```
/*****************************************************************/
/* Program Name: variable creation with arrays                   */
/* Program Location: C:\Users\dsingh\Dropbox\Stat_604\Homework\HW6 */
/* Date Created: 6/13/2019                                       */
/* Author: Dhruv Singh                                           */
/* Purpose: loops and arrays
*/

libname hwdata 'C:\Users\dsingh\Dropbox\Tamu\Stat_604\Homework\hwdata'
access = readonly;
libname charity
'C:\Users\dsingh\Dropbox\Tamu\Stat_604\Homework\HW6_DueJune18';

filename report
'C:\Users\dsingh\Dropbox\Tamu\Stat_604\Homework\HW6_DueJune18\HW6DSingh_H
W06_PCoutput.pdf';


/* Step 2: reading in chopnjoe data */
data chopnjoe;
      set hwdata.chopnjoe19;
run;

* step 2 contd: rotating to narrow form using do loop;
data rotate2 (keep=Employee_Id i Organization);
   set hwdata.chopnjoe19
               (drop=name department salary amount1-amount10);
   array contrib{10} $ charity1-charity10;
   do i=1 to 10;
      if contrib{i} ne missing then do;
         Organization=contrib{i};
         output;
      end;
   end;
run;

/* Step 3: sorting narrow dataset in place */
proc sort data = rotate2;
      by Organization;
run;


/* Step 4: creating sorted charities data in work library */
data charities;
      set hwdata.charities;
run;

proc sort data = charities;
      by Organization;
run;


/* Step 5: combining datasets by organization */
data combined_data (drop = Org_id);
      merge rotate2 (in=r)
            charities (in=c);
      by organization;
      if r=1 and c=1;
run;
```

```
/* Step 6: transposing marged data from narrow to flat */
proc sort data = combined_data;
      by employee_id;
run;

proc transpose data = combined_data
                      out = charity_data (drop = _NAME_ _LABEL_)
                      prefix = Donee_Type;
      var category;
      by employee_id;
      id i;
run;

* reordering columns ;
data charity_data;
      retain employee_id donee_type1-donee_type10;
      set charity_data;
run;

/* Step 7: final merge, summary variables */
proc sort data = charity_data;
      by employee_id;
run;

* sort employee id for merge ;
proc sort data = chopnjoe;
      by employee_id;
run;

* merging;
data giving_analysis (drop = relief_amt1-relief_amt10 hunger_amt1-
hunger_amt10 i);
      merge chopnjoe (in=chop)
            charity_data (in=char); /* org types wide*/
      by employee_id;
      if chop=1 and char=1;

      * creating array for amount contributions;
      array contrib{*} amount1-amount10;

      * creating a char array for org type;
      array org_type{10} $ donee_type1-donee_type10;

      * creating empty arrays to record relief hunger amounts;
      array relief_amt{10};
      array hunger_amt{10};

      * populating amt arrays;
      do i=1 to 10;
            if org_type{i}='Relief' then relief_amt{i}=contrib{i};
            else relief_amt{i}=0;

            if org_type{i}='Hunger' then hunger_amt{i}=contrib(i);
            else hunger_amt{i}=0;
      end;

      * computing decomposed sums;
      chrty1_amt= sum(of relief_amt{*});
      chrty2_amt = sum(of hunger_amt{*});
```

```
      * creates total contribution variable;
      total = sum(of contrib{*});

      * percent column;
      gift_pct = total/salary;

      label chrty1_amt = "Relief Amount"
            chrty2_amt = "Hunger Amount"
            total = "Total Contributions"
            gift_pct = "% of Salary Given";

      format gift_pct percent6.1;

run;

/* Step 8: printing descriptor and data portions of final dataset */
ods pdf file = report;
title 'Step 8: Descriptor Portion of Giving Analysis Data Set';
proc contents data = giving_analysis;
run;

title 'Step 8: Data Portion of Giving Analysis Data Set';
proc print data = giving_analysis noobs label;
      var employee_id name department salary chrty1_amt chrty2_amt total
gift_pct;
run;
ods pdf close;
```