Analyzing Multiple Response Data Using the TABULATE Procedure
Tina Keene, SAS Institute Inc., Cary, NC

ABSTRACT
Analyzing multiple response data with the SAS® System is a difficult task. This paper provides one method of producing a multiple response report using the TABULATE procedure. The problems involved in generating such a table are presented. Also, the data manipulations necessary for preparing the data for the TABULATE procedure are discussed. Next, the PROC TABULATE statements needed to generate the report and the logic behind the percentage definitions are addressed. Special considerations for handling "NO RESPONSE" observations are also discussed.

INTRODUCTION
PROC TABULATE is a powerful tool within the SAS System. With just a handful of statements, you can produce detailed reports displaying statistics and summaries for many different categories of data. One of the more complex uses of PROC TABULATE involves the analysis of multiple response data. This particular application can be developed, although it is not simple to derive. This paper discusses a method of approaching the problem of multiple response data analysis.

DEFINING THE PROBLEM
The basic purpose of any analysis is to make comparisons in order to draw a conclusion. With single response data, the process is fairly straightforward. There is one response per question so you can input data as one variable whose value is the answer selected. You can then use this one variable as a classification variable in PROC TABULATE. The primary comparison involves comparing each frequency count with the overall frequency.

However, when there is the potential for multiple responses to a single question, the situation is more complicated. Because each respondent may select one or more answers, initial data entry is easiest if one variable is input for every possible response. The value of each variable is 1 if that answer is chosen and missing if it is not. For example, suppose you are asked which of the following activities you have participated in during the past month: swimming, fishing, and skiing.

To handle responses for this question, you create three separate variables: SWIM, FISH, and SKI. If you select swimming, then you assign SWIM the value 1. If you do not select swimming, you give SWIM a missing value. The same procedure applies to the other two variables. There are two difficulties that arise as a result of this method of entry:

- grouping responses to represent separate answers to a single question
- accessing the total number of respondents in PROC TABULATE.

Keeping in mind that placing a variable in the CLASS statement in PROC TABULATE causes all of its values to be displayed as separate categories under a single heading, the most logical approach to the first problem is to reshape the data, creating a new variable to use as a CLASS variable in PROC TABULATE.

Using the example described above, you would create one variable, which you could call ACTIVITY. If SWIM = 1, then assign ACTIVITY = SWIMMING; if FISH = 1, then assign ACTIVITY = FISHING; and if SKI = 1, then assign ACTIVITY = SKIING. If any of the three variables is missing, you do nothing. However, in creating this variable, one observation for each response instead of for each respondent, is output. After doing this reordering, the number of observations is no longer equal to the number of respondents, and the second problem emerges.

Another aspect of the multiple response dilemma is that you have a choice of comparisons; you can compare individual response frequencies to the number of people answering the survey (respondents) or to the total number of responses to the question. These two values can be different because one person may make more than one response to that question.

By manipulating the data correctly and by using certain features of PROC TABULATE, you can overcome all of these difficulties and obtain percent statistics to reflect both comparisons mentioned above.

SOLVING THE PROBLEM
The results of the fictitious survey in Figure 1 will serve as an example for producing a table from multiple response data. The purpose of this survey is to determine what effects, if any, age has on an employee's satisfaction with different aspects of his or her job.

1. What is your current age?

   20-30 __________  41-65 __________
   31-40 ________  over 55 ________

2. Which are the most satisfying aspects of your job? (Check all that apply)

   Salary ______ Retirement Benefits ______
   Medical Benefits ______ Working Environment ______
   Other (please specify) ______

Figure 1 Example Survey

Input the Data

The first step, if it has not already been done for you, is to enter the raw data and read it into a SAS data set. As mentioned before, the usual practice is to enter a separate variable for each possible response and code the values as 1 or missing () to indicate selection or lack of selection. A partial listing of the raw data file for the example survey is shown in Figure 2. The first column contains the respondent's age category number, and each of the remaining columns represents the person's response (or lack of response) to each choice presented in question 2 of the survey.
10 sketch the layout of the data. You can then easily work backwards to derive the appropriate TABLE statement. Figure 3 contains an outline of the table for the survey example.

**JOB SATISFACTION vs. AGE**

<table>
<thead>
<tr>
<th>All AGE</th>
<th>AGE Groups (list each age group)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Respondents count</td>
<td></td>
</tr>
<tr>
<td>Aspects:</td>
<td></td>
</tr>
<tr>
<td>each count</td>
<td></td>
</tr>
<tr>
<td>job % responses</td>
<td></td>
</tr>
<tr>
<td>aspect % respondents</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 2** Partial Listing of Raw Data File

**Sketch the Table**

The second step in any PROC TABULATE application should be to sketch the layout of the desired table. You can then easily work backwards to derive the appropriate TABLE statement. Figure 3 contains an outline of the table for the survey example.

**Figure 3** Table Layout

**Reshape the Data**

One of the most difficult steps in producing a table for a multiple response application is reshaping the data. The first problem is how to group the responses for the multiple response variable so they represent different alternatives to the same question. The best solution is to create one variable and output a separate observation for each response. The following code illustrates how to do this for the first two aspects of the job satisfaction survey:

```
DATA MULTRESP;
  SET MULTRESP;
  LENGTH ASPECT $10;
  IF ASPECT='1' THEN DO;
    ASPECT='SALARY';
    OUTPUT;
  END;
  IF ASPECT='2' THEN DO;
    ASPECT='MEDICAl BENEFITS';
    OUTPUT;
  END;
```

The `LENGTH` statement creates a character variable `ASPECT` with a length of 10 bytes. Then, because variable `ASPECT1` represents the selection `SALARY`, check to see if it has a value of 1 (that is, it has been selected). If so, assign the value `SALARY` to `ASPECT` and output the observation. If `SALARY` has not been chosen, do nothing. The same logic is used for each of the four job aspects on the survey. The only difference occurs when coding the OTHER selection:

```
IF OTHERRESP NE ' 10' THEN DO;
  ASPECT='OTHER';
  OUTPUT;
END;
```

In this case, if an OTHER selection has been made, the respondent should have entered a value in his survey. You simply need to assign that value to the variable `ASPECT`.

Now another problem arises. Since the data has been reshaped to create the variable `ASPECT`, the potential exists for more than one observation per respondent. Therefore, you need a way to access the number of respondents in the PROC TABULATE step. One important feature of PROC TABULATE provides a relatively simple way to do this. In the TABULATE procedure, if you cross an analysis variable (a variable in the VAR statement) with the statistic N, the result is the frequency of observations for which the analysis variable has a non-missing value. In order to make use of this feature, you need to create a numeric variable whose value is non-missing for only one observation per respondent. The following statements code the variable `ID` so it has a missing value for all but the first observation for each respondent:

```
DATA MULTRESP;
  SET MULTRESP;
  ID=1;
  IF ASPECT='1' THEN DO;
    ASPECT='SALARY';
    OUTPUT;
    ID=;
  END;
  IF ASPECT='2' THEN DO;
    ASPECT='MEDICAL BENEFITS';
    OUTPUT;
    ID=;
  END;
```

By putting the statement

```
ID=;
```

after the `OUTPUT` statement in each DO loop, the current value of `ID` is output before it is reassigned. Thus, `ID` is equal to 1 in the observation containing the first response of each respondent. Then `ID` is recoded to a missing value. That missing value is carried through until another observation is read in from data set `MULTRESP` and `ID` is reassigned to 1. Then the process begins again. Now that the `ID` variable has been coded, you can derive the total number of respondents in PROC TABULATE by putting the variable `ID` in the `VAR` statement and requesting the frequency for `ID` in the `TABLE` statement. See `OUTPUT 1` at the end of this paper for a partial `PRINT` procedure listing of the final reshaped data set.

**Set up PROC TABULATE Statements**

The next step in producing the table is to set up the PROC TABULATE statements. Go back to the outline in Figure 3. The first thing you must do is identify the type of each variable for PROC TABULATE. Because you want statistics to be computed for the individual values of both `ASPECT` and `AGE`, you should place these in the `CLASS` statement. Also, as mentioned before, `ID` must be in the `VAR` statement.

Next, assemble the `TABLE` statement. The desired table is only two-dimensional, so begin by defining the row dimension expression. To produce the first row of the table (Number of Respondents), remember that you need to use only `ID`'N. The job aspects form the remainder of the rows; indicate this by concatenating the variable `ASPECT` with `ID`'N. To print the total for all job aspects, concatenate the universal class variable `ALL` with `ASPECT`. Finally, tell PROC TABULATE to compute the frequency and percentages for each level of `ASPECT` and for all

```
PROC TABULATE DATA=__;
CLASS ID I(N) ASPECT;
TABLE ASPECT*ID=1;
PRINT procad procedure listing of the final
res,ponsa of Each respondent.
```

Next, assemble the `TABLE` statement. The desired table is only two-dimensional, so begin by defining the row dimension expression. To produce the first row of the table (Number of Respondents), remember that you need to use only `ID`'N. The job aspects form the remainder of the rows; indicate this by concatenating the variable `ASPECT` with `ID`'N. To print the total for all job aspects, concatenate the universal class variable `ALL` with `ASPECT`. Finally, tell PROC TABULATE to compute the frequency and percentages for each level of `ASPECT` and for all

```
PROC TABULATE DATA=__;
CLASS ID I(N) ASPECT;
TABLE ASPECT*ID=1;
PRINT procad procedure listing of the final
res,ponsa of Each respondent.
```
ASPECTs by crossing the N and PCTN statistics with ASPECT and ALL.

The column dimension consists of a column for all age groups together and a column for each age group. As with the row dimension, the total column is defined by ALL and the age group columns result from concatenating the AGE variable with it. This is the resulting TABLE statement:

```
TABLE IDVAR (ASPECT ALL)="N"
   PCTN /for percentage of/ 
   PCTN /TOTAL RESPONSES /
   ALL /ALL AGE/;
```

Derive Denominator Definitions

Along with recoding data, defining percentage denominators to the TABULATE procedure is one of the most difficult aspects of this type of application. The purpose of a denominator definition is to tell the TABULATE procedure to collect totals for the denominator of a percentage calculation. PROC TABULATE collapses, or sums, over frequencies or sums associated with individual CLASS variable crossing categories. The categories that are included in this collapsing process are determined by the crossings specified in the denominator definition. A good way to approach defining any denominator in PROC TABULATE is to list all possible crossings that include the percent statistic. Use the TABLE statement from above with one of the PCTN statistics:

```
TABLE IDVAR (ASPECT ALL)="N"
   PCTN /for percentage of/ 
   PCTN /TOTAL RESPONSES /
   ALL /ALL AGE/;
```

and expand all crossings. Since the comma implies crossing between dimensions, the list of all possible crossings that include the PCTN reference is as follows:

```
ASPECT*ALL*ALL
ASPECT*ACCEPT*ALL
ALL*ASPECT*ALL
ALL*ACCEPT*ALL
```

Then select the class variable or variables from each crossing for which the total frequency for all levels produces the desired denominator. In other words, if you want the percentage based on column totals, include all class variable crossings from the row dimension and vice versa for row totals. In the case of the example survey, the desired denominators are the total number of respondents and responses for the entire data set. Thus, include all class variable crossings from both the row and column dimensions. Remember that you may not group variables using parentheses in a denominator definition; you must expand all crossings.

But, how do you differentiate between the total number of respondents and the total number of responses? The answer involves making use of the ID variable once again. The crossing ID'N was used in the TABLE statement to produce the number of respondents. This same technique can be used in the denominator definition to compute a percentage of the number of respondents. By crossing ID with the other variables in the denominator, you instruct PROC TABULATE to use the frequency of observations for which ID has a nonmissing value. It is necessary to include the class variables in the crossing to tell the TABULATE procedure to use the frequency of nonmissing ID values for all class variable levels. If you were to use ID alone as the denominator definition, PROC TABULATE would use the number of nonmissing ID values for each class variable value individually, rather than the total for the entire data set. Consequently, the percent specification for the percentage of total respondents looks like this:

```
PCTN=ID'N*ASPECT*ALL ID'N*ACCEPT*ALL
ID'N*ASPECT*ALL
ID'N*ACCEPT*ALL
```

To achieve the percentage of total responses, simply remove the ID variable from the specification above:

```
PCTN=ACCEPT*ALL ID'N*ACCEPT*ALL ALL
```

Or, in this case, PCTN with no denominator definition will use by default the total number of responses as the denominator.

After adding the percentages, the PROC TABULATE code looks like this:

```
PROC TABULATE DATA=EXEMPT2;
   CLASS ASPECT AGE;
   VAR ID;
   TABLE IDVAR (ASPECT ALL)="N"
      PCTN /ID'N*ASPECT*ALL ID'N*ACCEPT*ALL
         ID'N*ASPECT*ALL
         ID'N*ACCEPT*ALL
         ALL
         ALL AGE/;
```

Output 2 at the end of this paper contains the resulting table.

Enhance the Output

Now that the contents of the table are complete, all that remains is to enhance its appearance. This can be done by making use of formats, labels, and certain options specific to the TABULATE procedure. For the example survey table, you can use the FORMAT procedure to format the values of AGE into meaningful ranges. Also, you can create a format for putting percent signs in the percentage cells of the table by using the PICTURE statement in PROC FORMAT. The final code, after adding enhancements, might look like this:

```
PROC TABULATE DATA=EXEMPT2 FORMAT=PCTFMT 2;
   CLASS ASPECT AGE;
   VAR ID;
   TABLE IDVAR (ASPECT ALL)="N"
      PCTN /ID'N*ASPECT*ALL ID'N*ACCEPT*ALL
         ID'N*ASPECT*ALL
         ID'N*ACCEPT*ALL
         ALL
         ALL AGE/;
   FORMAT AGE AGENT;
   LABEL ASPECT="ASPECT"
   ASPECT="AGE"
   KEYLABEL ALL="TOTAL RESPONSES";
```

(For an explanation of the RTS-, ROW-, and BOX= TABLE statement options and the KEYLABEL statement, please refer to the SAS Guide to TABULATE Processing, 1.97 Edition.)

Output 3 at the end of this paper contains the enhanced table.

(Note: You may notice slight differences in values formatted with the PCTFMT format. These differences occur because the default format used by PROC TABULATE rounds the values; any format created with the PICTURE statement in PROC FORMAT truncates values. If precision is important, you may want to carry an extra decimal place or omit the percent format.)

HANDLING 'NO RESPONSE' DATA

One special case in this tabulation of multiple response data must be considered. This case involves the possibility of 'NO RESPONSE' observations; that is, situations where no alternatives are chosen. Until now this possibility has been ignored. In the example survey, a respondent who did not select any job...
aspect was not counted. It is valid to assume that such a respondent is not satisfied with any aspect of his job, and his survey should be included in the analysis.

It would seem to be a simple task to make 'NO RESPONSE' a valid value for the selection variable (ASPECT, in the case of the survey example) and continue on as discussed in the previous section. However, when you introduce 'NO RESPONSE' as a value for ASPECT, the AGE group frequencies for all levels of ASPECT increase by the number of 'NO RESPONSE' surveys in each AGE group. Since these total frequencies are used as the denominators for the percentages in the ALL row, those denominators will include the 'NO RESPONSE' frequencies. Also, the denominator for the percentage of total responses will include the 'NO RESPONSE' frequency along with the frequency counts for the other values of ASPECT. There is no method of selecting only certain class variable values for inclusion in the totals and percentage values.

As a result, it becomes necessary to derive new frequencies in the ALL row (and, hence, new denominators for the percentages in that row) and a new denominator for the percentage of total responses. This is not a difficult task since the same approach used for obtaining the number of respondents can be adopted here. Just create a new variable in the data set similar to ID, that is equal to 1 if any alternative is selected and equal to missing if no selections are made. The following DATA step creates this variable, called RESP, and also introduces the value 'NO RESPONSE' to the variable ASPECT.

The asterisks are included with the value 'NO RESPONSE' for two reasons. First of all, they help to set the value 'NO RESPONSE' apart from the other values of ASPECT. Secondly, an asterisk occurs before any alphabetic character in the sorting sequence. Placing an asterisk at the beginning of 'NO RESPONSE' causes this value to be placed before the other values of ASPECT in the table.

DATA MORES;
SET MORERESP;
LENGTH ASPECT $40;
RESP=1;
ID=1;
IF ASPECT=1 THEN DO;
   ASPECT='SALARY';
   OUTPUT;
   ID=1;
END;
  
   (repeat for all values of ASPECT)
   
  IF OTHERASP= ' ' THEN DO;
   ASPECT='OTHERASP';
   OUTPUT;
   ID=1;
END;

IF ASPECT=1 AND ASPECT=2 AND ASPECT=3.
   AND ASPECT=4. AND OTHERASP= ' ' THEN DO;
   ASPECT= '******** NO RESPONSE **********';
   RESP=1;
   OUTPUT;
   ID=1;
END;

KEEP 3D AGE ASPECT USE;

Then define the denominator the same as for the percentage of total respondents but substitute RESP for ID:

PROC TABULATE;
CLASS ID AGE ASPECT;
VAR PCTNS; PCTNS1; PCTNS2; PCTNS3; PCTNS4;
TITLE 'N RESP ASPECT ID AGE ASPECT';
RUN;

To obtain the total number of responses (excluding the 'NO RESPONSE' surveys) over all levels of ASPECT, cross RESP

with N and ALL in the row dimension expression. In order to exclude the 'NO RESPONSE' frequency from the denominators for the total percentages, you also need to cross RESP with the PCTNS statistics. However, if you include RESP in these crossings, you can no longer group ASPECT and ALL with parentheses; the crossings now will be uneven, so you must express them separately. Consequently, when you break up ASPECT and ALL, you must also break up the denominator definitions. In other words, make sure to include ASPECT only in the denominators for the PCTNS crossed with ASPECT and to include ALL only with the denominators for the PCTNS crossed with ALL. The new TABLE statement now looks like this:

PROC TABULATE;
CLASS ID ASPECT ID PCTNS ID ALL;
VAR PCTNS1; PCTNS2; PCTNS3; PCTNS4;
TITLE 'N RESP ASPECT ID ALL';
RUN;

The final table, including the 'NO RESPONSE' value, the corrected percentage of total responses, and the enhancements, appears in Output 4 at the end of this paper.

CONCLUSION

As the preceding discussion and example illustrate, the task of analyzing multiple response data using the TABULATE procedure is not a simple one. You must be concerned with how to input and rearrange the data correctly and efficiently and how to set up the TABLE statement to produce the desired table. Also, you are faced with the problem of defining the appropriate denominator definitions and handling observations for which no response was made. However, approaching the problem systematically and incorporating features of the DATA step in conjunction with PROCTABULATE make this task more manageable and less formidable. The result is a well-organized and informative report.

SAS is a registered trademark of SAS Institute Inc.
### Output 2: Table Before Enhancements

<table>
<thead>
<tr>
<th>ASPECT</th>
<th>I</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>ED IN</td>
<td>ALL</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SATISFY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PTC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MONATURE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESPONSE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>INCOME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SELF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENVIRONMENT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COUNT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Output 3: Table After Enhancements

<table>
<thead>
<tr>
<th>ASPECT</th>
<th>I</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>ED IN</td>
<td>ALL</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SATISFY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PTC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MONATURE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESPONSE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>INCOME</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SELF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ENVIRONMENT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COUNT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**167**
### Table

<table>
<thead>
<tr>
<th>Job Satisfaction</th>
<th>Age of Respondent</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>All Age</td>
<td>20-29</td>
<td>30-40</td>
<td>41-55</td>
</tr>
<tr>
<td><strong>Number of Respondents</strong></td>
<td>10</td>
<td>12</td>
<td>15</td>
<td>8</td>
</tr>
<tr>
<td><strong>Job Aspect</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Count</td>
<td>61</td>
<td>41</td>
<td>23</td>
</tr>
<tr>
<td><strong>No Response</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Count</td>
<td>13</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td><strong>Medical Benefits</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>% of Total</td>
<td>14.26%</td>
<td>11.18%</td>
<td>8.38%</td>
</tr>
<tr>
<td></td>
<td>Responses</td>
<td>20.3%</td>
<td>31.2%</td>
<td>31.2%</td>
</tr>
<tr>
<td><strong>Salary</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>% of Total</td>
<td>23.1%</td>
<td>30.5%</td>
<td>31.2%</td>
</tr>
<tr>
<td></td>
<td>Responses</td>
<td>23.1%</td>
<td>30.5%</td>
<td>31.2%</td>
</tr>
<tr>
<td><strong>Status</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>% of Total</td>
<td>16.2%</td>
<td>13.2%</td>
<td>16.2%</td>
</tr>
<tr>
<td></td>
<td>Responses</td>
<td>16.2%</td>
<td>13.2%</td>
<td>16.2%</td>
</tr>
<tr>
<td><strong>Working Environment</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>% of Total</td>
<td>15.0%</td>
<td>23.2%</td>
<td>11.8%</td>
</tr>
<tr>
<td></td>
<td>Responses</td>
<td>15.0%</td>
<td>23.2%</td>
<td>11.8%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Count</td>
<td>100</td>
<td>80</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>Responses</td>
<td>100.0%</td>
<td>80.0%</td>
<td>50.0%</td>
</tr>
</tbody>
</table>

Output 4 Table with 'NO RESPONSE' and Corrected Percentage