Paper 5169-2020

Joinless Join: The Impossible Dream Come True
Using SAS® Enterprise Guide®, PROC SQL, and DATA Step

Kent 💚 Ronda Team Phelps ~ The SASketeers ~ All for SAS & SAS for All!
Illuminator Coaching, Inc. ~ Des Moines, Iowa

ABSTRACT

SAS® Enterprise Guide® and Base SAS® can easily combine data from tables or data sets by using 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 Joinless Join session where we will empower you to expand the power of SAS Enterprise Guide and Base SAS in new ways by creatively overcoming 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 using SAS Enterprise Guide and Base SAS PROC SQL and DATA Step. 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 versatility of the Joinless Join to greatly expand your data transformation and analysis toolkit.

We look forward to introducing you
to the surprising paradox of the
Joinless Join.

INTRODUCTION

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 used 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?

1

☺ **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 SAS Enterprise Guide and Base SAS

❖ Access, combine, and analyze tables for the first time based upon dependencies, indirect relationships, or no relationships at all **~** even with system constraints on remerging

❖ Open up new worlds of table creations, calculations, validations, and filtrations

❖ Prepare unrelated joinless data for reporting 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

<div align="center">

We invite you to journey with us as we empower you to
E   X   P   A   N   D
the power of SAS Enterprise Guide and Base SAS in new ways.

</div>

Brief Overview of Standard PROC SQL Joins and DATA Step Merges

<div align="center">

Just traveling along...
side-by-side.
Harry Macgregor Woods

</div>

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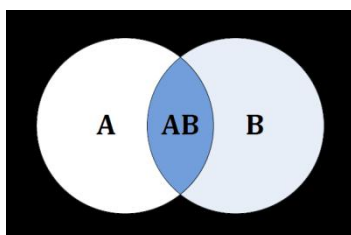
The result of an **Inner Join or Merge** produces only matched rows from the tables. The result is illustrated by the shaded area AB in **Figure 1**.

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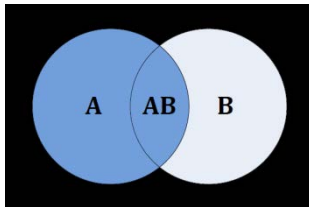
The result of a **Left Outer Join or Merge** produces matched rows from both tables while preserving all unmatched rows from the left table. The result is illustrated by the shaded areas A and AB in **Figure 2**.

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**.
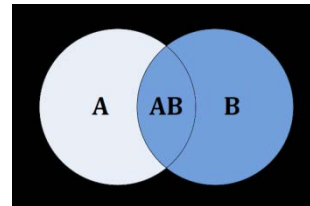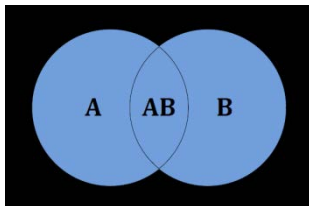

Figure 3. Venn Diagram – Right Outer Join or Merge

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…
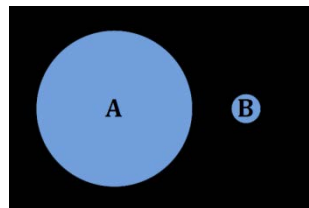

Figure 5. Venn Diagram – Tables Without Like Columns or the Same Variable Name

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?

In the next section we will continue to follow The Power To Know dominoes to find the answer.

☺ **Professor Domino will be our guide** ☺

*Sometimes success is seeing
what we already have
in a new light.*
Dan Miller

First we design a Program Node to create a source table:

```
DATA SMILEY_COMPANY;
  LENGTH Special_Person $20 Special_Num 8 Special_DC_Site $8 Special_Date 8;
  FORMAT Load_Date date9.;
  INFILE DATALINES DELIMITER=',';
  INPUT Special_Person $ Special_Num Special_DC_Site $ Special_Date;
DATALINES;
Smiley,10127911,Lincoln Memorial,22007
Smiley's Wife,10173341,Postal Museum,22007
Smiley's Son,10376606,Air and Space Museum,22007
Smiley's Daughter,10927911,Archives Museum,22007
Smiley's Brother,11471884,ML King Jr Memorial,22007
Smiley's Sister-In-Law,11573691,Holocaust Museum,22007
Smiley's Mother,11975386,Museum of the Bible,22007
Smiley's Father,12071884,Arlington Cemetery,22007
Smiley's Brother-In-Law,12871884,Washington Monument,22007
Smiley's 2nd Cousin,13173691,Jefferson Memorial,21914
RUN;
```

Create
SMILEY_COM
PANY

The Program Node creates the SMILEY_COMPANY source table:

	Special_Person	Special_Num	Special_DC_Site	Special_Date
1	Smiley	10127911	Lincoln Memorial	02APR2020
2	Smiley's Wife	10173341	Postal Museum	02APR2020
3	Smiley's Son	10376606	Air and Space Museum	02APR2020
4	Smiley's Daughter	10927911	Archives Museum	02APR2020
5	Smiley's Brother	11471884	ML King Jr Memorial	02APR2020
6	Smiley's Sister-In-Law	11573691	Holocaust Museum	02APR2020
7	Smiley's Mother	11975386	Museum of the Bible	02APR2020
8	Smiley's Father	12071884	Arlington Cemetery	02APR2020
9	Smiley's Brother-In-Law	12871884	Washington Monument	02APR2020
10	Smiley's 2nd Cousin	13173691	Jefferson Memorial	31DEC2019

❖ The `SMILEY_COMPANY` table contains a row for each employee of the ☺ **Smiley Company** ☺.

❖ Each row contains the `Special_Num` (sales) and the `Special_DC_Site` (where each employee wants to tour as a group) for each `Special_Person` (employee) of the ☺ **Smiley Company** ☺.

❖ `Special_Date` is the date when all of the employees will tour the winning DC Site together.

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

```
DATA National_Average (KEEP=National_Average)

     Date_Check (KEEP=Date_Check)

     DC_Site (KEEP=DC_Site);

     LENGTH Date_Check 8;
     FORMAT Date_Check date9.;

     National_Average = 11474419;
     OUTPUT National_Average;

     Date_Check = '02APR2020'd;
     OUTPUT Date_Check;

     DC_Site = 'Arlington Cemetery';
     OUTPUT DC_Site;

RUN;
```

Here are the 3 additional tables the Program Node creates:

National_Average table:

	National_Average
1	11474419

Date_Check table:

	Date_Check
1	02APR2020

DC_Site table:

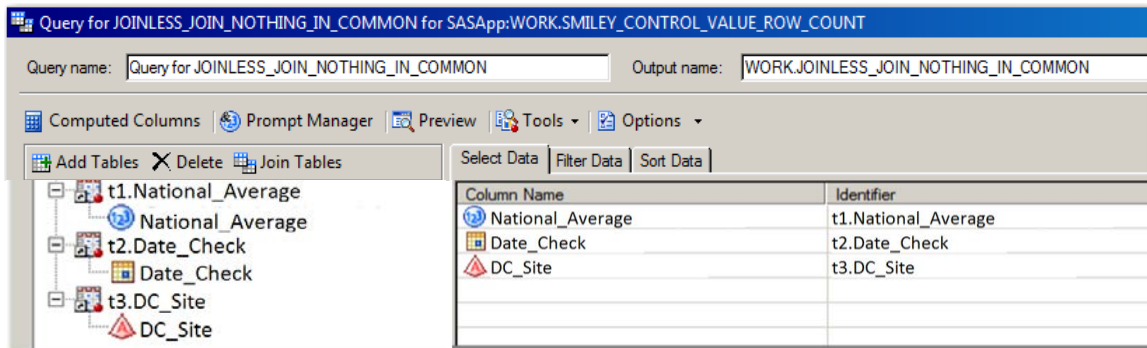|	DC_Site
1	Arlington Cemetery

❖ The `National_Average` table contains the average of the `Special_Num` columns from the **Smiley Company**.

 ➢ We will use a **Joinless Join** to calculate the ratio percentage of each `Special_Num` to `National_Average` without Data Remerging.

❖ The `Date_Check` table contains the `Special_Date` that the **Smiley Company** employees need to be registered for in order to tour the winning `Special_DC_Site` together after *SAS Global Forum 2020*.

 ➢ We will use a **Joinless Join** to validate the employees have a `Special_Date` matching the `Date_Check` and are registered correctly for the tour.

❖ The `DC_Site` table contains the randomly chosen winning `Special_DC_Site` from the entries of the **Smiley Company** employees.

 ➢ We will use a **Joinless Join** to filter down to the employee with the winning entry.

Designing a SAS Enterprise Guide GUI Joinless Join
to combine 3 tables with No Relationships At All
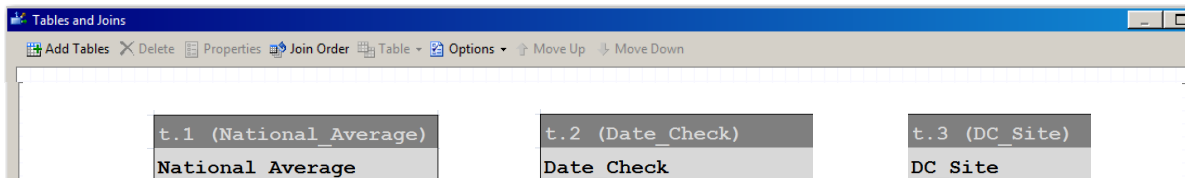## using the 3 additional tables that the 2nd Program Node created:

	National_Average		Date_Check		DC_Site
1	11474419	1	02APR2020	1	Arlington Cemetery

❖ Notice how the 3 columns in the 3 tables have No Relationships At All.

Query for JOINLESS_JOIN_NOTHING_IN_COMMON for SASApp:WORK.SMILEY_CONTROL_VALUE_ROW_COUNT

Query name: Query for JOINLESS_JOIN_NOTHING_IN_COMMON   Output name: WORK.JOINLESS_JOIN_NOTHING_IN_COMMON

Computed Columns | Prompt Manager | Preview | Tools ▾ | Options ▾

Add Tables ✕ Delete | Join Tables   Select Data | Filter Data | Sort Data

- t1.National_Average
  - National_Average
- t2.Date_Check
  - Date_Check
- t3.DC_Site
  - DC_Site

Column Name	Identifier
National_Average	t1.National_Average
Date_Check	t2.Date_Check
DC_Site	t3.DC_Site

❖ Build a Query with the **NATIONAL_AVERAGE**, **DATE_CHECK**, and **DC_SITE** tables.

Tables and Joins

Add Tables ✕ Delete | Properties | Join Order | Table ▾ | Options ▾ | Move Up | Move Down

t.1 (National_Average)	t.2 (Date_Check)	t.3 (DC_Site)
National_Average	Date_Check	DC_Site

❖ Design the Joinless Join based upon the **NATIONAL_AVERAGE**, **DATE_CHECK**, and **DC_SITE** tables having **No Column** to match on.
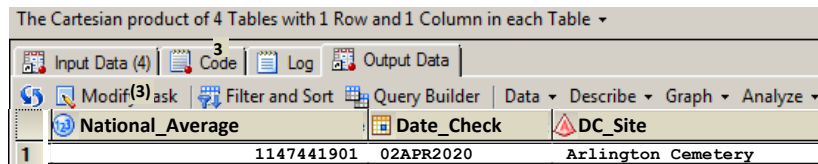

No Column to match on? No Problem ~

**We will design a Joinless Join
based upon No Relationship At All
between the tables.**

What to do, what to do…

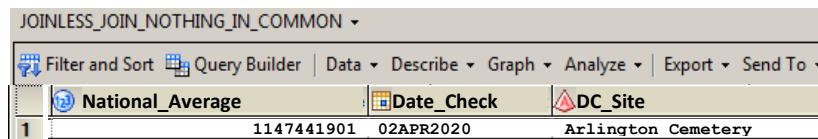*Necessity is the mother of all inventions.*
Plato / Einstein

How the Joinless Join works:

The Cartesian product of 4 Tables with 1 Row and 1 Column in each Table ▾

| Input Data (4) | Code [3] | Log | Output Data |

🔄 Modif[(3)]ask | Filter and Sort | Query Builder | Data ▾ Describe ▾ Graph ▾ Analyze ▾

National_Average	Date_Check	DC_Site
1	1147441901	02APR2020	Arlington Cemetery

❖ The Joinless Join automatically creates a **Cartesian Product** which places the **1 row and 1 column** of the `NATIONAL_AVERAGE`, `DATE_CHECK`, and `DC_SITE` tables to the right of each other.

JOINLESS_JOIN_NOTHING_IN_COMMON ▾

Filter and Sort | Query Builder | Data ▾ Describe ▾ Graph ▾ Analyze ▾ | Export ▾ Send To ▾

National_Average	Date_Check	DC_Site
1	1147441901	02APR2020	Arlington Cemetery

❖ Here is the final result from selecting all 3 columns which is equal to the **Cartesian Product**.

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 potentially become extremely large and unmanageable.

The most noticeable coding characteristic of a **PROC SQL join** which produces a **Cartesian Product i**s the absence of a **WHERE-clause**.

The most noticeable coding characteristics of a **PROC SQL Joinless Join** are a **FROM clause** which does NOT contain any join and the absence of a **WHERE clause** or **ON clause** specifying the column(s) to use for matching.

The most noticeable coding characteristic of a **DATA step Joinless Join i**s the careful and strategic use of multiple **SET statements**.

Always Remember, It's Too Soon To Quit!
Bob Wieland (Mr. Inspiration)
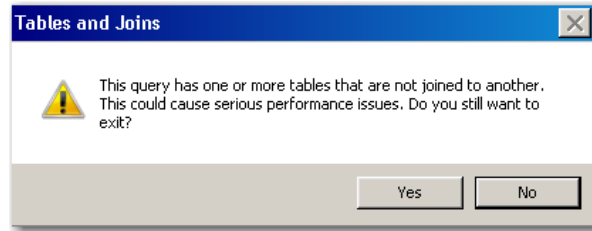
❖ This **Warning Message** always appears whenever tables are joined with a Joinless Join because SAS knows it will create a **Cartesian Product** which can take a lot of extra resources.

Cartesian Product Caution:

When you design your Joinless Join
make sure that one of the tables
has only <u>ONE</u> row!

This PROC SQL also uses a Joinless Join to create the JOINLESS_JOIN_NOTHING_IN_COMMON table:

```
PROC SQL;
    CREATE TABLE JOINLESS_JOIN_NOTHING_IN_COMMON AS
    SELECT National_Average,
           Date_Check,
           DC_Site
    FROM NATIONAL_AVERAGE, DATE_CHECK, DC_SITE;
QUIT;
```

❖ The `PROC SQL` creates a `TABLE` called `JOINLESS_JOIN_NOTHING_IN_COMMON` by selecting `National_Average` from the `NATIONAL_AVERAGE` table, `Date_Check` from the `DATE_CHECK` table, and `DC_Site` from the `DC_SITE` table.

❖ Design the Joinless Join based upon the `NATIONAL_AVERAGE`, `DATE_CHECK`, and `DC_SITE` tables having No Column to match on, a `FROM` clause which does NOT contain any join, and the ABSENCE of a `WHERE` clause or `ON` clause specifying the column(s) to use for matching.

❖ The result from this `PROC SQL` will be identical to what was achieved by designing the GUI Joinless Join on Pages 6 and 7.

This DATA step also uses a Joinless Join to create the JOINLESS_JOIN_NOTHING_IN_COMMON table:

```
DATA JOINLESS_JOIN_NOTHING_IN_COMMON;
    SET NATIONAL_AVERAGE;
    SET DATE_CHECK;
    SET DC_SITE;
RUN;
```

❖ The **DATA** step creates a data set called **JOINLESS_JOIN_NOTHING_IN_COMMON** by **SET**ing the **NATIONAL_AVERAGE**, **DATE_CHECK**, and **DC_SITE** tables using individual **SET** statements.

❖ Since each of the 3 data sets only contain 1 observation, this Joinless Join design individually **SET**'s each data set horizontally placing each column in the 3 data sets side-by-side.

❖ **Please be aware that this Joinless Join using the DATA step will only work if each data set contains only 1 observation.**

❖ Our next example will demonstrate how to design a Joinless Join with a data set which contains more than 1 observation.

Designing a SAS Enterprise Guide GUI Joinless Join to perform a Calculation, Validation, and Filtration:

❖ Build a Query with the **SMILEY_COMPANY** and **JOINLESS_JOIN_NOTHING_IN_COMMON** tables.

❖ The Joinless Join is based upon all 3 columns in the **JOINLESS_JOIN_NOTHING_IN_COMMON** table which indirectly relate to specific columns in the **SMILEY_COMPANY** table.

From Cartesian Product to Finished Product using a Joinless Join:

The Cartesian product of SMILEY_COMPANY and 1 Table with 1 Row and **3** Columns ▾

Input Data (2) | Code | Log | Output Data

↻ 🔍 Modify Task | 🔽 Filter and Sort | 🔳 Query Builder | Data ▾ Describe ▾ Graph ▾ Analyze ▾ | Export ▾ Send To ▾ | 🔳

	Special_Person	Special_ Num	Special_DC_Site	Special_ Date	National_ Average	Date_Check	DC_Site
1	Smiley	10127911	Lincoln Memorial	02APR2020	11474419	02APR2020	Arlington Cemetery
2	Smiley's Wife	10173341	Postal Museum	02APR2020	11474419	02APR2020	Arlington Cemetery
3	Smiley's Son	10376606	Air and Space Museum	02APR2020	11474419	02APR2020	Arlington Cemetery
4	Smiley's Daughter	10927911	Archives Museum	02APR2020	11474419	02APR2020	Arlington Cemetery
5	Smiley's Brother	11471884	ML King Jr Memorial	02APR2020	11474419	02APR2020	Arlington Cemetery
6	Smiley's Sister-In-Law	11573691	Holocaust Museum	02APR2020	11474419	02APR2020	Arlington Cemetery
7	Smiley's Mother	11975386	Museum of the Bible	02APR2020	11474419	02APR2020	Arlington Cemetery
8	Smiley's Father	12071884	Arlington Cemetery	02APR2020	11474419	02APR2020	Arlington Cemetery
9	Smiley's Brother-In-Law	12871884	Washington Monument	02APR2020	11474419	02APR2020	Arlington Cemetery
10	Smiley's 2nd Cousin	13173691	Jefferson Memorial	31DEC2019	11474419	02APR2020	Arlington Cemetery

❖ The Joinless Join automatically creates a **Cartesian Product** which places the **1 row and 3 columns** of the **NATIONAL_AVERAGE**, **DATE_CHECK**, and **DC_SITE** tables to the right of each of the 10 rows and 4 columns in the **SMILEY_COMPANY** table.

Computed Columns

Column	Details
Date_Validation	CASE WHEN t1.Special_Date EQ t2.Date_Check THEN 'SAS Global Forum 2020 Wash DC' ELSE 'Wrong Year' END
DC_Site_Winner	CASE WHEN t1.Special_DC_Site EQ t2.DC_Site THEN 'DC Site Winner!' ELSE 'Maybe Next Time' END
Special_Num_Percent	t.1Special Num/t2.National Average

❖ The Calculation, Validation, and Filtration are represented by Computed Columns derived as shown above using the 3 columns in the **JOINLESS_JOIN_NOTHING_IN_COMMON** table.

	Special_Person	Special_ Num	Special_DC_Site	Special_ Date	Special_ Num_ Percent	Date_Validation	DC_Site_Winner
1	Smiley	10127911	Lincoln Memorial	02APR2020	88.3%	SAS Global Forum 2020 Wash DC	Maybe Next Time
2	Smiley's Wife	10173341	Postal Museum	02APR2020	88.7%	SAS Global Forum 2020 Wash DC	Maybe Next Time
3	Smiley's Son	10376606	Air and Space Museum	02APR2020	90.4%	SAS Global Forum 2020 Wash DC	Maybe Next Time
4	Smiley's Daughter	10927911	Archives Museum	02APR2020	95.2%	SAS Global Forum 2020 Wash DC	Maybe Next Time
5	Smiley's Brother	11471884	ML King Jr Memorial	02APR2020	100.0%	SAS Global Forum 2020 Wash DC	Maybe Next Time
6	Smiley's Sister-In-Law	11573691	Holocaust Museum	02APR2020	100.9%	SAS Global Forum 2020 Wash DC	Maybe Next Time
7	Smiley's Mother	11975386	Museum of the Bible	02APR2020	104.4%	SAS Global Forum 2020 Wash DC	Maybe Next Time
8	Smiley's Father	12071884	Arlington Cemetery	02APR2020	105.2%	SAS Global Forum 2020 Wash DC	DC Site Winner!
9	Smiley's Brother-In-Law	12871884	Washington Monument	02APR2020	112.2%	SAS Global Forum 2020 Wash DC	Maybe Next Time
10	Smiley's 2nd Cousin	13173691	Jefferson Memorial	31DEC2019	114.8%	Wrong Year	Maybe Next Time

❖ Here is the final result with the Calculation, Validation, and Filtration Computed Columns to the right of each of the 10 rows and 4 columns of the **SMILEY_COMPANY** table.

This PROC SQL also uses a Joinless Join to create the SMILEY_JOINLESS_JOIN_ALL_AGAIN table:

```
PROC SQL;
    CREATE TABLE SMILEY_JOINLESS_JOIN_ALL_AGAIN AS
    SELECT t1.Special_Person,
           t1.Special_Num,
           t1.Special_DC_Site,
           t1.Special_Date,
        /* Special_Num_Percent */
           (t1.Special_Num/t2.National_Average)
            FORMAT=PERCENT8.1 AS Special_Num_Percent,
        /* Date_Validation */
           (CASE
               WHEN t1.Special_Date EQ t2.Date_Check
                   THEN 'SAS Global Forum 2020 Wash DC'
               ELSE 'Wrong Year'
            END) AS Date_Validation,
        /* DC_Site_Winner */
           (CASE
               WHEN t1.Special_DC_Site = t2.DC_Site
                   THEN 'DC Site Winner!'
               ELSE 'Maybe Next Time'
            END) AS DC_Site_Winner
    FROM SMILEY_COMPANY t1, JOINLESS_JOIN_NOTHING_IN_COMMON t2;
QUIT;
```

❖ The **PROC SQL** creates a **TABLE** called **SMILEY_JOINLESS_JOIN_ALL_AGAIN** by selecting **Special_ Person**, **Special_Num**, **Special_DC_Site**, and **Special_Date** from the **SMILEY_COMPANY** table.

❖ Design the Joinless Join based upon the **SMILEY_COMPANY** and **JOINLESS_JOIN_NOTHING_IN_ COMMON** tables having No Column to match on, a **FROM** clause which does NOT contain any join, and the ABSENCE of a **WHERE** clause or an **ON** clause specifying the column(s) to use for matching.

❖ The **Special_Num_Percent**, **Date_Validation**, and **DC_Site_Winner** columns are Computed Columns as shown in the GUI Joinless Join on Pages 9 and 10 using the 3 columns in the **JOINLESS_JOIN_NOTHING_IN_COMMON** table.

This DATA Step also uses a Joinless Join to create the SMILEY_JOINLESS_JOIN_ALL_AGAIN table:

```
DATA SMILEY_JOINLESS_JOIN_ALL_AGAIN;
    IF _N_ = 1 THEN SET JOINLESS_JOIN_NOTHING_IN_COMMON;
    SET SMILEY_COMPANY;
    LENGTH Date_Validation $29 DC_Site_Winner $15;
    FORMAT Special_Num_Percent PERCENT8.1;
    /* Special_Num_Percent */
    Special_Num_Percent = Special_Num/National_Average;
    /* Date_Validation */
    IF Special_Date EQ Date_Check
        THEN Date_Validation = 'SAS Global Forum 2020 Wash DC';
    ELSE Date_Validation = 'Wrong Year';
    /* DC_Site_Winner */
    IF Special_DC_Site = DC_Site
        THEN DC_Site_Winner = 'DC Site Winner!';
    ELSE DC_Site_Winner = 'Maybe Next Time';
RUN;
```

❖ The **DATA** step creates a data set called **SMILEY_JOINLESS_JOIN_ALL_AGAIN.**

❖ Before the first observation of the **SMILEY_COMPANY** data set is **SET**, the **JOINLESS_JOIN_ NOTHING_IN_COMMON** data set is **SET**, which causes the single observation to be Joinless Joined with all of the observations in the **SMILEY_COMPANY** data set.

❖ **Please be aware that this Joinless Join using the DATA step will only work if the data set that is SET ~ before the first observation of the second data set is SET ~ contains only 1 observation.**

❖ The **Special_Num_Percent**, **Date_Validation**, and **DC_Site_Winner** columns are Computed Columns as shown in the GUI Joinless Join on Pages 9 and 10 and the **PROC SQL** example on Page 11 with the exception of using **DATA** step syntax when necessary.

Design a Quarterly Report PDF using the results of the Joinless Join of the 3 tables with No Relationships At All:

JOINLESS_JOIN_NOTHING_IN_COMMON ▾		

| 🔍 Filter and Sort | 🔳 Query Builder | Data ▾ Describe ▾ Graph ▾ Analyze ▾ | Export ▾ Send To ▾ |

	National_Average	Date_Check	DC_Site
1	1147441901	02APR2020	Arlington Cemetery

❖ The **Smiley Company** has requested a quarterly report of the results of the Joinless Join of the 3 tables with No Relationships At All that we designed on Pages 6 and 7.

```
ODS PDF FILE='d:\GLOBAL\Smiley Company SAS Global Forum 2020.PDF' NOTOC;
TITLE 'Smiley Company - SAS Global Forum 2020 Washington, D.C. Parameters';
PROC REPORT DATA=JOINLESS_JOIN_NOTHING_IN_COMMON NOWD;
    COLUMNS National_Average Date_Check DC_Site;
    DEFINE National_Average / STYLE={WIDTH=25mm JUST=CENTER} "National Average";
    DEFINE Date_Check       / STYLE={WIDTH=25mm JUST=CENTER} "DC Site Date";
    DEFINE DC_Site          / STYLE={WIDTH=25mm JUST=CENTER} "Winning DC Site";
RUN;
ODS PDF CLOSE;
```

❖ The **ODS PDF FILE** statement opens the **PDF** with no table of contents **– NOTOC.**

❖ The **TITLE** statement includes the title shown at the top of the PDF.

❖ **PROC REPORT** is used to report the contents of the **JOINLESS_JOIN_NOTHING_IN_COMMON** table in the PDF with no default report window **– NOWD.**

❖ The **COLUMNS** statement tells **PROC REPORT** which columns to include in the report.

❖ The **DEFINE** statements provide a **WIDTH** and justification along with renaming each column.

❖ The **ODS PDF CLOSE** statement closes the PDF.

<div style="border:1px solid green;">

11:30 Tuesday, March 31, 2020 1

Smiley Company - SAS Global Forum 2020 Washington, D.C. Parameters

National Average	DC Site Date	Winning DC Site
11474419	02APR2020	Arlington Cemetery

</div>

❖ Here is the PDF of the results of the Joinless Join of the 3 tables with No Relationships At All.

Yea!!!

Strike up the band,

Toss the confetti,

Release the balloons!

Applause…   *Applause…*   *Applause…*

Bring out the treats for everyone!

☺ **Done and Done** ☺

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 SAS Enterprise Guide and Base SAS in new ways. **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 **~** ☺ try saying that statement really fast for fun ☺!

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 SAS Global Forum 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 ~
☺ **Happy SAS Trails to you… until we meet again** ☺

ACKNOWLEDGMENTS

MEET THE AUTHORS

Writing is a permanent legacy.

John C. Maxwell

Kent 💚 Ronda Team Phelps are the co-founders of Illuminator Coaching, Inc., and *The SASketeers: All for SAS and SAS for All!* **~** they have co-authored 14 SAS White Papers **~** presented and co-presented at the MidWest SAS® Users Group (MWSUG) Regional Conferences for the last 7 years, including 4 Hands-On Workshops **~** and co-presented at the SAS Global Forum (SGF) 2018 International Conference.

Kent wants to encourage and equip you to fulfill your life, career, and leadership potential as you build an enduring legacy of inspiration, excellence, and honor **~** B.S. Electrical Engineering **~** Consultant: SAS® Certified Professional Programmer, Senior Data Analyst, and Business Intelligence Developer **~** serving as an essential bridge builder of quality Analysis, Programming, and Data Governance partnering with and connecting the Business Intelligence needs of Leadership, Business, and IT **~** happily programmed in Base SAS® and SAS® Enterprise Guide® since 2007 with a strong focus on Data Excellence and Security while making valuable impacts through innovative, efficient, and cost-saving Business Intelligence solutions.

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 **~** *How wonderful it is that we need not wait a single minute before starting to improve ourselves and our world!* Anne Frank **~** Writer & Coach **~** gifted in helping others to explore and express their hearts and minds through writing **~** served in the Banking and Insurance industries for 19 years.

We invite you to share your valued comments with us:

Kent 💚 Ronda Team Phelps
The SASketeers ~ *All for SAS & SAS for All!*
E-mail: sasketeers@illuminatorcoaching.com

☺ ***We look forward to connecting with you in the future!*** ☺

REFERENCES

Carpenter, Art (2012), *PROC REPORT Basics: Getting Started with the Primary Statements*, Proceedings of the 6th Annual SAS Global Forum (SGF) 2012 International 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.
https://www.foxebook.net/joe-celkos-sql-for-smarties-4th-edition-advanced-sql-programming/

DelGobbo, Vincent (2017), *New for SAS® 9.4: Including Text and Graphics in Your Microsoft Excel Workbooks, Part 2*, Proceedings of SAS® Global Forum (SGF) 2017 International Conference, Cary, NC: SAS Institute Inc.
http://support.sas.com/resources/papers/proceedings17/SAS0127-2017.pdf

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.lexjansen.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 Regional Conference, Software Intelligence Corporation, Spring Valley, CA, USA.
http://www.lexjansen.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 International 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://www.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 International Conference, Software Intelligence Corporation, Spring Valley, CA, USA.
http://support.sas.com/resources/papers/proceedings11/273-2011.pdf

Phelps, Kent ♥ Ronda Team (2019), Hands-On Workshop (HOW): *Base SAS® & SAS® Enterprise Guide® Automate Your SAS® World with Dynamic Code ~ Forwards & Backwards*, Proceedings of the 30th Annual MidWest SAS Users Group (MWSUG) 2019 Regional Conference in Chicago, IL; The SASketeers ~ *All for SAS and SAS for All!* ~ Illuminator Coaching, Inc., Des Moines, IA, USA.
http://www.mwsug.org/proceedings/2019/HW/MWSUG-2019-HW-086.pdf

Phelps, Kent ♥ Ronda Team (2018), Hands-On Workshop (HOW): *The Joinless Join ~ The Impossible Dream Come True Using SAS® Enterprise Guide® and Base SAS® PROC SQL and DATA Step; Expand the Power of SAS® Enterprise Guide® and Base SAS® in New Ways*, Proceedings of the 29th Annual MidWest SAS Users Group (MWSUG) 2018 Regional Conference in Indianapolis, IN; The SASketeers ~ *All for SAS and SAS for All!* ~ Illuminator Coaching, Inc., Des Moines, IA, USA.
http://www.mwsug.org/proceedings/2018/HW/MWSUG-2018-HW-98.pdf

Phelps, Kent ♥ Ronda Team (2018), *Base SAS® and SAS® Enterprise Guide®: Automate Your SAS® World with Dynamic Code ~ Your Newest BFF (Best Friend Forever) in SAS*, Proceedings of 12th Annual SAS® Global Forum (SGF) 2018 International Conference in Denver, CO; The SASketeers *~ All for SAS and SAS for All! ~* Illuminator Coaching, Inc., Des Moines, IA, USA.
http://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2018/2857-2018.pdf

Phelps, Kent ♥ Ronda Team (2017), Hands-On Workshop (HOW): *Base SAS® and SAS® Enterprise Guide® ~ Automate Your SAS World With Dynamic Code; Your Newest BFF (Best Friend Forever) in SAS*, Proceedings of the 28th Annual MidWest SAS Users Group (MWSUG) 2017 Regional Conference in St. Louis, MO; The SASketeers *~ All for SAS and SAS for All! ~* Illuminator Coaching, Inc., Des Moines, IA, USA.
http://www.mwsug.org/proceedings/2017/HW/MWSUG-2017-HW03.pdf

Phelps, Kent ♥ Ronda Team (2016), Hands-On Workshop (HOW): *The Joinless Join ~ The Impossible Dream Come True; Expand the Power of Base SAS® and SAS® Enterprise Guide® in a New Way*, Proceedings of the 27th Annual MidWest SAS Users Group (MWSUG) 2016 Regional Conference in Cincinnati, OH; The SASketeers, Des Moines, IA, USA.
http://www.mwsug.org/proceedings/2016/HW/MWSUG-2016-HW03.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 Regional Conference in Cincinnati, OH; The SASketeers, Des Moines, IA, USA.
http://www.mwsug.org/proceedings/2016/TT/MWSUG-2016-TT11.pdf

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 Regional Conference in Omaha, NE; 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 Regional Conference in Omaha, NE; The SASketeers, Des Moines, IA, USA.
http://www.mwsug.org/proceedings/2015/TT/MWSUG-2015-TT-05.pdf

SAS Institute Inc. (2016), *SAS® 9.4 DATA Step Statements: Reference*; Cary, NC; SAS Institute Inc.
http://documentation.sas.com/?docsetId=lestmtsref&docsetTarget=p00hxg3x8lwivcn1f0e9axziw57y.htm&docsetVersion=9.4&locale=en#p02qkin2ubh7tcn1xael6b1o304f

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.

DISCLAIMER

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 company are/is 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.