MWSUG 2016 - Paper HW03

The Joinless Join ~ The Impossible Dream Come True; Expand the Power of Base SAS[®] and SAS[®] Enterprise Guide[®] in a New Way

Kent ♥ Ronda Team Phelps, The SASketeers, Des Moines, IA All for SAS and SAS for All!



Base SAS and SAS Enterprise Guide can easily combine data from tables or data sets by using a PROC SQL Join to match on like columns or by using a DATA Step Merge to match on the same variable name. However, what do you do when tables or data sets do not contain like columns or the same variable name and a Join or Merge cannot be used? We invite you to attend our exciting presentation on the Joinless Join where we will teach you how to expand the power of Base SAS and SAS Enterprise Guide in a new way.

We will empower you to creatively overcome the limits of a standard Join or Merge. You will learn how to design a Joinless Join based upon dependencies, indirect relationships, or no relationships at all between the tables or data sets. In addition, we will highlight how to use a Joinless Join to prepare unrelated joinless data to be utilized by ODS and PROC REPORT in creating a PDF. Come experience the power and the versatility of the Joinless Join to greatly expand your data transformation and analysis toolkit.

We look forward to introducing you



The tagline for SAS is *The Power To Know*[®] and your 'power to know' greatly expands with your ability to access, combine, and analyze important data from tables or data sets (referred to as tables going forward). **The Power To Know** sets off **The Power To Create** which leads to **The Power To Automate** ~ much like an intricate and fluid domino design. However, this power will quickly become disjointed if you do not know how to effectively Join or Merge tables of data ~ **even when the tables do not have a relationship**.

Here are 2 questions to ask yourself when analyzing 2 or more tables:

- Do the tables contain like columns or the same variable name which can be utilized in a Join or Merge?
- If the tables do not contain like columns or the same variable name and a standard Join or Merge cannot be used, have I reached a *cavernous and insurmountable 'woe is me' research impasse* in my data analysis?

🙂 There is no need to fear, the Joinless Join is here! 🙂

The Joinless Join will bridge your research impasse and empower you to:

- Creatively overcome the limits of a standard Join or Merge using Base SAS and SAS Enterprise Guide
- Access, combine, and analyze tables for the first time based upon dependencies, indirect relationships, or no relationships at all
- Open up new worlds of table creations, calculations, validations, and filtrations
- Prepare unrelated joinless data to be utilized by ODS and PROC REPORT
- Increase your ability to detect and resolve errors including hidden errors
- Prevent validation process failure ~ yea! ~ and completely... yes, completely automate your projects

The SAS project in this presentation demonstrates:

The Power To Know how to design a Joinless Join

The Power To Create tables based upon dependencies, indirect relationships, or no relationships at all

The Power To Automate projects even when tables cannot be directly joined or merged

We invite you to journey with us as we help you Е Х Ρ Α Ν D

the power of Base SAS and Enterprise Guide in a new way.



A standard Join or Merge enables you to combine tables side-by-side horizontally by matching related rows. A like column or the same variable name, with the same attributes and like values, is used to connect the tables and bring together some or all of each table's contents.

An Inner Join or Merge is a symmetrical process of matching related rows in tables ~ an Inner Join can match related rows in **2 to 256** tables, and a Merge can match related rows in **2** tables.



The result of an **Inner Join or Merge** produces only matched rows from the tables. The result is illustrated by the shaded area AB in Figure 1.

Figure 1. Venn Diagram - Inner Join or Merge

An **Outer Join or Merge** is an asymmetrical process of matching related rows in **2** tables. The resulting set of data also contains **unmatched** rows from the left, right, or both tables.



The result of a **Left Outer Join or Merge** produces matched rows from both tables while preserving all unmatched rows from the left table. The result is illustrated by the shaded areas A and AB in **Figure 2**.

Figure 2. Venn Diagram – Left Outer Join or Merge

The result of a **Right Outer Join or Merge** produces matched rows from both tables while preserving all unmatched rows from the right table. The result is illustrated by the shaded areas B and AB in **Figure 3**.



Figure 3. Venn Diagram - Right Outer Join or Merge



The result of a **Full Outer Join or Merge** produces matched rows while preserving all unmatched rows from both tables. The result is illustrated by the shaded areas A, AB, and B in **Figure 4**.

Figure 4. Venn Diagram – Full Outer Join or Merge

All of these Joins and Merges have an important common denominator ~ each of them requires a like column or the same variable name for matching. Thus, we now return to the core focus of this presentation...



Figure 5. Venn Diagram - Tables Without Like Columns or the Same Variable Name

What do you do when the tables you want to analyze do not contain like columns or the same variable name (**Figure 5**) and a standard Join or Merge cannot be used?



Illuminating the Paradox of the Joinless Join



The development of the **Joinless Join** came about during a recent project when the need arose to overcome the limitations of a standard Join and to resolve unforeseen issues which occurred with a **One-Way Frequency**.

SAS Highlight

A One-Way Frequency contains a distribution list of values, counts, and percentages for a column.



We design a Program Node to create a source table:



DATA SMILEY COMPANY; LENGTH Special Person \$20 Special Number 8 Special Code \$1 Load Date 8; FORMAT Load Date date9.; INFILE DATALINES DELIMITER=','; INPUT Special Person \$ Special Number Special Code \$ Load Date; DATALINES; Smiley,10127911, ,20090 Smiley's Son,10173341,K,20090 Smiley's Twin,10376606,B,20090 Smiley's Wife, 10927911, A, 20090 Smiley's Son, 11471884, E, 20090 Smiley's Twin, 11573691, G, 20090 Smiley's Daughter,11975386,C,20090 Smiley's Son,12071884,J,20090 Smiley's Son, 12871884, D, 20090 Smiley's Twin, 13173691, A, 20090 Smiley's Wife, 13771202, D, 20090 Smiley's Daughter, 13775498, H, 20090 Smiley's Son,14171884,I,20090 Smiley's Twin,15373691,F,20090 Smiley's Son, 15471884, C, 20090 Smiley's Son,16074330,H,20090 Smiley's Daughter, 16175498, B, 20090 Smiley's Wife, 16176964, I, 20088 Smiley, 16279111, E, 20090 Smiley's Twin, 16573691, K, 20090 RUN; This is the code you will need to recreate this table.

The Program Node creates the SMILEY_COMPANY source table:

SMTLEV		-
JIVILET	COMPANY	

	A Special Person	Special Number	A Special Code	- Load Date
1	Smiley	10127911	Special_code	02 IAN2015
2	Smilev's Son	10127311	ĸ	02JAN2015
2	Smiley's Twin	10376606	R	02.0AN2015
4	Smiley's Wife	10927911	Δ	02JAN2015
5	Smiley's Son	11471884	E	02JAN2015
6	Smiley's Twin	11573691	G	02JAN2015
7	Smiley's Daughter	11975386	С	02JAN2015
8	Smiley's Son	12071884	J	02JAN2015
9	Smiley's Son	12871884	D	02JAN2015
10	Smiley's Twin	13173691	A	02JAN2015
11	Smiley's Wife	13771202	D	02JAN2015
12	Smiley's Daughter	13775498	Η	02JAN2015
13	Smiley's Son	14171884	1	02JAN2015
14	Smiley's Twin	15373691	F	02JAN2015
15	Smiley's Son	15471884	С	02JAN2015
16	Smiley's Son	16074330	H	02JAN2015
17	Smiley's Daughter	16175498	В	02JAN2015
18	Smiley's Wife	16176964	1	31DEC2014
19	Smiley	16279111	E	02JAN2015
20	Smiley's Twin	16573691	K	02JAN2015

- The SMILEY_COMPANY table is used throughout this presentation.
- This table contains each Special Person, Special Number, and Special Code of the ⁽²⁾ Smiley Company ⁽³⁾ employees.
- Load_Date is the date when each row was created.

This Query creates the SMILEY_CONTROL_VALUE table:



Query for SMILEY_CONTROL_VALUE for SASMain:WORK.SMILEY_COMPAN									
Query name: Query for SMILEY_CONTROL_VALUE									
Computed Columns S Prompt Manager R Preview R Tools • P Options •									
It (SMILEY_COMPANY) Special_Person Special_Number Special_Code Load_Date Computed Columns Special_Person_Flag Special_Person_Flag Special_Person_Flag Special_Code_Flag Load_Date_Flag	Column Name Special Person_Flag Special_Number_Flag Load_Date_Flag Special_Person Special_Number Special_Code Load_Date	Identifier Special_Person_Flag Special_Number_Flag Special_Code_Flag Load_Date_Flag t1.Special_Person t1.Special_Number t1.Special_Code t1.Load_Date							

Please see Appendix A to learn how to create Computed Columns and see Appendix B for the Base SAS code which corresponds to each example. A Control Value table is created in which Computed Columns are set to 1 if any data is missing in the SMILEY_COMPANY table: Special_Person_Flag: CASE

WHEN t1.Special_Code = '' THEN 1
ELSE 0
END
Special_Number_Flag:
CASE

WHEN t1.Special_Number = 0 THEN 1 WHEN t1.Special_Number is missing THEN 1 ELSE 0

END

Special_Code_Flag:

CASE WHEN t1.Special_Code = '' THEN 1 ELSE 0

END

Load_Date_Flag:

CASE WHEN t1.Load_Date = . THEN 1 ELSE 0 END

The output is filtered to include only rows where a flag is set to 1:







This One-Way Frequency is setup to automatically send an email when this project is run.

Then one day NOTHING was missing from the SMILEY_COMPANY table...

- To replicate this scenario you will need to perform the following:
 - Replace the Smiley, 10127911, ,20090 DATALINE with Smiley, 10127911, A, 20090 in the SMILEY_COMPANY Program Node on Page 6 and rerun to have no missing data in the table.
 - Rerun the Query for the SMILEY_CONTROL_VALUE table and the Control Value Report One-Way Frequency.



🦺 This report is from an older run of the task or program. Output from the most recent run was not available.

This warning message unfortunately means that we are looking at the previous successful run of this One-Way Frequency instead of the current results which we are seeking.

When the Smiley_Company table processed error free and no data was missing for the first time, it was ironic that the resulting empty Smiley_Control_Value table caused the One-Way Frequency to **not** run! Consequently, the previous results were generated on the monthly report instead of the current results.

Here is a review of the One-Way Frequency issue before we explore the solution:

- When data is missing in the Smiley_Company table a row is created in the Smiley_Control_Value table with the column flags set to 1.
- When the Smiley_Control_Value table is populated with at least 1 row the One-Way Frequency runs correctly and generates current results.
- However, when data is not missing from the Smiley_Company table no rows are created in the Smiley_Control_Value table.
- When the Smiley_Control_Value table is created empty the One-Way Frequency does not run correctly and does not generate current results but instead displays the previous results.
- In summary, the One-Way Frequency runs correctly and generates current results only when the Smiley_Control_Value table is populated with at least 1 row created by missing data detected in the Smiley_Company table.



In response to this dilemma, SAS Intuition kicked in and a quest was undertaken to find a permanent workaround solution that would enable the project to run successfully – **even if all the tables were empty.**

Here is the solution which arose during the quest to resolve this issue:

- Create a Smiley_Control_Value_Row_Count table with the row count of the Smiley_Control_Value table.
- Create a Smiley_Control_Value_Mock_Row table based upon an indirect relationship between the Smiley_Control_Value_Row_Count table and the Smiley_Company table.
- When the Smiley_Control_Value table is populated with rows, the Smiley_Control_Value_Row_Count table will contain a non-zero row count, and the Smiley_Control_Value_Mock_Row table will be created empty.
- When the Smiley_Control_Value table is empty, the Smiley_Control_Value_Row_Count table will contain a zero row count, and the Smiley_Control_Value_Mock_Row table will be created with 1 mock row of column flags set to 0.
- Append the Smiley_Control_Value table and the Smiley_Control_Value_Mock_Row table to ensure that the appended output is always populated with either real data or mock data instead of being created empty.
- Use this appended output as the input to the One-Way Frequency to enable it to always run correctly and to generate current results.

Always Remember, It's Too Soon To Quit! Bob Wieland (Mr. Inspiration)







No Problem ~

We will use a Joinless Join based upon an indirect relationship between the tables.

	LET_COMPANY ·					
罚 同	ilter and Sort 🏨 Que	ry Builder Data 👻 I	Describe 👻 Graph 👻	Analyze - Exp		
	🔌 Special_Person 🧕	Special_Number	🔌 Special_Code 🚺	Load_Date	(• • 🖣	
1	Smiley	10127911	4	02JAN2015	\neg \mathbf{J}	
2	Smiley's Son	10173341	٢	02JAN2015		
3	Smiley's Twin	10376606	3	02JAN2015		
1	Smiley's Wife	10927911	4	02JAN2015		
5	Smiley's Son	114/1884 1		02JAN2015		
5	Smiley's I win	11075000	3	02JAN2015		
<u>/</u>	Smiley's Daughter	119/5386	-	02JAN2015		
, ,	Smiley's Son	1207 1004 0	, ר	02JAN2015	S	
10	Smiley's Twin	13173691	4	02JAN2015		
11	Smiley's Wife	13771202	D	02JAN2015		
12	Smiley's Daughter	13775498	4	02JAN2015		1) I)
13	Smiley's Son	14171884		02JAN2015		
4	Smiley's Twin	15373691	=	02JAN2015	•	
15	Smiley's Son	15471884 (C	02JAN2015		
6	Smiley's Son	16074330 I	4	02JAN2015	MILEY_CONTROL_VALUE_ROW	_COONT +
17	Smiley's Daughter	16175498	В	02JAN2015	🖞 Filter and Sort 🛛 🕮 Query Bu	ilder Data 🕶 Describ
18	Smiley's Wife	16176964 I		31DEC2014		
9	Smiley	16279111	Ξ	02JAN2015		
20	Smiley's Twin	16573691	<	02JAN2015		
ill ec	LEY_CONTRO ause it conta e a Joinless Jo	L_VALUE_R(ins the row oin to create	OW_COUNT count of the a Cartesiar	table indir error rows Product ba	ectly relates to s in the SMILEY_ ased upon this in	the SMILEY COMPANY ta ndirect relat
iiz	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product o nut Data) 🛄 Coc	L_VALUE_R(ins the row oin to create	DW_COUNT count of the a Cartesiar and SMILEY_CONTI put Data	table indir error rows n Product ba	rectly relates to s in the SMILEY_v ased upon this in ount -	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product o in Input Data i i Coo Modify Task i	L_VALUE_R(ins the row oin to create of SMILEY_COMPANY de Company is Filter and Sort	DW_COUNT count of the a Cartesiar and SMILEY_CONTI put Data guery Builder Da	table indir error rows n Product ba ROL_VALUE_ROW_C	ectly relates to s in the SMILEY_ ased upon this in OUNT ~	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product o input Data i co Modify Task i Special_Perso	L_VALUE_R(ins the row oin to create of SMILEY_COMPANY de 1 1 Log 2 Out Filter and Sort 2 n 2 Special_Numb	DW_COUNT count of the a Cartesian and SMILEY_CONTI put Data g Query Builder Data Special_Cod	table indir error rows n Product ba ROL_VALUE_ROW_C ata + Describe + C e Load_Date	ectly relates to s in the SMILEY_ ased upon this in OUNT - iraph - Analyze - Export SMILEY_CONTROL	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product o input Data Coo S Modify Task of Special_Perso Smiley	L_VALUE_R(ins the row oin to create of SMILEY_COMPANY de i log i out Filter and Sort in Special_Numb 101279	DW_COUNT count of the a Cartesian and SMILEY_CONTI put Data g Query Builder Dz er Special_Cod	table indir e error rows n Product ba ROL_VALUE_ROW_CC ata ~ Describe ~ CC e Load_Date 02JAN2016	ectly relates to s in the SMILEY_ ased upon this in OUNT - iraph + Analyze + Export	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product o he C	L_VALUE_R(ins the row oin to create f SMILEY_COMPANY de i log i Out Filter and Sort i f Special_Numb 101279 101733	DW_COUNT count of the e a Cartesiar and SMILEY_CONTI put Data g Query Builder Da er Special_Coot 11 A 41 K	table indir e error rows n Product ba ROL_VALUE_ROW_C ata ~ Describe ~ C e Load_Date 02JAN2015 02JAN2015	ectly relates to s in the SMILEY_' ased upon this in OUNT ~ iraph ~ Analyze ~ Export	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product o public Data [2000 miley 1000 Smiley 1	L_VALUE_RC ins the row oin to create of SMILEY_COMPANY de i log i out Filter and Sort i Nilley Special_Numb 101279 101733 103766	DW_COUNT count of the e a Cartesiar and SMILEY_CONTI put Data query Builder Da er Special_Cod 11 A 41 K 06 B	table indir e error rows n Product ba ROL_VALUE_ROW_C ata ~ Describe ~ C e Load_Date 02JAN2015 02JAN2015	ectly relates to s in the SMILEY_ ased upon this in OUNT ~ iraph ~ Analyze ~ Export SMILEY_CONTROL_	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product o nput Data Coo Smiley Smiley's Son Smiley's Son Smiley's Son	L_VALUE_RC ins the row oin to create f SMILEY_COMPANY de i log i out Filter and Sort i Niter and Sort i Special_Numb 101279 101733 103766 109279	DW_COUNT count of the e a Cartesian and SMILEY_CONTI put Data guery Builder Da er Special_Cod 11 A 41 K 06 B 11 A	table indir error rows n Product ba ROL_VALUE_ROW_C ata • Describe • C e II Load_Date 02JAN2016 02JAN2016 02JAN2016	ectly relates to s in the SMILEY_ ased upon this in OUNT - iraph - Analyze - Export	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product o nput Data Co Modify Task Modify Task Smiley's Son Smiley's Son Smiley's Wife Smiley's Vife	L_VALUE_RC ins the row oin to create of SMILEY_COMPANY de Company Filter and Sort Tol279 101733 103766 109279 114718	A COUNT Count of the a Cartesian and SMILEY_CONTH but Data Query Builder Da provide B 11 A 41 K 06 B 11 A 42 E 00 C	table indir error rows n Product ba ROL_VALUE_ROW_C ata + Describe + C 02JAN2015 02JAN2015 02JAN2015 02JAN2015	rectly relates to s in the SMILEY_ ased upon this in OUNT - iraph - Analyze - Export SMILEY_CONTROL	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product of he Cartesian	L_VALUE_RC ins the row oin to create of SMILEY_COMPANY de i log i or Filter and Sort i filter and Sort	DW_COUNT count of the a Cartesian and SMILEY_CONTI put Data guery Builder Dz er Special_Cod 11 A 41 K 06 B 11 A 84 E 91 G	table indir error rows Product bi ROL_VALUE_ROW_CO ata ~ Describe ~ CO e I Load_Date 02JAN2016	ectly relates to s in the SMILEY_ ased upon this in OUNT - iraph - Analyze - Export SMILEY_CONTROL_	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product of he Cartesian Product of Smiley's Son Smiley's Son	L_VALUE_RC ins the row oin to create of SMILEY_COMPANY de illog illog out Filter and Sort illo Special_Numb 101279 101733 103766 109279 114718 115736 119753	DW_COUNT count of the a Cartesian and SMILEY_CONTI put Data guery Builder Da Special_Cod 11 A 41 K 06 B 11 A 44 E 91 G 56 C	table indir error rows Product bases ROL_VALUE_ROW_CO ata - Describe - CO table - Describe -	rectly relates to s in the SMILEY_' ased upon this in OUNT - iraph - Analyze - Export SMILEY_CONTROL_	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product of he Cartesian Product of Smiley's Son Smiley's Son	L_VALUE_RC ins the row oin to create f SMILEY_COMPANY de i log i Out Filter and Sort i f Special_Numb 101733 103766 109279 114718 115736 119753 120718	DW_COUNT count of the a Cartesiar and SMILEY_CONTI put Data Query Builder Da Special_Cod 11 A 41 K 06 B 11 A 42 E 91 G 86 C 84 J 94 P	table indir error rows Product ba ROL_VALUE_ROW_C ata - Describe - C e Load_Date 02JAN2016	ectly relates to s in the SMILEY_' ased upon this in OUNT ~ iraph ~ Analyze ~ Export SMILEY_CONTROL_	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product of he Cartesian	L_VALUE_RC ins the row oin to create f SMILEY_COMPANY de i log 0 out Filter and Sort 1 n Special_Numb 101279 101733 103766 109279 114718 115736 119753 120718 122718 122718 122718 122718 122718	DW_COUNT count of the a Cartesiar and SMILEY_CONTH put Data Query Builder Da Special_Cod 11 A 41 K 06 B 11 A 41 K 06 B 11 A 42 Secial_Cod 13 A 44 B 45 C 46 C 46 J 46 D 16 C 16 C	table indir error rows Product bases ROL_VALUE_ROW_C ata Describe COLANZONE 02JAN2018 02JAN20	ectly relates to s in the SMILEY_ ased upon this in OUNT ~	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product of he Cartesian	L_VALUE_RC ins the row oin to create f SMILEY_COMPANY de i log i out Filter and Sort i f Special_Numb 101279 101733 103766 109279 114718 115736 119753 120718 128718 131736	DW_COUNT count of the a Cartesiar and SMILEY_CONTH put Data Query Builder Da count of the put Data Query Builder Da count of the Special_Cool 11 A 41 K 06 B 11 A 41 K 06 B 11 A 41 K 06 B 11 A 42 C 11 A 43 Special_Cool 11 A 44 J 44 J 45 D 11 A 46 C 12 A 12 C 12 C	table indir error rows Product back ROL_VALUE_ROW_C ata Describe Col_VALUE_ROW_C table Col_VALUE_ROW_C Col_VALUE Col_VALUE Col_VALUE Col_VALUE Col_VALUE Col_VALUE Col_VALUE Col_VALUE Col_VALUE COL_VALUE C	ectly relates to s in the SMILEY_ ased upon this in OUNT - iraph - Analyze - Export SMILEY_CONTROL	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product of he Cartesian Product of Smiley's Son Smiley's Son	L_VALUE_RC ins the row oin to create of SMILEY_COMPANY de Company filter and Sort Company filter and S	DW_COUNT count of the a Cartesian and SMILEY_CONTI put Data Query Builder Da er Special_Cod 11 A 41 K 06 B 11 A 42 E 91 G 65 C 84 J 84 D 91 A 02 D 91 A 02 D 84 H	table indir error rows Product base ROL_VALUE_ROW_C ata • Describe • C e III Load_Date 02JAN2016 02JAN2017 02JAN2018 02JAN2019	ectly relates to s in the SMILEY_ ased upon this in OUNT - iraph - Analyze - Export SMILEY_CONTROL_	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product of he Cartesian	L_VALUE_RC ins the row oin to create of SMILEY_COMPANY de i log i or Filter and Sort i filter and Sort i Special_Numb 103766 109279 114718 115736 119753 120718 128718 131736 13774 141718	DW_COUNT count of the a Cartesian and SMILEY_CONTI put Data guery Builder Dz Special_Cod 1 A 41 K 06 B 11 A 42 E 91 G 66 C 84 J 91 G 66 C 84 J 91 A 02 D 91 A 02 D 938 H	table indir error rows Product bases ROL_VALUE_ROW_CO ata - Describe - CO table - Describe	rectly relates to s in the SMILEY_' ased upon this in OUNT - iraph - Analyze - Export SMILEY_CONTROL	the SMILEY COMPANY ta ndirect relat
iz T 1234567899111111	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product of he Cartesian Product Modify Task Smiley's Son Smiley's Son	L_VALUE_RC ins the row oin to create of SMILEY_COMPANY fe illog illog com Filter and Sort illog Special_Numb 101279 101733 103766 109279 114718 115736 119753 128718 131736 137754 131756 1157754	DW_COUNT count of the a Cartesian and SMILEY_CONTI put Data guery Builder Da special_Cod 11 A 41 K 06 B 11 A 44 E 91 G 65 C 84 J 91 G 66 C 84 J 91 A 10 D 11 A 11 A 11 A 11 A 11 A 11 A 11 A 11	table indir error rows Product ba ROL_VALUE_ROW_CO ata - Describe - CO tata - Describe - Descri	rectly relates to s in the SMILEY_ ased upon this in OUNT - iraph - Analyze - Export S SMILEY_CONTROL_	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product of he Cartesian Product Modify Task Smiley's Son Smiley's Twin Smiley's Son Smiley's Twin Smiley's Son Smiley's Twin Smiley's Twin Smiley's Twin Smiley's Twin Smiley's Twin Smiley's Twin Smiley's Son Smiley's Son Smiley's Son Smiley's Son Smiley's Son Smiley's Son Smiley's Son	L_VALUE_RC ins the row oin to create of SMILEY_COMPANY de into create de	DW_COUNT count of the a Cartesian and SMILEY_CONTI put Data Query Builder Da Special_Cod 11 A 41 K 06 B 11 A 44 E 91 G 86 C 84 J 91 A 02 D 98 H 84 I 91 F 84 C	table indir error rows Product ba ROL_VALUE_ROW_C tata - Describe - C tata - Describe -	ectly relates to s in the SMILEY_' ased upon this in OUNT - iraph - Analyze - Export SMILEY_CONTROL_	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product of he Cartesian Product Modify Task Smiley's Son Smiley's Son	L_VALUE_RC ins the row oin to create f SMILEY_COMPANY de i log i Out Filter and Sort i f Special_Numb 101733 103766 109279 114718 115736 119753 120718 128718 128718 131736 131736 131736 131736 131736 131736 131736 131736 131736 131736 131736 131736 131736 131736 131736 131736 131736 131736 131736 131741 131736 131741 131736 131741 13174	DW_COUNT count of the a Cartesiar and SMILEY_CONTI put Data Query Builder Da Special_Cod 11 A 41 K 06 B 11 A 41 K 06 B 11 A 42 E 91 G 86 C 84 J 91 A 02 D 98 H 84 I 91 A 02 D 98 H 84 I 91 F 84 C 03 H	table indir error rows Product ba ROL_VALUE_ROW_C tata - Describe - C tata - Describe - D	ectly relates to s in the SMILEY_ ased upon this in OUNT - iraph - Analyze - Export SMILEY_CONTROL_	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product of he Cartesian	L_VALUE_RC ins the row oin to create of SMILEY_COMPANY is Special_Numb for	DW_COUNT count of the a Cartesian and SMILEY_CONTH put Data Query Builder Da Count of the Special_Cool 11 A 41 K 06 B 11 A 42 K 07 K 11 K 06 B 11 A 42 K 07 K 11 K 06 B 11 A 42 K 07 K 11 K 12 K 1	table indir error rows Product bases ROL_VALUE_ROW_C ata Describe COL_VALUE_ROW_C table COL_VALUE_ROW_C COL_VALUE	rectly relates to s in the SMILEY_ ased upon this in OUNT - iraph - Analyze - Export SMILEY_CONTROL	the SMILEY COMPANY ta ndirect relat
	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product of he Cartesian	L_VALUE_RC ins the row oin to create of SMILEY_COMPANY de Organization of SMILEY_COMPANY de Organization de Organiza	DW_COUNT count of the a Cartesian and SMILEY_CONTI put Data guery Builder Da composition of the guery Builder Da composition of the special_Cool 1 A 41 K 50 B 50 B 11 A 42 E 91 G 50 B 11 A 44 E 91 G 50 B 11 A 44 E 91 G 50 C 50 B 11 A 50 C 50 C 50 C 50 C 50 C 50 C 50 C 50 C	table indir error rows Product bases ROL_VALUE_ROW_CO ata - Describe - CO table - CO COLANZONE 02JAN2016 02JAN20 02JAN2016 02JAN2016 02JAN2016 02JAN2016	ectly relates to s in the SMILEY_ ased upon this in OUNT - iraph - Analyze - Export SMILEY_CONTROL	the SMILEY COMPANY ta ndirect relat
11 20 12 12 12 12 12 12 12 12 12 12 12 12 12	LEY_CONTRO ause it conta e a Joinless Jo he Cartesian Product of he Cartesian	L_VALUE_RC ins the row oin to create of SMILEY_COMPANY fe i log i or Filter and Sort i of Special_Numb i Special_Numb i Special_Numb i 103766 109279 114718 115736 119753 120718 137754 137754 137754 137754 141718 153736 154718 154718 154718 166743 161754 162751	OW_COUNT count of the a Cartesiar and SMILEY_CONTI put Data guery Builder joury Builder Diagonal guery Builder Diagonal Special_Cod 11 A B6 C B7 G B6 C B1 A D2 D1 A B2 B4 D1 F B2 C D30 H B1 B2 B3 B4 D31 E B4 D31 B4 B4 B4 B4 B4 B4 B5 B6 B7 B8	table indir error rows Product ba ROL_VALUE_ROW_CC ata - Describe - CC table Col_VALUE_ROW_CC COL_VALUE COL_VALU	rectly relates to s in the SMILEY_ ased upon this in OUNT - iraph - Analyze - Export S S S S S S S S S S S S S S S S S S S	the SMILEY COMPANY ta ndirect relat

The Joinless Join automatically creates a Cartesian Product which places the 1 row and 1 column of the SMILEY_CONTROL_VALUE_ROW_COUNT table to the right of each of the 20 rows and 4 columns in the SMILEY_COMPANY table.

SAS Highlight

A Cartesian Product is a result set of all the possible rows and columns contained in 2 or more tables. The resulting set of data can be extremely large and unwieldy. The DATA Step does not easily lend itself to creating a Cartesian Product thus PROC SQL is the desired approach. Its most noticeable coding characteristic is the absence of a WHERE-clause. Although rarely produced, a Cartesian Product Join nicely illustrates a base (or internal representation) for all Joins.









Create	→ J =- SMILEY_CO.		Final ntrol V	HTML - Final
				Value Report for Smiley Company
Fin	al Control Smiley	Value R Compa	eport for ny	
	The FRE	Q Proced	ure	
Special_Person_Flag	J Frequenc	y Percer	Cumulative t Frequenc	e Cumulative y Percent
()	1 100.0	0	1 100.00
Special_Number_Fla	g Frequen	cy Percer	Cumulativ nt Frequenc	e Cumulative cy Percent
	0	1 100.0	00	1 100.00
Special_Code_Flag	Frequency	Percent	Cumulative Frequency	Cumulative Percent
0	1	100.00	1	100.00
Load_Date_Flag	Frequency	Percent	Cumulative Frequency	Cumulative Percent
0	1	100.00	1	100.00

The One-Way Frequency correctly displays that all 4 flags are set to 0 and therefore no data is missing – thanks to the Joinless Join ⁽²⁾.



0 Oh but wait... your new friend, the Joinless Join, is just getting started! 0

Next we design another Program Node to create 3 additional tables:



This is the code you will need to recreate these tables.

Here are the 3 additional tables the Program Node creates:

SPECIAL_NUMBER_NATIONAL_AVERAGE -
🐺 Filter and Sort 🖷 Query Builder Data 🗸
Special_Number_National_Average
1 12000000
LOAD_DATE_CHECK -
🚯 🐺 Filter and Sort 🕮
Load_Date_Check
1 01JAN2015
SPECIAL_CODE_NATIONAL_FOCUS -
🐺 Filter and Sort 🖳 Query Builder Da
A Special_Code_National_Focus
1 <u>K</u>

- The Special_Number_National_Average table contains the average of all the Special_Number columns from each Smiley Company nationwide which we will use in a Joinless Join to calculate a percentage of the Special_Number column in our SMILEY_COMPANY table.
- The Load_Date_Check table contains a Load Date which we will use in a Joinless Join to validate that all of our SMILEY_COMPANY table rows were created in 2015.
- The Special_Code_National_Focus table contains a Special Code from the Smiley Company National Headquarters which we will use in a Joinless Join to filter our SMILEY_COMPANY table output.

De	esigning	a Joinless	Join to perfe	orm a Calculat	tion:
				TI	
III.	Juany for SMILEY I		TION for SASMain-MORK SMI		_
	aru name: Oueru for SMI				IL ATION
	Add Tables X Delete	Prompt Manager 100 Previe	Select Data Filter Data Sort D.	ata	
	1 (SMILEY_COMP	ANY)	Column Name I	dentifier Summary Fo	rmat
	Special_Person Special_Number Special_Code		Special_Person t Special_Number t	1.Special_Person 1.Special_Number	
	Load_Date	ER NATIONAL AVERAGE)	Special_Code t Load_Date t	1.Special_Code 1.Load_Date	
	Computed Columns	_National_Average	Special_Number_Percent S	ipecial_Number_Percent PE	RCENT8.1
	Special_Number	_Percent			
 Build a Quer Headquarters 	ry with th Special NI	NE SMILEY_C	OMPANY table	and the Smiley	Company National
neauquarters	SFECIAL_ING		UNAL_AVERAGE	able.	
	💒 Table	s and Joins			
	Ad	d Tables 🔀 Delete 🗄 Pro	operties 📑 Join Order 🖷 Table 🔹	🖌 🞦 Options 👻 🕆 Move Up	
			1		
	Spe	cial_Person	t2 (SPECIAL_NUMB Special_Number_Nation	ER_NATIONAL_AVERAGE)	
	Spe	cial_Number cial_Code			
	Loa	d_Date			
• m1 · · 1					
Ine joinless indirectly role	Join is bas	sed upon the	e SPECIAL_NUM	BER_NATIONAL_AV	ERAGE table which
Special Number	ntes to the	SMILET_COM	PANY LADIE DECA	tuse it contains th	le average of all the
Special_Nullibe		ii oili eacii Sm		lable nationwide.	
Th	e Cartesian product o	f SMILEY_COMPANY and	SPECIAL_NUMBER_NATIONAL_	AVERAGE -	
E	🕽 Input Data (2) 🛛 🧾 (Code 📋 Log 🔡 Outpu	ut Data		
<u></u>	Modify Task	Filter and Sort 😐 Que	ery Builder Data - Describe -	Graph • Analyze • Export • S	end To 👻
1	Smiley	10127911 A	Special_Code iii Load_Da	015 Special_Number_National_	_Average 2000000
2	Smiley's Son	10173341 K	02JAN2	015 1	2000000
34	Smiley's Wife	10927911 A	02JAN2 02JAN2	015	2000000
5	Smiley's Son	11471884 E	02JAN2	015 1	2000000
	Smiley's Daughter	11975386 C	02JAN2	015 1	2000000
8	Smiley's Son	12071884 J	02JAN2	015 1	2000000
9	Smiley's Son	12871884 D	02JAN2	015	2000000
1	Smiley's Wife	13771202 D	02JAN2	015 1	2000000
12	2 Smiley's Daughter	13775498 H	02JAN2	015	2000000
1:	3 Smiley's Son	14171884 I	02JAN2	015	2000000
14	Smiley's Twin	15373691 F	02JAN2	015 1	2000000
1	Smiley's Son	154/1884 C 16074330 H	02.IAN2	015 1	2000000
1	7 Smiley's Daughter	16175498 B	02JAN2	015	2000000
1	Smiley's Wife	16176964 I	31DEC2	014 1	2000000
19	Smiley	16279111 E	02JAN2	015 1	2000000
20	Jonneys I Will	10073031 K	UZJANZ	viu:	2000000
* The Isimlass I	ain anton	tically and t	a Contacion D	noduct which also	and the 1 row and 1
v The Johness J	oni automa	incany create	es a cartesian P	router which place	les the I row and I
column of the	SPECIAL_NU	JMBER_NATI	ONAL_AVERAGE (table to the right o	f each of the 20 rows
and 4 columna	in the SMII	EV COMDANN	Vtabla		

	Caluma	r) ataila		
				0 . I.N. I. N.S.	1.0
	Special_N	umber_Percent t	1.Special_Number/t2.	Special_Number_Natio	onal_Average
	-				
		D			
e a	Special_Numb	er_Percent Co	omputed Colu	nn using the S	pecial_Number
LEY	Y_COMPANY t	able and the	e Special_Nun	nber_National_	Average colum
n D	- roduct roculto		• -		
пP	Touuci results	•			
	A Special Person	😥 Special Number	A Special Code	Load Date 😡	Special Number Percer
1	Smilev	10127911	A	02JAN2015	84.43
2	Smilev's Son	10173341	K	02JAN2015	84.83
3	Smilev's Twin	10376606	В	02JAN2015	86.53
4	Smilev's Wife	10927911	_ A	02JAN2015	91.13
5	Smilev's Son	11471884	E	02JAN2015	95.63
6	Smilev's Twin	11573691	G	02JAN2015	96.43
7	Smiley's Daughter	11975386	С	02JAN2015	99.83
8	Smiley's Son	12071884	J	02JAN2015	100.63
9	Smiley's Son	12871884	D	02JAN2015	107.3
10	Smiley's Twin	13173691	A	02JAN2015	109.85
11	Smiley's Wife	13771202	D	02JAN2015	114.85
12	Smiley's Daughter	13775498	Н	02JAN2015	114.85
13	Smiley's Son	14171884	1	02JAN2015	118.13
14	Smiley's Twin	15373691	F	02JAN2015	128.1
15	Smiley's Son	15471884	С	02JAN2015	128.9
16	Smiley's Son	16074330	H	02JAN2015	134.05
17	Smiley's Daughter	16175498	В	02JAN2015	134.8
18	Smiley's Wife	16176964	1	31DEC2014	134.8
19	Smiley	16279111	E	02JAN2015	135.7
	1				

Here is the final result of the SMILEY_COMPANY table with the Special_Number_Percent column to the right of each of the 20 rows and 4 columns.



SMILEY_COMPANY table.

Jmn	Details				then 'AOK'	Date ge t2.Load_Date
Date_Validation	case when t1.Load_D) ate ge t2.Load_Date_(Check then 'AO	K' else 'NOT_AOK'	end end	¢
_	_			_		
alidate a D	Date_Validation	Computed	Column u	ising the l	Load_Date co	olumn from
ALLEV COMP	ANV table and	the Load Dat	o Chock co	lumn from t	he Cartesian	Product ros
ILLI_COM	ANT table and	the Loau_Dat	e_check.co		lie cal testali	Touterrea
	A Special_Person	Special_Number	A Special_Co	de 🔢 Load_Date	e 🔌 Date_Validatio	on
1	Smiley	10127911	A	02JAN20	15 AOK	
2	Smiley's Son	10173341	К	02JAN20	15 AOK	
3	Smiley's Twin	10376606	В	02JAN20	15 AOK	
4	Smiley's Wife	10927911	Α	02JAN20	15 AOK	
5	Smiley's Son	11471884	E	02JAN20	15 AOK	
6	Smiley's Twin	11573691	G	02JAN20	15 AOK	
7	Smiley's Daughter	11975386	С	02JAN20	15 AOK	
8	Smiley's Son	12071884	J	02JAN20	15 AOK	
9	Smiley's Son	12871884	D	02JAN20	15 AOK	
10	Smiley's Twin	13173691	Α	02JAN20	15 AOK	
11	Smiley's Wife	13771202	D	02JAN20	15 AOK	
12	Smiley's Daughter	13775498	Η	02JAN20	15 AOK	
13	Smiley's Son	14171884	1	02JAN20	15 AOK	
14	Smiley's Twin	15373691	F	02JAN20	15 AOK	
15	Smiley's Son	15471884	С	02JAN20	15 AOK	
16	Smiley's Son	16074330	Η	02JAN20	15 AOK	
17	Smiley's Daughter	16175498	В	02JAN20	15 AOK	
18	Smiley's Wife	16176964	Ī	31DEC20	14 NOT AOK	
19) Smilev	16279111	E	02JAN20	15 AOK	
		10570001	-	02 14100		

Here is the final result of the SMILEY_COMPANY table with the Special_Number_Percent column to the right of each of the 20 rows and 4 columns.

Designing a Joinless Join to perform a Filtration:						
	SPECIAL_C Query for					
	MLESS JOIN FILTRATIO					
	N					
🚟 Query for SMILEY_JOINLES	OIN_FILTRATION for SASMain:WORK.SMILEY_COMPANY					
Query name: Query for SMILE	JOINLESS_JOIN_FILTRATION Output name: WORK.SMILEY_JOINLESS_JOIN_FILTRATION					
🧱 Computed Columns 🍕	Prompt Manager 📷 Preview 🏠 Tools 👻 🔛 Options 🔹					
🖽 Add Tables 🗙 Delete	Join Tables Select Data Filter Data Sort Data					
□ I (SMILEY_COMPAN	Column Name Identifier Summary Format					
	Special_Number 11.Special_Number Secial_Number					
□ Load_Date	TIONAL FOCUS)					
Special_Code_Nat	nal_Focus					
Build a Query with the	SMILEY_COMPANY table and the Smiley Company Nat	tional				
Headquarters SPECIAL_COD	_NATIONAL_FOCUS table.					
🕷 Tables a	loins					
Hade	Jes 🔀 Delete 🗉 Properties 動 Join Order 🕮 Table - 😭 Options - 🕆 Move					
t1 (Si Specia	EY_COMPANY t2 (SPECIAL_CODE_NATIONAL_FOCUS) Person Special Code National Focus					
Specia Specia	Aumber Code					
Load_	e					
The Joinless Join is based a	non the SDECIAL CODE NATIONAL FOCUS table which indi	noatlu				
* The Johness John is Dased relates to the SMILEV COMP	NV table because it contains the Special Code to be focused	unon				
nationwide within the Speci	Code column in the SMILEY COMPANY table.	upon				
The Cartesian product of S	LEY_COMPANY and SPECIAL_CODE_NATIONAL_FOCUS +					
Input Data (2)	Log 🛃 Output Data					
	Special Number Aspecial Code H Load Date Aspecial Code National Focus					
1 Smiley	10127911 A 02JAN2015 K					
2 Smiley's Son 3 Smiley's Twin	10376606 B 02JAN2015 K					
4 Smiley's Wife	10927911 A 02JAN2015 K					
5 Smiley's Son 6 Smiley's Twin	114/1884 E 02JAN2015 K 11573691 G 02JAN2015 K					
7 Smiley's Daughter	11975386 C 02JAN2015 K					
8 Smiley's Son	12071884 J 02JAN2015 K					
9 Smiley's Son	12871884 D 02JAN2015 K					
11 Smiley's Wife	13773031 A 02JAN2015 K					
12 Smiley's Daughter	13775498 H 02JAN2015 K					
13 Smiley's Son	14171884 I 02JAN2015 K					
14 Smiley's Twin	15373691 F 02JAN2015 K					
15 Smiley's Son 16 Smiley's Son	16074330 H 02JAN2015 K					
17 Smiley's Daughter	16175498 B 02JAN2015 K					
18 Smiley's Wife	16176964 I 31DEC2014 K					
19 Smiley 20 Smiley's Twin	16279111 E 02JAN2015 K 16573691 K 02JAN2015 K					
ZU [Smiley's twin						
♦ The Joinless Join automatic	ally creates a Cartesian Product which places the 1 row :	and 1				
	NATIONAL FOCUS toking the state of the second					
column of the SPECIAL_COD	_NATIONAL_FUCUS TABLE to the right of each of the 20 rows a	ana 4				
columns in the SMILEV COM	ANV table					

Select Data	Filter Data Sort Data							
Filter the ra	Filter the raw data							
🖃 📑 Whe	🖃 🛃 Where							
- T t	1.Special_Code = t2.Special_Code_National_Focus							

Filter the raw data to include the rows where the value of the Special_Code column from the SMILEY_COMPANY table is equal to the value of the Special_Code_National_Focus column from the Cartesian Product results.

	🔌 Special_Person	😡 Special_Number	🔌 Special_Code	Load_Date
1	Smiley's Son	10173341	К	02JAN2015
2	Smiley's Twin	16573691	K	02JAN2015

Here is the final result of the SMILEY_COMPANY table with the Special_Code column filtered by the Special_Code_National_Focus column.



_	Input Data (5)	Code	🔲 Log	8	Dutput Da	ita							
\$5	<u>N</u> Modify Task 🛛	🐺 Filt	ter and So	rt 🖳	Query B	uilder Data 🕶	Describe 👻 Graph 👻	Analyze 👻 E	xport +	Send To 👻			
	A Special_Pers	on 🔞	Special_		Special_	Load_Date		ROL_VALUE_	D Spei	cial_Number_	Load	Date_ 🔥 Spe	cial_Code_N
	Smiley	-	10127911	Δ	Coue	02JAN2015		0	- 1100	1200000	0 01,14	N2015 K	
2	Smiley's Son		10173341	ĸ		02JAN2015	5	0		1200000	0 01JA	N2015 K	
3	Smiley's Twin		10376606	В		02JAN2015	5	0		1200000	0 01JA	N2015 K	
4	Smiley's Wife		10927911	A		02JAN2015	5	0		1200000	0 01JA	N2015 K	
5	Smiley's Son		11573691	G		02JAN2015 02.IAN2015		0		1200000	0 01JA	N2015 K N2015 K	
7	Smiley's Daughter		11975386	c		02JAN2015	5	0		1200000	0 01JA	N2015 K	
8	Smiley's Son		12071884	J		02JAN2015	5	0		1200000	0 01JA	N2015 K	
9	Smiley's Son		12871884	D		02JAN2015	5	0		1200000	0 01JA	N2015 K	
10	Smiley's Iwin		13771202	A		02JAN2015 02.IAN2015		0		1200000	0 01JA	N2015 K N2015 K	
12	Smiley's Daughter		13775498	H		02JAN2015	5	0		1200000	0 01JA	N2015 K	
13	Smiley's Son		14171884	I		02JAN2015	5	0		1200000	0 01JA	N2015 K	
14	Smiley's Twin		15373691	F		02JAN2015		0		1200000	0 01JA	N2015 K	
15	Smiley's Son		15471884	C H		02JAN2015		0		1200000	01JA	N2015 K N2015 K	
10	Smiley's Daughter		16175498	B		02JAN2015 02JAN2015	5	U 0		1200000	0 01JA	N2015 K	
18	Smiley's Wife	l	16176964	Ĺ		31DEC2014	•	0		1200000	0 01JA	N2015 K	
19	Smiley		16279111	E		02JAN2015	5	0		1200000	0 01JA	N2015 K	
20	Smiley's Twin		16573691	K		02JAN2015	5	0		1200000	0 01JA	N2015 K	
		Load_ Specia	Date_Flag al_Code_Fl	ag	ca ca	ase t2.SMILEY_CO ase t2.SMILEY_CO	_Date ge t4.Load_Date)NTROL_VALUE_ROW)NTROL_VALUE_ROW	_Cneck th I_COUNT whe I_COUNT whe	en Othen en Othen	Delse NOT_A Delse . end Delse . end	OK'end		
		Load_ Specia Specia Specia Specia	Date_Flag al_Code_Fl al_Code_M al_Number al_Number al_Person_	ag latch _Flag _Perce Flag	ca ca ca ent t1 ca	ase t2.SMILEY_C0 ase t2.SMILEY_C0 ase when t1.Speci ase t2.SMILEY_C0 .Special_Number/ ase t2.SMILEY_C0	_Uate ge t4.Load_Uate NTROL_VALUE_ROW NTROL_VALUE_ROW ial_Code = t5.Special_C NTROL_VALUE_ROW t3.Special_Number_Nat NTROL_VALUE_ROW	COUNT whe COUNT whe ode_National_ COUNT whe ional_Average	en Othen en Othen Focus the en Othen en Othen	else NOT_A D else . end D else . end n 'MATCH' e D else . end D else . end	OK' end else 'NO MAT	'CH' end	
e M lum N S	flock Row nns which pecial_Co	Load_ Specia Specia Specia Specia Specia Cr A Cr A ar Ode_	Date_Rag al_Code_Ri al_Code_M al_Number, al_Number, al_Person_ eatio e der _Matc	ag latch _Flag Flag n, (ive	ca ca ca ca ca ca ca ca ca ca ca ca ca c	see t2.SMILEY_CCC see t2.SMILEY_CC see when t1.Special see t2.SMILEY_CC .Special_Number/ see t2.SMILEY_CCC ulation, V the same puted Col	Late get 4.Load_Late NITROL_VALUE_ROW NITROL_VALUE_ROW ial_Code = t5.Special_C INTROL_VALUE_ROW t3.Special_Number_Nat NITROL_VALUE_ROW Validation, a e way as sho umn repres		tratic the piltr	else Nor_A 0 else end 1 else else 1 else else 1 else else 1 e	ok' end else 'NO MAT repre: 15 exar	CH'end sented	by Cor long w
e M lum w S	Aock Row Ans which Special_Co	Load_ Special	Date_Rag al_Code_Fl al_Code_M al_Number, al_Number, al_Person_ eatio e der _Matc	ag Iatch _Rag _Percr Rag n, (ive h C	Calcu d in Comp	In the same control of the second control of	Date get 4.Load_Date DitRoL_VALUE_ROW DITROL_VALUE_ROW DITROL_VALUE_ROW DIALOGE =15.Special_O DINTROL_VALUE_ROW DINTROL_VALUE_ROW DINTROL_VALUE_ROW Validation, a e way as sho umn represe Person Special	COUNT when Count Count when Count Count when Count Count when Count Co	en Othen Focus the en Othen focus the en Othen tratic the p	else i vol _A 0 else , end 0 else , end 1	ok'end else 'NO MAT repre: 15 exar Special_) Number	Sented	by Cor long w
e M lum w S	Aock Row Ans which Special_Co	Load_ Specia Specia Specia Specia Special Ode_ Special	Date_Rag al_Code_R al_Code_R al_Number al_Number al_Number al_Person_ e der Matc - 3 Sc - 3 C - 5 C - 5 C	ag latch _Rag _Perca Rag n , (ive h C	Calcu d in Comp	Here the second	Late get 4.Load_late VALUE_ROW NOTROL_VALUE_ROW VALUE_ROW VALUE_R	COUNT why COUNT why de National - COUNT why de National - COUNT why and Filt own in senting	en Othen en Othen Focus the en Othen tratio	else NOT_A 0 else end 1 else	ok end else NO MAT repre: 15 exar 15 exar 15 exar	CH' end sented nples al	by Cor long w
e N um w S	Alock Row ans which opecial_Co	Load_ Special Special Special Special Code_	Date_Rag al_Code_R al_Code_R al_Number; al_Number; al_Number; al_Person_ eatio e der 	ag latch _Rag _Perca Rag ive ive cide_ lag	Calcu d in Calcu d in Date Fla	ese t2.SMILEY_CCC see t2.SMILEY_CC see when t1.Special see t2.SMILEY_CC .Special_Number/A see t2.SMILEY_CCC ulation, V the same puted Col d g 0 Special ag	Late get 4.Load_Late, NATROL_VALUE_ROW NATROL_VALUE_ROW ial_Code = t5.Special_C NATROL_VALUE_ROW i3.Special_Number_Nat NATROL_VALUE_ROW //alidation, a e way as sho umn repres	COUNT why COUNT why de_National_ COUNT why and Filt COUNT why and Filt COUNT why and Filt COUNT why and Filt Count in Senting	tratic the p	else NOT_A 0 else end 1 else	Repres repres second special Number Percent 84.4%	CH'end sented nples al	by Cor long w
e N um w S	Alock Row ans which special_Co <u>Special_</u> S <u>Person_</u> N <u>Special_</u>	Load_ Specia Specia Specia Specia Special Ode_ Flag	Date_Flag al_Code_Fl al_Code_Mal_Number al_Number al_Person_ eatio e der Matc - (3) St - (3) St	ag latch _Flag Percial Rag ive ive cide lag 0 0	Calcu d in Calcu d in Data	se t2.SMILEY_CCC see t2.SMILEY_CC see whent1.Special see t2.SMILEY_CC .Special_Number/ see t2.SMILEY_CCC ulation, V the same puted Col de g 0 Smiley 0 Smiley Son	Late get 4.Load_Jate NATROL_VALUE_ROW NATROL_VALUE_ROW dia_Code = t5.Special_C NATROL_VALUE_ROW dia_Special_Number_Nat NATROL_VALUE_ROW //alidation, a e way as sho umn repres Special_Number 10127911 1017334	COUNT why _COUNT why ode_National _COUNT why and Filt own in senting Special A K	tratic tratic	else NOT_A 0 else end 1 else	OK end Item Provide the second secon	CH'end sented nples al Date_Valis ation	by Collong w ong w d Moman Match
e N un w S	Aock Row ans which pecial_Co <u>Special_</u> S <u>Person_</u> N 0 0 0	Load_ Specia Specia Specia Specia Specia 7 Cr 1 ar ode_ Flag	Date_Rag al_Code_R al_Code_R al_Number al_Number al_Number al_Person_ eatio e der Matc - 3 - 3 - 5 - 6 - 7 - 7 - 7 - 7 - 7 - 7 - 7 - 7	ag latch _Flag Perci Flag ive ive cide_ lag 0 0 0	Calcu d in Calcu Calcu d in Calcu ca Calcu ca Calcu ca Calcu ca Calcu ca Calcu	se t2.SMILEY_CC see t2.SMILEY_CC see t3.SMILEY_CC see t2.SMILEY_CC .Special_Number/ see t2.SMILEY_CC .see t2.SMILEY_CC .see t2.SMILEY_CC .the same puted Col .see t2.SMILEY_CC .see t2.SMILEY_CC	Late ge 14.Load_Jate INTROL_VALUE_ROW NITROL_VALUE_ROW Ial_Code = t5.Special_C INTROL_VALUE_ROW I3.Special_Number_Nat NNTROL_VALUE_ROW Validation, a e way as sho umn repres 	COUNT when COUNT	en Othen Focus the en Othen Focus the en Othen tratic the p Filtr	else NOT_A 0 else end 1 else	OK end Item Prepresent Special Number Percent 84.4% 84.8% 85.5% 01.19°	CH'end Sented I nples al AOK AOK AOK AOK	by Cor long w d Spec Coo Match NO MATC NO MATC
e N lun w S	Aock Row ans which pecial_Co <u>Special_</u> S Person_ Flag 0 0 0 0	Load_ Specia Specia Specia Specia Special of Cr n aru	Date_Flag al_Code_Fl al_Code_Mal_Number; al_Number; al_Number; al_Number; al_Person e der 	ag latch _Flag Perca Flag ive ib C b code lag 0 0 0 0 0 0 0 0 0 0 0 0 0	ca ca ca ca ca ca ca ca ca ca ca ca ca c	se t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC .Special_Number/ see t2.SMILEY_CC .special_Number/ see t2.SMILEY_CC .special_Number/ see t2.SMILEY_CC .special_ set 2.SMILEY_CC .special_ set 2.SMILEY_CC .smiley_Son 0_Smiley'Son 0_Smiley'Son 0_Smiley'Son 0_Smiley'Son 0_Smiley'Son	Late ge 14.Load_Jate INTROL_VALUE_ROW NITROL_VALUE_ROW Ial_Code = t5.Special_C INTROL_VALUE_ROW Ial_Code = t5.Special_ROW Ial_Code = t5.Special_C INTROL_VALUE_ROW Validation, a e way as sho umn repres Person Special_Number 10127911 10173341 n 10376606 e 10927901 111271824	CUERCK IN COUNT whe COUNT WHE	tratic the p Focus the en 0 then of then tratic the p Filtr	else NOT_A 0 else end 0 else end 1 else end 0 else else else 0 else end 0 else end 0 else end 0 else else 0 else 0 else 0 else else 0	OK end else 'NO MAT repre: IS exal Special_ Percent 84.4% 85.5% 95.6%	CH'end sented nples al AOK AOK AOK AOK	by Con long w d Spec Coo Matt NO MATC NO MATC NO MATC
e N lun w S	Alock Row ans which special_Co	Load_ Specia Specia Specia Specia Cr n aru ode_	Date_Flag al_Code_Fl al_Code_Mal_Number; al_Number; al_Number; al_Person e atio e der 	ag latch _Flag Perca Flag ive ib C b c b c c ag 0 0 0 0 0 0 0 0 0 0 0 0 0	ca ca ca ca ca ca ca ca ca ca ca ca ca c	se t2.SMILEY_CC see t2.SMILEY_CC see when t1.Speci see t2.SMILEY_CC .Special_Number// see t2.SMILEY_CC ulation, V the same puted Col section of Smiley Soft of Smiley's Soft of Smiley's Soft of Smiley's Soft of Smiley's Territy of Smiley's Territy of Smiley's Territy of Smiley's Territy	Late ge 14.Load_Late, DirtRoL_VALUE_ROW DNTROL_VALUE_ROW DNTROL_VALUE_ROW DNTROL_VALUE_ROW DNTROL_VALUE_ROW DNTROL_VALUE_ROW Validation, a e way as she umn represe Lerson Special Number 10127911 10173341 n 10376606 e 10327814 n 11477884 n 11477894	COUNT when COUNT	tratic tratic the p	else NOT_A 0 else end 0 else else else 0 else end 0 else end 0 else end 0 else end 0 else end 0 else else 0 else else else 0 else else 0 else else else else else 0 else else else else else else else els	OK end Ise NO MAT repre: Special_ Number Percent 84.4% 86.5% 91.1% 95.6%	CH'end Sented I Date_Vali ation AOK AOK AOK AOK AOK AOK	by Cor long w d Spectro Cox Match NO MATC NO MATCH NO MATCH
e N lun w S	Aock Row ans which pecial_Co <u>Special_</u> > N Person_ Person_ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Load_ Specia Specia Specia Specia Special ode_	Date_Flag al_Code_Fl al_Code_Fl al_Number; al_Number; al_Number; al_Person e atio e der Sr F 0 0 0 0 0 0 0 0 0 0 0 0 0	ag latch _Flag _Perci Flag ive ive ib C b b c c de _ lag 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	ca ca ca ca ca ca ca ca ca ca ca ca ca c	se t2.SMILEY_CC see t2.SMILEY_CC see t3.SMILEY_CC see t2.SMILEY_CC .Special_Number/ see t2.SMILEY_CC ulation, V the same puted Col sectamiley o Smiley's Son o Smiley's Twi o Smiley's Twi o Smiley's Twi o Smiley's Twi o Smiley's Twi	Late get 4.Load_Jate Load_Jate Load_Jate Row NATROL_VALUE_ROW NATROL_VALUE_ROW NATROL_VALUE_ROW Ial_Code = 15.Special_C NATROL_VALUE_ROW I3.Special_Number_Nat NATROL_VALUE_ROW Zalidation, a e way as sho umn represe Special_Number 10127911 10173341 n 10376606 e 10927911 11471884 n 1157691 1097384 n 11573691	COUNT why COUNT why COUNT why de_National_ COUNT why and Filt COUNT why and Filt COUNT why and Filt COUNT why Special A A K B A Code C C	tratic tratic Focus the en 0 then i Focus the en 0 then i Focus the en 0 then fritter Filter	else NOT_A 0 else end 0 else else 0 else else 0 else else 0 else else 0 else else 0	OK end I CEDTC: Special Special Number Percent 84.4% 84.5% 91.1% 95.6% 99.8%	CH'end Sented I nples al AOK AOK AOK AOK AOK AOK AOK	by Corlong works of the second
e N lun w S	Aock Row nns which special_Co Person_ Person_ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Load_ Specia Specia Specia Specia Special umber, Flag	Date_Flag al_Code_Fl al_Code_Fl al_Number; al_Number; al_Person_ e atio e der 	ag latch _Flag _Perci Flag ive ive ib C b b c c de lag 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Calcu Calcu Calcu d in Comp	ese t2.SMILEY_CCC see t2.SMILEY_CC see when t1.Special see t2.SMILEY_CC .Special_Number/A see t2.SMILEY_CC ulation, V the same puted Col	Late get 4.Load_Jate NatroL_VALUE_ROW NatroL_VALUE_ROW NatroL_VALUE_ROW Special_COMPTROL_VALUE_ROW Special_Number_Nat NatroL_VALUE_ROW Special_Number_Nat IntroL_VALUE_ROW Special_Number_Nat Introl_VALUE_ROW Special_Number_Nat Introl_VALUE_ROW Special_Number_Nat Introl_VALUE_ROW Special_Number_Nat Introl_VALUE_ROW Introl_VALUE_ROW Introl_VALUE_ROW Special_Number_Nat Introl_VALUE_ROW	COUNT why COUNT why count why de_National_ COUNT why and Filt own in senting Special A K B A Code G G C J	tratic tratic Focus the en 0 then i Focus the en 0 then i tratic the p Filtr	else NOT_A 0 else end 0 else else else 0 else end 0 else end 0 else end 0 else end 0 else end 0 else else 0 else 0 else else 0 else else 0 else else 0 else else 0 else 0 else 0 else 0 else 0 else 0 else else 0	Special Special Number Percent 84.4% 84.8% 86.5% 91.1% 95.6% 96.4% 99.8% 100.6%	CH'end Sented nples al AOK AOK AOK AOK AOK AOK AOK	by Corlong w Special Mo Matter NO Matter
e N lum w S	Alock Row ans which pecial_Co Person_ Person_ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Load_ Specia Specia Specia Specia Specia Special Date Special	Date_Flag al_Code_Pl al_Code_Mal_Number al_Number al_Number al_Person_ e der 	ag Iatch Flag Perce Flag ive ive ib C b code lag 0 0 0 0 0 0 0 0 0 0 0 0 0	Calcu Calcu d in Data Data Fiz	set2.SMILEY_CC see t2.SMILEY_CC see t3.SMILEY_CC see t2.SMILEY_CC .Special_Number/ set2.SMILEY_CC .special_Number/ set2.SMILEY_CC .special_Number/ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_ set2.SMILEY_CC .special_set2.SMILEY_CC .special_Number/ .specia	Late get 4.Load_Jate National Content of the second seco	CUERCK IN COUNT Why COUNT WHY	In Adv. I	else NOT_A 0 else . end 0 el	OK end It is exar Special Number Percent 84.4% 84.8% 86.5% 95.6% 95.6% 99.8% 100.6% 107.3%	CH'end Sented nples al AOK AOK AOK AOK AOK AOK AOK AOK	by Cor long w d Match NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC
e N um w S	Alock Row ans which pecial_Co person_ > N Person_ > N 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Load_ Specia Specia Specia Specia Specia Specia Special umber Flag	Date_Flag al_Code_Fl al_Number, al_Number, al_Number, al_Number, al_Person	ag latch -Flag -Perce Flag ive ih C b code - lag 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Calcu ca ca ca ca ca ca ca ca ca ca ca ca ca	se t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC .Special_Number/ see t2.SMILEY_CC .special_Number/ see t2.SMILEY_CC .special_Number/ see t2.SMILEY_CC .special_ set 2.SMILEY_CC .special_ set 2.SMILEY_CC .special_ special_	Late ge 14.Load_Jate INTROL_VALUE_ROW NITROL_VALUE_ROW Ial_Code = t5.Special_C INTROL_VALUE_ROW Ial_Code = t5.Special_C INTROL_VALUE_ROW Validation, a e way as sho umn repres Person Special Person Special Person 10127911 10173341 n 10376000 e 10927911 11471884 n 11573691 19471884 n 12071884 n 12071884 n 12071884 n 12071884 n 12071894 n 1207	COUNT whe COUNT whe count whe count whe count whe count whe count whe and Filt count whe and Filt count whe and Filt count whe count count co	In a dock of the second distance of the se	else NOT_A 0 else end 0 else else else 0 else end 0 else end 0 else end 0 else end 0 else else 0 else else else 0 else else 0 else else else 0 else else 0 else else 0 else else 0 else else 0 else else else else else else else els	OK end It is in the second se	CH'end Sented nples al AOK	by Cor long w d Spec Cod Match NO Matc NO Matc NO Matc NO Matc NO Matc NO Matc
e N lun w S	Alock Row Ans which Special_Co Special_ Person_ Person_ 0 0 0 0 0 0 0 0 0 0 0 0 0	Load Specia Specia Specia Specia Specia Special under_ Special under_	Date_Flag al_Code_Fl al_Code_Fl al_Number al_Number al_Number al_Person e der e der 	ag [atch] [Flag] Percod ive ive ib C b C b code_ lag 0 0 0 0 0 0 0 0 0 0 0 0 0	Calcu d in Data Data	se t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC .Special_Number/ see t2.SMILEY_CC .special_Number/ see t2.SMILEY_CC .special_Number/ see t2.SMILEY_CC .special_Number/ sectors .special_Number/ sectors .special_ .specialspecial_ .specialspecial_ .specialspecial_ .specialspecial_ .specials	Late get 4.Load_Late, Late get 4.Load_Late, NUTROL_VALUE_ROW 1017391 1017911 1017391 11017391 11175380 112071884 11373691 13771202	COUNT when	Tratic tratic tratic the p Filtr	eise not	OK end It is exal Special Number Percent 84.4% 86.5% 91.1% 96.4% 99.8% 100.6% 100.6% 114.8%	CH'end Sented nples al AOK	by Cor long w d Spec Cod Matc NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC
	Aock Row ans which pecial_Co Person_ Person_ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Load Special Special Special Special Special Special Decelal	Date_Flag al_Code_Flag al_Code_Mal_Number; al_Number; al_Number; al_Person e atio e der e atio e der 	ag [Aatch [Flag Percord ive h C b code lag 0 0 0 0 0 0 0 0 0 0 0 0 0	Calcu ca ca ca ca ca ca ca ca ca ca ca ca ca	Interference of the second sec	Date get 4.Load_Date Date get 4.Load_Date DATROL_VALUE_ROW NNTROL_VALUE_ROW NALDE_ROW VALUE_ROW	COUNT why COUNT why COUNT why de_National_ COUNT why and Filt Down in Senting Special Code A A K B A Code C C D A D A D A	Image of the set of	eise NOT_A D else end D else else else Else else else D else else else D else else else D else else else D else else else else else else else els	OK end I CEDTC: Special Special Number Percent 84.4% 84.5% 91.1% 95.6% 95.6% 91.1% 95.6% 100.6% 100.8% 100.8% 114.8% 14.8%	CH'end Sented nples al AOK	by Corlong w long w d Spectronal of the second Match NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC
	Aock Row ans which pecial_Co Person_ Person_ Person_ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Load Specia Specia Specia Specia Specia Special Cr Cr Cr Cr Cr Cr Cr Cr Cr Cr Cr Cr Cr	Date_Flag al_Code_V al_Code_Fl al_Code_Mai_Number; al_Number; al_Number; al_Person e atio e der Sr Sr Sr 0 0 0 0 0 0 0 0 0 0 0 0 0 0	ag [atch [Fag Percial rote Percial (ive bode lag 0 0 0 0 0 0 0 0 0 0 0 0 0	Calcu Calcu d in Data Fiz	In the same set 2.SMILEY_CC see t2.SMILEY_CC see when t1.Special see t2.SMILEY_CC .Special_Number/ .Special_Number/ .Special_Number/ .Special_Number/ .Special_Number/ .Special_Number/ .Special_ .SpecialSpecialSpecialSpecialSpecialSpecialSpecialSpecialSpecialSpecialSpecialSpecialSpecialSpecialSpeci	Date get 4.Load_Date Date get 4.Load_Date Date get 4.Load_Date Date get 4.Load_Date NNTROL_VALUE_ROW NDTROL_VALUE_ROW Ial_code = t5.Special_C NNTROL_VALUE_ROW Xalidation, a e way as shot umn represe 10127911 10173341 10173341 10173341 10173341 10173341 10173341 10173341 10173341 10173341 1017341 1147188 1207184 1217184 1317602 ughter 13775492 1417188 1417188	COUNT why COUNT why COUNT why de_National_ COUNT why and Filt own in senting Special A K B A E G G Code J L D A A E Code	Filtr	eise NOT_A 0 else . end 0 el	OK end I CEDTE: Special Special Special Sumber Percent 84.4% 84.8% 95.6% 96.4% 95.6% 100.6% 107.3% 100.6% 107.3% 104.8% 114.8% 114.8% 114.8%	CH'end Sented nples al AOK	by Cor long w d Match NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC
e N un w S	Alock Row ans which special_Co Person_ Person_ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	LoadSpecia Specia Specia Specia Specia Special 7 Cr n ar ode Special	Date_Flag al_Code_N al_Code_N al_Code_N al_Number al_Number al_Person	ag [atch [Fag Percial Fag ive h C bode lag 0 0 0 0 0 0 0 0 0 0 0 0 0	Calcu ca ca ca ca ca ca ca ca ca ca ca ca ca	set2.SMILEY_CC see t2.SMILEY_CC see t3.SMILEY_CC see t2.SMILEY_CC .Special_Number/ set2.SMILEY_CC .special_Number/ set2.SMILEY_CC .special_Number/ set2.SMILEY_CC .special_number/ set2.SMILEY_CC .special_number/ set2.SMILEY_CC .special_number/ set2.SMILEY_CC .special_number/ set2.SMILEY_CC .special_number/ set2.SMILEY_CC .special_number/ set2.SMILEY_CC .special_number/ set2.SMILEY_CC .special_number/ set2.SMILEY_CC .special_number/ set2.SMILEY_CC .special_number/ set2.SMILEY_CC .special_number/ set2.SMILEY_CC .special_number/ set2.SMILEY_CC .special_number/ set2.SMILEY_CC .special_number/ set2.SMILEY_CC .special_number/ set2.SMILEY_CC .special_number/ .special_n	Late get 4. Load_Late, Date get 4. Load_Late, NITROL_VALUE_ROW, NITROL_VALUE_ROW, NITROL_VALUE_ROW, 3. Special_CUNTROL_VALUE_ROW, Validation, a validation, a e way as shot umn repres 10127911 1017341 n 10173741 10173341 n 10173600 e 10173741 10173741 10173741 10173741 10173741 10173741 10173741 10173741 10173741 10173741 10173749 1147184 n 1147184 n 1377549 1377549 1417184 n 1537369 1471884 n 1537549	COUNT why COUNT why COUNT why count why count why and Filt count why and Filt count why and Filt count why count why and Filt count why count	- E Lo	else NOT_A D else end D else else else D else end D else end D else end D else end D else else D else else else D else else D else else else else D else else else else D else else else D else else else else else else else els	OK end Special Special Number Percent 84.4% 84.8% 95.6% 96.4% 99.8% 100.6% 107.3% 109.8% 114.8% 114.8% 114.8% 114.8% 114.8% 114.8% 129.6%	CH'end Sented nples al AOK	by Cor long w d Spec Cod Match NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC
e N um w S	Aock Row ans which pecial_Co Person_ > S Person_ > N 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	LoadSpecia Specia Specia Specia Specia Special Cr Cr n aru odeSpecial	Date_Flag al_Code_Pl al_Code_Mal_Number al_Number al_Number al_Person	ag latch _Flag _Perci Flag h C h C 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Calcu ca ca ca ca ca ca ca ca ca ca ca ca ca	se t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC .Special_Number/ see t2.SMILEY_CC .Special_Number/ see t2.SMILEY_CC .Special_Number/ see t2.SMILEY_CC .Special_ set t2.SMILEY_CC .Special_set t2.SMILEY_SPEcial_set t2.SMILEY	Late ge 14.Load_Jate John Fol_VALUE_ROW NATROL_VALUE_ROW John Fol_VALUE_ROW John Fol_VALUE_RO	CUERCK IN COUNT Wh COUNT Wh count wh count wh count wh and Fill own in senting Special A K B A E G C C J D A A E G C D A A E G C D H E C D H	- Lo	else NOT_A 0 else end 0 else else else 0 else end 0 else end 0 else end 0 else end 0 else else 0 else else else 0 else else else 0 else else 0 else else 0 else else else 0 else else else 0 else else 0 else else else else 0 else else else else 0 else else else else 0 else else else else else 0 else else else else else else else els	OK end It is exar Special Number Percent 84.4% 84.8% 86.5% 95.6% 96.4% 99.8% 100.6% 107.3% 109.8% 114.8% 114.8% 114.8% 114.8% 114.8% 118.1% 128.1% 128.9%	CH'end Sented nples al ADK AOK AOK AOK AOK AOK AOK AOK AOK AOK AO	by Cor long w d Spec Cod Match NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC
e N um w S	Aock Row ans which pecial_co person_ > N Construction Person_ > N Construction Cons	Load Specia Specia Specia Specia Specia Specia Specia Umber Flag	Date_Flag al_Code_V al_Number, al_Number, al_Number, al_Number, al_Number, al_Person_ e der 	ag latch _Flag _Perci Flag h C b	Calcu d in Omp	se t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC .Special_Number/ see t2.SMILEY_CC .Special_Number/ see t2.SMILEY_CC	Late ge 14.Load_Jate Late ge 14.Load_Jate INTROL_VALUE_ROW NNTROL_VALUE_ROW 13.Special_Code = t5.Special_C NNTROL_VALUE_ROW 13.Special_Number_Nat NNTROL_VALUE_ROW 13.Special_Number 10127911 10173341 10173341 10173341 10173341 10173691 11471884 11573691 1147184 11573691	CUERCK IN COUNT Why COUNT WHY	Image: A constraints of the second	eise NOT_A D else end D else else else D else end D else end D else else D else else else D else else D else else D else else else else D else else else else D else else else else D else else else else else else else els	OK end It is in the interval of the interval	CH'end Sented nples al AOK	by Cor long w d Spec Cod Match NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC
e N un w S	Alock Row Ans which pecial_Co person_ Person_ Person_ 0 0 0 0 0 0 0 0 0 0 0 0 0	Load Specia Specia Specia Specia Specia Special Arecial Special Code_	Date_Flag al_Code_Flag al_Code_Mal_Number al_Number al_Number al_Number al_Person_ e der Matc C C C C C C C C C C C C C	ag latch _Rag _Perce Flag ive th C b code _ lag 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Calcu d in Data Data Calcu	se t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC the same puted Col Smiley Sof 0 Smiley's Sof 0 Smiley's Sof 0 Smiley's Sof 0 Smiley's Dat 0 Smiley's Sof 0 Smiley's Sof	Late ge 14.Load_Late, Late ge 14.Load_Late, NUTROL_VALUE_ROW NUTROL_VALUE_ROW NUTROL_VALUE_ROW NUTROL_VALUE_ROW Ial Code = 15.Special_0 DNTROL_VALUE_ROW NUTROL_VALUE_ROW NUTROL_VALUE_ROW Validation, a e way as shot umn repress 10127911 10173341 1017341 1017358 10173600 e 1037600 e 1037600 e 10377600 e 10377600 a 10376601 e 10377602 a 1207184 a 1207184 a 1207184 a 13173691 a 13775492 a 1547184 a 1547184 a 16074333 ughter 16176964	COUNT why COUNT why COUNT why de_National_ COUNT why and Filt Down in Senting Special A A A B A Code A C C D A C D A C D A C D A C D A C D A C D A C D A C D A C C C C C C C C C C C C C	Tratic focus the en 0 then i focus the en 0 then i focus the en 0 then i focus the focus the en 0 then focus the focus the foc	eise NOT_A 0 else . end 0 el	OK end I CEDTC: Special Special Number Percent 84.4% 84.8% 84.5% 95.6% 95.6% 95.6% 100.6% 100.6% 100.6% 114.8% 114.8% 114.8% 114.8% 114.8% 128.9% 134.8% 134.8% 134.8%	CH'end Sented nples al AOK	by Cori long w d Specific Cod Mate NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC NO MATC
	Aock Row ans which pecial_Co Person_ > N Person_ > N 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Load_ Specia Specia Specia Specia Specia Specia Prove Cr 1 ar ode_ Special Special	Date_Flag al_Code_M al_Number, al_Number, al_Number, al_Number, al_Person_ e der Matc C C C C C C C C C C C C C	ag ag Flag Perce Flag n, (ive h C bode bode bode code	ca ca ca ca ca ca ca ca ca ca ca ca ca c	se t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC see t2.SMILEY_CC .Special_Number/ see t2.SMILEY_CC .Special_Number/ see t2.SMILEY_CC .Special_Number/ the same puted Col .Smiley_CCC 0 Smiley's Son 0 Smiley's Twi 0 Smiley's Twi 0 Smiley's Twi 0 Smiley's Son 0 Smiley's Son	_Date get 4.Load_Date _Date get 4.Load_Date _Date get 4.Load_Date _NITROL_VALUE_ROW _NITROL_VALUE_ROW _ial_code = t5.Special_C _NITROL_VALUE_ROW /Aalidation, a e way as shot	COUNT why COUNT why COUNT why de_National_ COUNT why and Filt own in senting Special Code A A K B A Code J D A A E C C H B C C C C C C C C C C C C C	Image of the set of	else NOT_A 0 else end 0 else	OK end Ise NO MAT Special S	CH'end Sented nples al CH'end	by Cor long w

Here is the final result with the Flags to the left and the Calculation, Validation, and Filtration Computed Columns to the right of each of the 20 rows and 4 columns.

(Smile)	Control Va / Compar	alue Rep ny All Jo	ool	rt for less Joins	5		
	The FRE	Q Proce	dur	e			
Special_Person_Flag	J Frequenc	y Perce	nt	Cumulative Cumulative Frequency Perce			
() 2	20 100.	00	2	0 100.00		
Special_Number_Fla	g Frequen	cy Perce	ent	Cumulativ Frequence	ve Cumulativ cy Percen		
	0	20 100	00	2	20 100.00		
Special_Code_Flag	Frequency	Percer	nt	Cumulative Frequency	Cumulative Percent		
0	20	0 100.0	0	20	20 100.00		
Load_Date_Flag	Frequency	Percent	C	Cumulative Cumulativ Frequency Perce			
0	20	100 00		20	100 00		

The One-Way Frequency correctly displays that all 4 flags are set to 0 and therefore no data is missing ~ thanks to the Joinless Join ⁽²⁾.



The Cartesian product of 4 Tables with 1 Row and 1 Column in each Table 👻
🛺 Input Data (4) 🛄 Code 📋 Log 📳 Output Data
🝤 🔍 Modify Task 🐺 Filter and Sort 🖽 Query Builder Data 🗸 Describe 🗸 Graph 🗸 Analyze 🗸 Export 🗸 Send To 🗸 🗄
😡 SMILEY_CONTROL_VALUE_ROW_COUNT 🔕 Special_Number_National_Average 📰 Load_Date_Check 💩 Special_Code_National_Focus
1 0 1200000 01JAN2015 K
column of the SMILEY_CONTROL_VALUE_ROW_COUNT, SPECIAL_NUMBER_NATIONAL _AVERAGE, LOAD_DATE_CHECK, and SPECIAL_CODE_NATIONAL_FOCUS tables to the right of each other.
🐺 Filter and Sort 🖳 Query Builder Data 🔹 Describe 👻 Graph 👻 Analyze 👻 Export 👻 Send To 👻 🧮
😡 SMILEY_CONTROL_VALUE_ROW_COUNT 🕲 Special_Number_National_Average 🧮 Load_Date_Check 💩 Special_Code_National_Focus
1 0 1200000 01.IAN2015 K

• Here is the final result from selecting all 4 columns which is equal to the Cartesian Product.



The Joinless Join is based upon all 4 columns in the JOINLESS_JOIN_NOTHING_IN_COMMON table which indirectly relate to the SMILEY_COMPANY table as shown in the previous examples.

\$5	民 Modify Task 🐺	Filter and Sor	t 💾 Query B	uilder Data + Des	cribe 🕶 Graph 👻 Analyze 👻	Export +	Send To 👻	a a	
	🔌 Special_Person	Special_ Number	à Special_ Code	🖪 Load_Date 😡	SMILEY_CONTROL_VALUE_ ROW_COUNT	📵 Spe Nati	cial_Number_ ional_Average	Load_Date_ Check	Special_Code_ National_Focus
1	Smiley	10127911	A	02JAN2015	0		1200000	01JAN2015	к
2	Smiley's Son	10173341	ĸ	02JAN2015	0		1200000	01JAN2015	К
3	Smiley's Twin	10376606	В	02JAN2015	0		1200000	01JAN2015	К
4	Smiley's Wife	10927911	A	02JAN2015	0		1200000	01JAN2015	K
5	Smiley's Son	11471884	E	02JAN2015	0		1200000	01JAN2015	K
6	Smiley's Twin	11573691	G	02JAN2015	0		1200000	01JAN2015	К
7	Smiley's Daughter	11975386	С	02JAN2015	0		1200000	01JAN2015	К
8	Smiley's Son	12071884	J	02JAN2015	0		1200000	01JAN2015	К
9	Smiley's Son	12871884	D	02JAN2015	0		1200000	01JAN2015	К
10	Smiley's Twin	13173691	A	02JAN2015	0		1200000	01JAN2015	K
11	Smiley's Wife	13771202	D	02JAN2015	0		1200000	01JAN2015	K
12	Smiley's Daughter	13775498	Η	02JAN2015	0		1200000	01JAN2015	K
13	Smiley's Son	14171884	I	02JAN2015	0		1200000	01JAN2015	K
14	Smiley's Twin	15373691	F	02JAN2015	0		1200000	01JAN2015	K
15	Smiley's Son	15471884	С	02JAN2015	0		1200000	01JAN2015	К
16	Smiley's Son	16074330	Η	02JAN2015	0		1200000	01JAN2015	К
17	Smiley's Daughter	16175498	В	02JAN2015	0		1200000	01JAN2015	K
18	Smiley's Wife	16176964	I	31DEC2014	0		1200000	01JAN2015	K
19	Smiley	16279111	E	02JAN2015	0		1200000	01JAN2015	K
20	Smiley's Twin	16573691	K	02JAN2015	0		1200000	01JAN2015	K

columns of the SMILEY_CONTROL_VALUE_ROW_COUNT, SPECIAL_NUMBER_NATIONAL _AVERAGE, LOAD_DATE_CHECK, and SPECIAL_CODE_NATIONAL_FOCUS tables to the right of each of the 20 rows and 4 columns in the SMILEY_COMPANY table.

С	omputed Columns	
	Column Load_Date_Validation Load_Date_Rag Special_Code_Rag Special_Code_Match Special_Number_Rag	Details Case when 11.Load_Date ge 12.Load_Date_Check then 'AOK' else 'NOT_AOK' end Case 12.SMILEY_CONTROL_VALUE_ROW_COUNT when 0 then 0 else . end Case the:SMILEY_CONTROL_VALUE_ROW_COUNT when 0 then 0 else . end Case the:SMILEY_CONTROL_VALUE_ROW_COUNT when 0 then 0 else . end Case t2.SMILEY_CONTROL_VALUE_ROW_COUNT when 0 then 0 else . end Case t2.SMILEY_CONTROL_VALUE_ROW_COUNT when 0 then 0 else . end
	Special_Number_Percent Special_Person_Flag	t1.Special_Number/t2.Special_Number_National_Average case t2.SMILEY_CONTROL_VALUE_ROW_COUNT when 0 then 0 else . end

The Mock Row Creation, Calculation, Validation, and Filtration are represented by Computed Columns which are derived in the same way as shown in the previous examples along with one new Special_Code_Match Computed Column representing Filtration.

	Special_ Person_ Flag	Special_ Number_ Flag	Special _Code_ Flag	Load Date _Flag	🔌 Special_Person	Special_ Number	Special_ Code	Load_Date	Special_ Number_ Percent	▲ Date_Valid ation	Special_Code _Match
1	0	0	0	0	Smiley	10127911	А	02JAN2015	84.4%	AOK	NO MATCH
2	0	0	0	0	Smiley's Son	10173341	К	02JAN2015	84.8%	AOK	MATCH
3	0	0	0	0	Smiley's Twin	10376606	В	02JAN2015	86.5%	AOK	NO MATCH
4	0	0	0	0	Smiley's Wife	10927911	A	02JAN2015	91.1%	AOK	NO MATCH
5	0	0	0	0	Smiley's Son	11471884	E	02JAN2015	95.6%	AOK	NO MATCH
6	0	0	0	0	Smiley's Twin	11573691	G	02JAN2015	96.4%	AOK	NO MATCH
7	0	0	0	0	Smiley's Daughter	11975386	С	02JAN2015	99.8%	AOK	NO MATCH
8	0	0	0	0	Smiley's Son	12071884	J	02JAN2015	100.6%	AOK	NO MATCH
9	0	0	0	0	Smiley's Son	12871884	D	02JAN2015	107.3%	AOK	NO MATCH
10	0	0	0	0	Smiley's Twin	13173691	A	02JAN2015	109.8%	AOK	NO MATCH
11	0	0	0	0	Smiley's Wife	13771202	D	02JAN2015	114.8%	AOK	NO MATCH
12	0	0	0	0	Smiley's Daughter	13775498	Н	02JAN2015	114.8%	AOK	NO MATCH
13	0	0	0	0	Smiley's Son	14171884	I	02JAN2015	118.1%	AOK	NO MATCH
14	0	0	0	0	Smiley's Twin	15373691	F	02JAN2015	128.1%	AOK	NO MATCH
15	0	0	0	0	Smiley's Son	15471884	С	02JAN2015	128.9%	AOK	NO MATCH
16	0	0	0	0	Smiley's Son	16074330	Н	02JAN2015	134.0%	AOK	NO MATCH
17	0	0	0	0	Smiley's Daughter	16175498	В	02JAN2015	134.8%	AOK	NO MATCH
18	0	0	0	0	Smiley's Wife	16176964	I	31DEC2014	134.8%	NOT_AOK	NO MATCH
19	0	0	0	0	Smiley	16279111	E	02JAN2015	135.7%	AOK	NO MATCH
20	0	0	0	0	Smiley's Twin	16573691	К	02JAN2015	138.1%	AOK	MATCH

Here is the final result with the Flags to the left and the Calculation, Validation, and Filtration Computed Columns to the right of each of the 20 rows and 4 columns.

Smiley C	Control V Company /	alue Rep All Joinle	ort for ss Joins A	gain
	The FR	EQ Proced	ure	
Special_Person_Fla	ig Frequen	cy Perce	Cumulativ t Frequence	ve Cumulative cy Percent
	0	20 100.0	0 2	20 100.00
Special_Number_Fla	ag Frequen	cy Perce	Cumulation Cumulation	ve Cumulative cy Percen
	0 20 100.00 20 100		20 100.0	
Special_Code_Flag	g Frequenc	y Percen	Cumulative Frequency	e Cumulative Percent
() 2	0 100.00	20	0 100.00
Load_Date_Flag	Frequency	Percent	Cumulative Frequency	Cumulative Percent
0	20	100.00	20	100.00



CONCLUSION

The **Joinless Join** empowers you to creatively overcome the limits of a standard Join or Merge and enables you to expand the power of Base SAS and SAS Enterprise Guide in a new way. **The Power To Know** how to design a Joinless Join sets off **The Power To Create** tables based upon dependencies, indirect relationships, or no relationships at all which leads to **The Power To Automate** projects even when tables cannot be directly joined or merged **~** ⁽²⁾ try saying that statement really fast for fun ⁽²⁾!

The Joinless Join bridges the research impasse you experience when needing to combine data from tables which do not contain like columns or the same variable name. New worlds of table creations, calculations, validations, filtrations, and PROC REPORTing have opened up to greatly expand your data transformation and analysis toolkit. Begin thinking about how you can benefit from the power and versatility of the Joinless Join.



How wonderful it is that we need not wait a single minute before starting to improve ourselves and our world! Anne Frank

SAS Programming is like a series of intricate and fluid domino designs and you are the **Designer**. Your desire to design a quality program fuels your thoroughness and attention to detail. As a SAS Professional, your inquisitive nature, research oriented mindset, and solution driven focus are among your greatest assets.



Your life is like a campfire at night -You never know how many people will see it and be comforted and guided by your light. Claire Draper

Rule #6: Study hard and learn all you can. © Roy Rogers Riders Club Rules ©



Always remember – *It's not what the SAS World holds for you, it's what YOU bring to it! Continue to develop and build on your many skills and talents. Keep looking for different ways to share your God-given abilities and ideas. You will soon discover new and creative ways to design your SAS programs. Plan on coming back to the MWSUG Conference next year to shed some light on the exciting things you are learning. All of us are on the SAS journey with you and we look forward to your teaching sessions in the future.*

As we conclude, we want to introduce you to our **SAS Mascot, Smiley**. Smiley represents the **SAS Joy** which each of us experience as we find better ways to accomplish mighty and worthy deeds using SAS. The three of us, along with Professor Domino, hope we have expanded and enriched your SAS knowledge.

Thank You for sharing part of your SAS journey with us ~ ③ Happy SAS Trails to you... until we meet again ③





Writing is a permanent legacy.

John C. Maxwell

Kent Phelps ~ *SAS Certified Professional* ~ *B.S. Electrical Engineering* ~ *Writer* ~ *Teacher* ~ *Coach* ~ has presented at the MWSUG Conference for 3 years, worked in IT and Data Governance since 1990, programmed in SAS since 2007, and specializes in blending the best of Base SAS with SAS Enterprise Guide to engineer automated solutions. He co-created/taught *Intro to SAS EG* classes, offered *SAS News You Can Use*, presented at the Iowa SAS Users Group (IASUG), studied Transformational Leadership, Dynamic Teamwork, and Personal Growth since 1994, and is certified as a *John Maxwell Team* and *48 Days To The Work You Love* Coach. Past highlights include acting for over ten years, co-leading *WOW Drama*, singing a drama solo with a live orchestra, and auditioning in Branson, MO. Kent wants to encourage and equip you to fulfill your life and leadership potential as you build an enduring legacy of inspiration, excellence, and honor.

Ronda Phelps ~ *Writer* ~ *Teacher* ~ *Coach* ~ has presented at the MWSUG Conference for 2 years, formerly worked in the Banking and Insurance industries for 19 years, studied Transformational Leadership, Dynamic Teamwork, and Personal Growth since 1994, and is certified as a *John Maxwell Team* and *48 Days To The Work You Love* Coach. Past highlights include speaking in Siberia, acting for over ten years, co-leading *WOW Drama*, and developing life-changing presentations. Ronda believes that YOU are a gift the world is waiting to receive, and she wants to encourage and equip you to pursue your unique destiny as you navigate your life journey with intentionality, fulfilling purpose, and enduring hope.

We invite you to share your valued comments with us:

Kent ♥ Ronda Team Phelps The SASketeers ~ All for SAS & SAS for All! E-mail: <u>SASketeers@q.com</u>

🙂 We look forward to connecting with you in the future! 🙂





New Computed Column	
3 of 4 Modify ad	litional options
Identifier: Special_P	rson_Flag
Column Name: Special_P	rson_Rag
Label: Summary: NONE	
Expression: CASE	
THE ELSE O FAIL	1.specia_Person =
ENU	
Format:	
	k V Nexts 5
• Entor the New Computed Column as the Identif	ior and Column Name and sligh Next
• Enter the New Computed Column as the Identif	er and column Name and ChCK Next.
New Computed Column	
4 of 4 Summary of properties	
Identifier: Special_Person_Flag Column Name: Special_Person_Flag	
Label: Default Format: Default Length: Default	
Summary: None Expression:	
CASE WHEN t1.Special_Person = " THEN 1	
ELSE 0 END Comput	ed Columns X
Colum	Details New
qد 🗮	ccial_Person CASE WHEN t1.Special_Person = "
	Delete
	Rename
<back next="" v=""> Finish</back>	Close
Click Finish and then click Close to close the Co	mputed Column.
	Output name: WORK SMILEY CONTROL VALUE
🖬 Computed Columns 🌒 Prompt Manager	review 🕼 Tools 🗸 😰 Options 👻
H Add Tables X Delete Join Tables Select	Data Filter Data Sort Data
Actin (SMILEY-COMPANY) Colum Aspecial Person Secial Number	n Name Identifier ecial_Person t1.Special_Person
Special_Code	ecal_Number t1.Special_Number ecial_Code t1.Special_Code ad_Date t11.and_Date
□ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □	ecial_Person_Flag Special_Person_Flag
The Special_Person_Flag now appears under Co	mputed Columns and in the Selected Data.
Repeat this process to create the 3 additional C	omputed Columns that are needed.

APPENDIX B How To Code a Joinless Join Using Base SAS

The following Base SAS Code was generated by SAS Enterprise Guide for all examples and creates the same results when copied to and run in Base SAS.



- The PROC SQL creates a TABLE called SMILEY_CONTROL_VALUE by assigning the value 0 (present) or 1 (missing) to Special_Person_Flag, Special_Number_Flag, Special_Code_Flag, and Load_Date_Flag and selecting Special_Person, Special_Number, Special_Code, and Load_Date from the SMILEY COMPANY table.
- ***** Each CASE statement ends with AS and a variable name because the result of each CASE statement is stored in a flag variable.
- The WHERE clause contains the word CALCULATED before each variable name because these variables are calculated rather than selected from the table while also limiting the output data set to contain only rows in which 1 or more of the calculated variables are missing (= 1).

This code creates the Control Value Report for the Smiley Company One-Way Frequency:

```
PROC SQL;
    CREATE VIEW WORK.SORT AS
           SELECT T.Special Person Flag, T.Special Number Flag,
                   T.Special_Code_Flag, T.Load_Date_Flag
    FROM WORK.SMILEY CONTROL VALUE AS T;
QUIT;
TITLE;
TITLE1 "Control Value Report for";
TITLE2 "Smiley Company";
FOOTNOTE ;
FOOTNOTE1 "Generated by the SAS System on %TRIM(%QSYSFUNC(DATE(),
           NLDATE20.)) at %TRIM(%SYSFUNC(TIME(), TIMEAMNP12.))";
PROC FREQ DATA=WORK.SORT
       ORDER=INTERNAL;
       TABLES Special Person Flag / SCORES=TABLE;
       TABLES Special_Number_Flag / SCORES=TABLE;
TABLES Special_Code_Flag / SCORES=TABLE;
       TABLES Load Date Flag / SCORES=TABLE;
RUN;
RUN; QUIT;
TITLE; FOOTNOTE;
```

- The **proc** sol creates a **view** from the SMILEY_CONTROL_VALUE table containing only the variables which are to be included in the One-Way Frequency.
- ***** The **TITLE** and **FOOTNOTE** statements with no title or footnote clear all titles and footnotes, and the **TITLE1**, **TITLE2**, and **FOOTNOTE1** statements set the titles and footnote.
- **FOOTNOTE1** is an optional default which is always added by SAS Enterprise Guide.
- The PROC FREQ creates TABLES for each Flag containing the SCORES or values of each Flag listed by INTERNAL or numeric/alphabetic ORDER.
- ***** Both order=internal and scores=table are optional defaults for proc freq which are always added by SAS Enterprise Guide.

This code creates the SMILEY_CONTROL_VALUE_ROW_COUNT table:

```
PROC SQL;
```

```
CREATE TABLE WORK.SMILEY_CONTROL_VALUE_ROW_COUNT AS
SELECT /* SMILEY_CONTROL_VALUE_ROW_COUNT */
(COUNT(t1.Special_Person)) AS SMILEY_COUNTROL_VALUE_ROW_COUNT
FROM WORK.SMILEY_CONTROL_VALUE t1;
QUIT;
```

The proc sol creates a TABLE called SMILEY_CONTROL_VALUE_ROW_COUNT from the SMILEY_CONTROL_VALUE table containing the COUNT of the values of the Special_Person variable stored in the SMILEY_COUNTROL_VALUE_ROW_COUNT variable.



- The proc sql creates a TABLE called SMILEY_CONTROL_VALUE_MOCK_ROW containing one observation (OUTOBS=1).
- The value 0 is assigned to Special_Person_Flag, Special_Number_Flag, Special_Code_Flag, and Load_Date_Flag, and Special_Person, Special_Number, Special_Code, and Load_Date are selected from the SMILEY_COMPANY table.
- Notice the **FROM** does not contain any type of join between the 2 tables thus a Joinless Join.
- The where clause causes the output row to be created with all 4 flags set to 0 only when the value of the SMILEY CONTROL VALUE ROW COUNT = 0 and therefore is a 'Mock Row'.

```
This code creates the Append of the Smiley_Control_Value
        table and the Smiley_Control_Value_Mock_Row table:
PROC SQL;
    CREATE TABLE WORK. SMILEY CONTROL VALUE FINAL AS
    SELECT * FROM WORK. SMILEY CONTROL VALUE
        OUTER UNION CORR
    SELECT * FROM WORK. SMILEY CONTROL VALUE MOCK ROW;
QUIT;
OR
DATA WORK. SMILEY CONTROL VALUE FINAL;
    SET WORK. SMILEY CONTROL VALUE
        WORK.SMILEY CONTROL VALUE MOCK ROW;
RUN:
The PROC SQL creates a TABLE called SMILEY CONTROL VALUE FINAL by concatenating the
results (OUTER UNION) of all columns (SELECT *) from the SMILEY CONTROL VALUE and the
SMILEY CONTROL VALUE MOCK ROW tables and overlaying all corresponding (CORR) columns.
```

- The DATA step creates a data set (or table) called SMILEY_CONTROL_VALUE_FINAL by SETING the SMILEY_CONTROL_VALUE data set and the SMILEY_CONTROL_VALUE_MOCK_ROW data set with all columns (which are the same in this case) from both data sets.
- The **proc** sol and the **data** step create the same results, thus either can be used.

This code creates the Final Control Value Report for the Smiley Company One-Way Frequency:

```
PROC SQL;
    CREATE VIEW WORK.SORT AS
           SELECT T.Special Person Flag, T.Special Number Flag,
                  T.Special Code Flag, T.Load Date Flag
   FROM WORK. SMILEY CONTROL VALUE FINAL AS T;
QUIT;
TITLE ;
TITLE1 "Final Control Value Report for";
TITLE2 "Smiley Company";
FOOTNOTE ;
FOOTNOTE1 "Generated by the SAS System on %TRIM(%QSYSFUNC(DATE(),
          NLDATE20.)) at %TRIM(%SYSFUNC(TIME(), TIMEAMNP12.))";
PROC FREQ DATA=WORK.SORT
       ORDER=INTERNAL;
       TABLES Special Person Flag / SCORES=TABLE;
      TABLES Special_Number_Flag / SCORES=TABLE;
      TABLES Special_Code_Flag / SCORES=TABLE;
      TABLES Load Date Flag / SCORES=TABLE;
RUN;
RUN; QUIT;
TITLE; FOOTNOTE;
```

- The proc sol creates a view from the SMILEY_CONTROL_VALUE_FINAL table containing only the variables which are to be included in the One-Way Frequency.
- ***** The **TITLE** and **FOOTNOTE** statements with no title or footnote clear all titles and footnotes, and the **TITLE1**, **TITLE2**, and **FOOTNOTE1** statements set the titles and footnote.
- **FOOTNOTE1** is an optional default which is always added by SAS Enterprise Guide.
- The PROC FREQ creates TABLES for each Flag containing the SCORES or values of each Flag listed by INTERNAL or numeric/alphabetic ORDER.
- ***** Both order=internal and scores=table are optional defaults for proc freq which are always added by SAS Enterprise Guide.

This code creates the SMILEY_JOINLESS_JOIN_CALCULATION table:

- The PROC SQL creates a TABLE called SMILEY_JOINLESS_JOIN_CALCULATION by selecting Special_Person, Special_Number, Special_Code, and Load_Date from the SMILEY_COMPANY table.
- The Special_Number_Percent column is calculated by taking the ratio of Special_Number from the SMILEY_COMPANY table and Special_Number_National_Average from the SPECIAL_NUMBER_NATIONAL_AVERAGE table and applying the FORMAT=PERCENT8.1 to obtain the resulting percent instead of the ratio.
- **Notice the FROM does not contain any type of join between the 2 tables and thus is a Joinless** Join.

This code creates the SMILEY_JOINLESS_JOIN_VALIDATION table:

- The PROC SQL creates a TABLE called SMILEY_JOINLESS_JOIN_VALIDATION by selecting Special_Person, Special_Number, Special_Code, and Load_Date from the SMILEY_COMPANY table.
- The Date_Validation column is derived by checking if Load_Date from the SMILEY_COMPANY table is greater than or equal to (GE) Load_Date_Check from the LOAD_DATE_CHECK table and assigning 'AOK' Or 'NOT AOK' as a result.
- Notice the FROM does not contain any type of join between the 2 tables and thus is a Joinless Join.

This code creates the SMILEY_JOINLESS_JOIN_FILTRATION table:

- The PROC SQL creates a TABLE called SMILEY_JOINLESS_JOIN_CALCULATION by selecting Special_Person, Special_Number, Special_Code, and Load_Date from the SMILEY_COMPANY table.
- The where clause filters the output to include only observations in which t1.Special_Code from the SMILEY_COMPANY table is equal to Special_Code_National_Focus from the SPECIAL_CODE_NATIONAL_FOCUS table.
- Notice the FROM does not contain any type of join between the 2 tables and thus is a Joinless Join.

This code creates the SMILEY_JOINLESS_JOIN_ALL_CHECKS table:



- The PROC SQL creates a TABLE called SMILEY_JOINLES_JOIN_ALL_CHECKS by selecting Special_Person, Special_Number, Special_Code, and Load_Date from the SMILEY_COMPANY table.
- If SMILEY_CONTROL_VALUE_ROW_COUNT is 0 then a mock row needs to be created with Special_Person_Flag, Special_Number_Flag, Special_Code_Flag, and Load_Date_Flag assigned a value 0; otherwise a mock row is not needed and the flags are set to Null (.).
- The Special_Number_Percent, Date_Validation, and Special_Code_Match columns are calculated or derived as in the previous examples; however Special_Code_Match is a derived column rather than an applied filter.
- Notice the FROM does not contain any type of join between the 5 tables and thus is a Joinless Join.

This code creates the Append of the Smiley_Control_Value table and the Smiley_Joinless_Join_All_Checks table:

```
PROC SQL;
CREATE TABLE WORK.SMILEY_CONTROL_VALUE_FINAL_ALL AS
SELECT * FROM WORK.SMILEY_CONTROL_VALUE
OUTER UNION CORR
SELECT * FROM WORK.SMILEY_JOINLESS_JOIN_ALL_CHECKS;
QUIT;
OR
DATA WORK.SMILEY_CONTROL_VALUE_FINAL_ALL;
SET WORK.SMILEY_CONTROL_VALUE
WORK.SMILEY_JOINLESS_JOIN_ALL_CHECKS;
RUN;
```

- The PROC SQL creates a TABLE called SMILEY_CONTROL_VALUE_FINAL_ALL by concatenating the results (OUTER UNION) of all columns (SELECT *) from the SMILEY_CONTROL_VALUE and the SMILEY_JOINLESS_JOIN_ALL_CHECKS tables and overlaying all corresponding (CORR) columns.
- The DATA step creates a data set (or table) called SMILEY_CONTROL_VALUE_FINAL_ALL by SETING the SMILEY_CONTROL_VALUE data set and the SMILEY_JOINLESS_JOIN_ALL_CHECKS data set with all columns (which are the same in this case) from both data sets.
- ***** The **proc** sql and the **data** step create the same results, thus either can be used.

This code creates the Control Value Report for the Smiley Company All Joinless Joins One-Way Frequency:

```
PROC SOL;
     CREATE VIEW WORK.SORT AS
             SELECT T.Special_Person_Flag, T.Special_Number_Flag,
                      T.Special_Code_Flag, T.Load_Date_Flag
     FROM WORK.SMILEY CONTROL VALUE FINAL ALL AS T;
QUIT;
TITLE :
TITLE1 "Control Value Report for";
TITLE2 "Smiley Company All Joinless Joins";
FOOTNOTE ;
FOOTNOTE1 "Generated by the SAS System on %TRIM(%QSYSFUNC(DATE(),
             NLDATE20.)) at %TRIM(%SYSFUNC(TIME(), TIMEAMNP12.))";
PROC FREQ DATA=WORK.SORT
        ORDER=INTERNAL;
        TABLES Special_Person_Flag /SCORES=TABLE;TABLES Special_Number_Flag /SCORES=TABLE;TABLES Special_Code_Flag /SCORES=TABLE;TABLES Load_Date_Flag /SCORES=TABLE;
RUN;
RUN; QUIT;
TITLE; FOOTNOTE;
```

- The proc sql creates a view from the SMILEY_CONTROL_VALUE_FINAL_ALL table containing only the variables which are to be included in the One-Way Frequency.
- **The TITLE and FOOTNOTE statements with no title or footnote clear all titles and footnotes, and the TITLE1, TITLE2, and FOOTNOTE1 statements set the titles and footnote.**
- **FOOTNOTE1** is an optional default which is always added by SAS Enterprise Guide.
- The PROC FREQ creates TABLES for each Flag containing the SCORES or values of each Flag listed by INTERNAL or numeric/alphabetic ORDER.
- **Both Order=internal and scores=table are optional defaults for proc freq which are always added by SAS Enterprise Guide.**

This code creates the JOINLESS_JOIN_NOTHING_IN_COMMON table:

PROC SQL;
CREATE TABLE WORK. JOINLESS JOIN NOTHING IN COMMON AS
SELECT t1.SMILEY CONTROL VALUE ROW COUNT,
t2.Special Number National Average,
t3.Load_Date_Check,
t4.Special_Code_National_Focus,
FROM WORK.SMILEY_CONTROL_VALUE_ROW_COUNT t1,
WORK.SPECIAL_NUMBER_NATIONAL_AVERAGE t2, WORK.LOAD_DATE_CHECK t3,
WORK.SPECIAL_CODE_NATIONAL_FOCUS t4;
QUIT;

The PROC SQL creates a TABLE called JOINLESS_JOIN_NOTHING_IN_COMMON by selecting SMILEY_CONTROL_VALUE_ROW_COUNT from the SMILEY_CONTROL_VALUE_ROW_COUNT table, Special_Number_National_Average from the SPECIAL_NUMBER_NATIONAL_AVERAGE table, Load_Date_Check from the LOAD_DATE_CHECK table, and Special_Code_National_Focus from the SMILEY_COMPANY table.

Notice the FROM does not contain any type of join between the 4 tables and thus is a Joinless Join.

This code creates the SMILEY_JOINLESS_JOIN_ALL_AGAIN table:



- The PROC SQL creates a TABLE called SMILEY_JOINLES_JOIN_ALL_CHECKS by selecting Special_Person, Special_Number, Special_Code, and Load_Date from the SMILEY_COMPANY table.
- If SMILEY_CONTROL_VALUE_ROW_COUNT is 0 then a mock row is needs to be created with Special_Person_Flag, Special_Number_Flag, Special_Code_Flag, and Load_Date_Flag assigned a value 0; otherwise a mock row is not needed and the flags are set to Null (.).
- The Special_Number_Percent, Date_Validation, and Special_Code_Match columns are calculated or derived as in the previous examples; however Special_Code_Match is a derived column rather than an applied filter.
- Notice the FROM does not contain any type of join between the 2 tables and thus is a Joinless Join.

This code creates the Control Value Report for the Smiley Company All Joinless Joins Again One-Way Frequency:

```
PROC SQL;
    CREATE VIEW WORK.SORT AS
           SELECT T.Special Person Flag, T.Special Number Flag,
                  T.Special Code Flag, T.Load Date Flag
    FROM WORK.SMILEY JOINLESS JOIN ALL AGAIN AS T;
QUIT;
TITLE;
TITLE1 "Control Value Report for";
TITLE2 "Smiley Company All Joinless Joins Again";
FOOTNOTE ;
FOOTNOTE1 "Generated by the SAS System on %TRIM(%QSYSFUNC(DATE(),
          NLDATE20.)) at %TRIM(%SYSFUNC(TIME(), TIMEAMNP12.))";
PROC FREQ DATA=WORK.SORT
      ORDER=INTERNAL;
      TABLES Special_Person_Flag / SCORES=TABLE;
      TABLES Special_Number_Flag / SCORES=TABLE;
      TABLES Special Code Flag / SCORES=TABLE;
                                / SCORES=TABLE;
      TABLES Load Date Flag
RUN;
RUN; QUIT;
TITLE; FOOTNOTE;
```

- The PROC SQL creates a VIEW from the SMILEY_JOINLESS_JOIN_ALL_AGAIN table containing only the variables which are to be included in the One-Way Frequency.
- ***** The **TITLE** and **FOOTNOTE** statements with no title or footnote clear all titles and footnotes, and the **TITLE1**, **TITLE2**, and **FOOTNOTE1** statements set the titles and footnote.
- **FOOTNOTE1** is an optional default which is always added by SAS Enterprise Guide.
- The PROC FREQ creates TABLES for each Flag containing the SCORES or values of each Flag listed by INTERNAL or numeric/alphabetic ORDER.
- ***** Both order=internal and scores=table are optional defaults for proc freq which are always added by SAS Enterprise Guide.

ACKNOWLEDGMENTS

We want to thank the 27th Annual MWSUG 2016 Hands-On Workshop Section Co-Chairs, **Dave Foster** and **Chuck Kincaid**, for graciously accepting our abstract and paper. In addition, we want to express our appreciation to the Conference Co-Chairs, **Richann Watson** (Academic Chair) and **Adrian Katschke** (Operations Chair), the Executive Committee and Conference Leaders, and SAS Institute for their diligent efforts in organizing this illuminating and energizing conference.

We also offer our deep gratitude to our friend, mentor, and fellow SASketeer, **Kirk Paul Lafler**. Your heart to continuously share what you are learning, blended with your servant leadership and supportive guidance, is a constant light of encouragement to us. And in conclusion, we want to give a shout out to our friend, **Charlie Shipp**, for his friendship and faithful service to the SAS World. You both inspire us to share what we are learning and our hope is to be a light of encouragement to you as well ~ *All for SAS & SAS for All!*



Carpenter, Art (2012), *PROC REPORT Basics: Getting Started with the Primary Statements*, Proceedings of the 6th Annual SAS Global Forum (SGF) 2012 Conference, California Occidental Consultants, Anchorage, AK, USA. http://support.sas.com/resources/papers/proceedings12/242-2012.pdf

Celko, Joe (2010), *Joe Celko's SQL for Smarties, Fourth Edition: Advanced SQL Programming (The Morgan Kaufmann Series in Data Management Systems)*; November 10, 2010; ISBN-10: 0123820227; ISBN-13: 978-0123820228. http://www.accuteach.com/book/joe-celkos-sql-for-smarties-fourth-edition-advanced-sql-programming-the-morgan-kaufmann-series-in-data-management-systems-by-joe-celko/#

Foley, Malachy J. (2005), *Merging vs. Joining: Comparing the DATA Step with SQL*, Proceedings of the 30th Annual SAS Users Group International (SUGI) 2005 Conference, University of North Carolina, Chapel Hill, NC, USA. http://www.scsug.org/SCSUGProceedings/2005/Foley_Merging%20vs%20Joining%20-%20184.pdf

Kent, Paul, *SQL Joins -- The Long and The Short of It*; SAS Institute Inc., Cary, NC, USA. http://support.sas.com/techsup/technote/ts553.html

Kerman, Jonathan (2010), Ordering PROC FREQ Around, NorthEast SAS Users Group (NESUG) 2010 Conference, Johns Hopkins University, S Baltimore, MD, USA. http://www.lexiansen.com/nesug/nesug10/cc/cc16.pdf

Lafler, Kirk Paul (2013), *PROC SQL: Beyond the Basics Using SAS, Second Edition*; SAS Press. http://support.sas.com/publishing/authors/lafler.html

Lafler, Kirk Paul and Mira Shapiro (2013), *Point-and-Click Programming Using SAS® Enterprise Guide®*, NorthEast SAS Users Group (NESUG) 2013 Conference, Software Intelligence Corporation, Spring Valley, CA, USA. http://www.lexiansen.com/nesug/nesug13/63 Final Paper.pdf

Lafler, Kirk Paul (2012), *Exploring DATA Step Merges and PROC SQL Joins*, Proceedings of the 6th Annual SAS Global Forum (SGF) 2012 Conference, Software Intelligence Corporation, Spring Valley, CA, USA. http://support.sas.com/resources/papers/proceedings12/251-2012.pdf

Lafler, Kirk Paul (2012), *Exploring DATA Step Merges and PROC SQL Joins*, Proceedings of the 14th Annual Pharmaceutical SAS Users Group (PharmaSUG) 2012 Conference, Software Intelligence Corporation, Spring Valley, CA, USA. http://pharmasug.org/proceedings/2012/TA/PharmaSUG-2012-TA02.pdf

Lafler, Kirk Paul (2011), *Output Delivery System (ODS)– Simply the Basics*, Proceedings of the 5th Annual SAS Global Forum (SGF) 2011 Conference, Software Intelligence Corporation, Spring Valley, CA, USA. http://support.sas.com/resources/papers/proceedings11/273-2011.pdf Phelps, Kent 🎔 Ronda Team (2016), Base SAS[®] and SAS[®] Enterprise Guide[®] ~ Automate Your SAS World With Dynamic Code; Your Newest BFF (Best Friend Forever) in SAS, Proceedings of the 27th Annual MidWest SAS Users Group (MWSUG) 2016 Conference, The SASketeers, Des Moines, IA, USA.

Phelps, Kent 🎔 Ronda Team (2015), The Joinless Join ~ The Impossible Dream Come True; Expanding the Power of SAS® Enterprise Guide® in a New Way, Proceedings of the 26th Annual MidWest SAS Users Group (MWSUG) 2015 Conference, The SASketeers, Des Moines, IA, USA.

http://www.mwsug.org/proceedings/2015/BI/MWSUG-2015-BI-11.pdf

Phelps, Kent 🖤 Ronda Team (2015), SAS[®] Enterprise Guide[®] Base SAS[®] Program Nodes ~ Automating Your SAS World With a Dynamic FILENAME Statement, Dynamic Code, and the CALL EXECUTE Command; Your Newest BFF (Best Friends Forever) in SAS, Proceedings of the 26th Annual MidWest SAS Users Group (MWSUG) 2015 Conference, The SASketeers, Des Moines, IA, USA. http://www.mwsug.org/proceedings/2015/TT/MWSUG-2015-TT-05.pdf

Phelps, Kent 🎔 Ronda Team and Kirk Paul Lafler (2014), The Joinless Join; Expand the Power of SAS® Enterprise Guide® in a New Way, Proceedings of the 25th Annual MidWest SAS Users Group (MWSUG) 2014 Conference, The SASketeers, Des Moines, IA, and Software Intelligence Corporation, Spring Valley, CA, USA.

http://www.mwsug.org/proceedings/2014/BI/MWSUG-2014-BI12.pdf

Phelps, Kent 🎔 Ronda Team and Kirk Paul Lafler (2014), SAS® Commands PIPE and CALL EXECUTE; Dynamically Advancing From Strangers to Your Newest BFF (Best Friends Forever), Proceedings of the 25th Annual MidWest SAS Users Group (MWSUG) 2014 Conference, The SASketeers, Des Moines, IA, and Software Intelligence Corporation, Spring Valley, CA, USA. http://www.mwsug.org/proceedings/2014/BI/MWSUG-2014-BI13.pdf

Phelps, Kent Provide Ronda Team and Kirk Paul Lafler (2013), The Joinless Join; Expand the Power of SAS® Enterprise Guide® in a New Way, Presented at Iowa SAS Users Group (IASUG), The SASketeers, Des Moines, IA, and Software Intelligence Corporation, Spring Valley, CA, USA.

Phelps, Kent 🎔 Ronda Team and Kirk Paul Lafler (2013), SAS® Commands PIPE and CALL EXECUTE; Dynamically Advancing From Strangers to Best Friends, Presented at Iowa SAS Users Group (IASUG), The SASketeers, Des Moines, IA, and Software Intelligence Corporation, Spring Valley, CA, USA.

Phelps, Kent Provide Ronda Team and Kirk Paul Lafler (2013), The Joinless Join; Expand the Power of SAS® Enterprise Guide® in a New Way, Proceedings of the 24th Annual MidWest SAS Users Group (MWSUG) 2013 Conference, The SASketeers, Des Moines, IA, and Software Intelligence Corporation, Spring Valley, CA, USA.

http://www.mwsug.org/proceedings/2013/BB/MWSUG-2013-BB06.pdf

Phelps, Kent 🎔 Ronda Team and Kirk Paul Lafler (2013), SAS® Commands PIPE and CALL EXECUTE; Dynamically Advancing From Strangers to Best Friends, Proceedings of the 24th Annual MidWest SAS Users Group (MWSUG) 2013 Conference, The SASketeers, Des Moines, IA, and Software Intelligence Corporation, Spring Valley, CA, USA. http://www.mwsug.org/proceedings/2013/00/MWSUG-2013-0003.pdf

TRADEMARK CITATIONS

SAS and all other SAS Institute, Inc., product or service names are registered trademarks or trademarks of SAS Institute, Inc., in the USA and other countries. The symbol, [®], indicates USA registration. Other brand and product names are registered trademarks or trademarks of their respective companies.



We have endeavored to provide accurate and helpful information in this SAS White Paper. The information is provided in 'Good Faith' and 'As Is' without any kind of warranty, either expressed or implied. Recipients acknowledge and agree that we and/or our companies are not, and never will be, liable for any problems and/or damages whatsoever which may arise from the recipient's use of the information in this paper. Please refer to your specific Operating System (e.g. UNIX, Windows, or z/OS) Manual, Installation Configuration, and/or in-house Technical Support for further guidance in how to create the SAS code presented in this paper.