in OS/OD datasets into patient-level variables;
 data temp_OD2;
  set temp_OD;
  %proc_rename1(&OD_names2,&new_names);
run;

data temp_OS2;
  set temp_OS;
  %proc_rename1(&OS_names2,&new_names);
run;
* combine the OS/OD datasets;
data &outdata;
        set temp_OD2 temp_OS2;
run;

proc datasets nolist;
        delete temp: ;
run;
```

```
%mend proc_reshape;
```