Top 10 SAS® Functions in 2017

A brief summary of SAS Communities Survey

- by Flora Fang Liu
What are SAS Functions? Why use SAS Functions?

- **What?**
  - SAS functions perform computations, data manipulation, and enhancement
    - Character
    - Numeric
    - Dates/temporal
    - Mathematics

- **Why?**
  - You need to transform raw/operational data into report-and analytics-ready structures
    - DATA step programming
    - SQL

---

Note: This document is a summary from SAS Communities website. https://communities.sas.com/t5/Ask-the-Expert/Top-10-SAS-Functions/ta-p/391244
Where to find SAS Functions

Function Dictionary
Character manipulation

**SUBSTR**

Extracts a substring from an argument -returns the characters from start to end

**SUBSTR** (string, position<, length>)

- string specifies any SAS character expression.
- position specifies a numeric expression that is the beginning character position.
- length specifies a numeric expression that is the length of the substring to extract.
- If you omit length, SAS extracts the remainder of the expression.

```sas
data newinfo;
set info;
  idnoC=put(idno, 4.);
divld=substr(idnoC, 4, 1);
run;
```

**SCAN**

Selects a given word from a character expression

**SCAN** (string ,n<, delimiter(s)>)

- string specifies any character expression.
- n specifies a numeric expression that produces the number of the word in the character string you want SCAN to select.
- delimiter specifies a numeric expression that produces characters that you want SCAN to use as a word separator in the character string.

```sas
lname= scan(name,1, ',');
fname= scan(name,2, ',');
```

**CATX**

Concatenates character strings, removes leading and trailing blanks, and inserts separators

**CATX** (separator, string<1<, ...string< n>)

- separator specifies a character string that is used as a separator between concatenated strings.
- string specifies a SAS character string.
- The CATX function returns a value to a variable, or returns a value in a temporary buffer.

```sas
newname = catx(' ',fname, lname);
lnname= scan(name,1, ',');
fname= scan(name,2, ',');
newname = catx(' ',fname, lname);
```

**FIND**

Searches for a specific substring of characters within a character string that you specify

**FIND** (string, substring<,modifiers>). 

- string specifies a character constant, variable, or expression that will be searched for subtrings.
- substring is a character constant, variable, or expression that specifies the substring of characters to search for in string.
- Modifiers is a character constant, variable, or expression that specifies one or more modifiers. The following modifiers can be in uppercase or lowercase:
  - I ignores character case during the search. If this modifier is not specified, FIND only searches for character substrings with the same case as the characters in substring.

Use the FIND function to identify a misspelled name and write a message to the SAS log.

```sas
if find(newname, 'Lau', 'i') > 0 then
  put 'value found ' newname;
```

Note: This document is a summary from SAS Communities website. https://communities.sas.com/t5/Ask-the-Expert/Top-10-SAS-Functions/ta-p/391244
Temporal / date / time

**DATEPART**

Extracts the date from a SAS datetime value

**DATEPART (datetime)**

- Datetime specifies a SAS expression that represents a SAS datetime value.

Example:

Extract the SAS date portion of HIREDATE using the DATEPART function

```
newhire= datepart(hiredate);
```

**DATEDIF**

Returns the number of days between two dates

**DATEDIF (sdate, edate, basis)**

- `sdate` specifies a SAS date value that identifies the starting date.
- `edate` specifies a SAS date value that identifies the ending date.
- `basis` identifies a character constant or variable that describes how SAS calculates the date difference.

- '30/360' or '360' specifies a 30 day month and a 360 day year.
- 'ACT/ACT' or 'Actual' uses the actual number of days between dates.

Example:

Calculate the number of days (years) between the 2 new hire and termination dates

```
if newtermne .thendo;
    dayofserv= datdif(newhire, newterm, 'ACT/ACT');
    yearofserv= yrdif(newhire, newterm, 'ACT/ACT');
end;
```

Note: This document is a summary from SAS Communities website. [https://communities.sas.com/t5/Ask-the-Expert/Top-10-SAS-Functions/ta-p/391244](https://communities.sas.com/t5/Ask-the-Expert/Top-10-SAS-Functions/ta-p/391244)
**Data conversion**

**PUT**

PUT (source, format)
- Converts the expression to a character string.
- Always returns a character value.
- Source argument can be character or numeric.
- Format contains the SAS format that you want applied to the variable or constant that is specified in the source.
- To override the default alignment, you can add an alignment specification to a format:
  - L: left aligns the value.
  - C: centers the value.
  - R: right aligns the value.

Example:
**Convert Numeric to Character**

Since IDNO is stored as numeric, it must be converted to character via the PUT function.

```sas
idnoC = put(idno, 4.);
if find(newname, 'Lau', 'i') > 0 then
  put 'value found ' newname;
```

**INPUT**

INPUT (character-expression, informat)
- Converts a string expression using the specified informat.
- Often used to convert character to numeric.

Example:
**Convert TERMDATE to a SAS date using the INPUT function**

```sas
newterm = input(termdate, date9.);
```

---

*Note: This document is a summary from SAS Communities website. [https://communities.sas.com/t5/Ask-the-Expert/Top-10-SAS-Functions/ta-p/391244]*
Numeric

**SUM**

SUM Syntax

```plaintext
SUM (argument1, argument2, ...)
```

- Argument(s) are numeric.
- The argument list can consist of a variable list, which can be preceded by OF.
- If all the arguments have missing values, the result is a missing value.

**ROUND**

The ROUND function will express the total compensation as a whole number without decimal positions.

ROUND Syntax

```plaintext
ROUND (argument <,rounding-unit>)
```

- argument is a numeric constant, variable, or expression to be rounded.
- rounding-unit is a positive, numeric constant, variable, or expression that specifies the rounding unit.

```plaintext
totalcomp = sum(salary, bonus, merit);
totalcomp = round(totalcomp);
```

**Do you use it?**

Yes | No

---

Note: This document is a summary from SAS Communities website. https://communities.sas.com/t5/Ask-the-Expert/Top-10-SAS-Functions/ta-p/391244