When ANY Function Will Just NOT Do

Richann Watson, Experis
Karl Miller,inVentiv Health

Introduction to the ‘ANY’ or ‘NOT’ Functions

The What and the Why

- What are the ‘ANY’ and ‘NOT’ functions?
  - Functions that search a string for a presence or absence of specific type of character(s)
  - ‘ANY’ returns position of first instance the character is found
  - ‘NOT’ returns position of first instance the character is **not** found

- Why use them?
  - Reduce assumptions regarding data
  - Reduce extended time in debugging

‘ANY’ or ‘NOT’ Functions

Function Search Criteria	ANY Function	NOT Function
alpha-numeric character	ANYALNUM(*string*, <*start*>)	NOTALNUM(*string*, <*start*>)
alphabetic character	ANYALPHA(*string*, <*start*>)	NOTALPHA(*string*, <*start*>)
digit	ANYDIGIT(*string*, <*start*>)	NOTDIGIT(*string*, <*start*>)
valid as the first character of a SAS variable*	ANYFIRST(*string*, <*start*>)	NOTFIRST(*string*, <*start*>)
lowercase letter	ANYLOWER(*string*, <*start*>)	NOTLOWER(*string*, <*start*>)
valid in a SAS variable*	ANYNAME(*string*, <*start*>)	NOTNAME(*string*, <*start*>)
punctuation character	ANYPUNCT(*string*, <*start*>)	NOTPUNCT(*string*, <*start*>)
white-space character	ANYSPACE(*string*, <*start*>)	NOTSPACE(*string*, <*start*>)
uppercase letter	ANYUPPER(*string*, <*start*>)	NOTUPPER(*string*, <*start*>)

* SAS variable name under VALIDVARNAME=V7

Examples

Use in SDTM

- Populating LBSTRESN in LB domain and determining data issues

  anyalpha(lborres)

LBORRES	ANYALPHA	Handling
1.4	0	Convert to standardized unit and populate LBSTRESC and LBSTRESN
TRACE	1	No conversion required. Copy to LBSTRESC.
CANCELLED	1	Data should be queried or fixed so that LBSTAT = 'NOT DONE' and LBREASND = 'CANCELLED'

Use in ADaM

- Determine if AVALC is a partial numeric
- Determine if AVAL is integer or float

```
not(anyalpha(AVALC))
anypunct(AVALC)
```

Row	AVALC	ANYALPHA	NOTALPHA	ANYPUNCT	NOTDIGIT
1	TESTING	1	8	0	1
2	<1.0	0	1	1	1
3	1.2	0	1	2	2
5	354.12	0	1	4	4
6		0	1	0	1
7	___123	0	1	1	1
8	124!	0	1	4	4
9	47821	0	1	0	6
10	1TEST	2	1	0	2

Week Number from Text

Row	VISIT	VISIND	VISCOMP	VISANY
1	w4	4	4	4
2	w 4	4	4	4
3	W-4	4	4	4
5	WK4	4	4	4
6	Wk: 4	4	4	4
7	Week 4	4	4	4
8	WEEK:4	4	4	4

- More commonly used approaches

```
visind = input(substr(visit,
                 indexc(visit, '123456790')), best.);
viscomp = input(compress(visit,,'dk'), best.);
```

- Another suggestion would be

```
visany = input(substr(visit, anydigit(visit)), best.);
```

Other Examples Used

- Parsing out the week number and day number
- Defining ISO 8601 date/time variable

*Details of these examples along with the code can be found in the paper.

Conclusion

- Simple yet powerful functions when used correctly
- Perform well across multiple data types, studies or compounds
- Applied structure to meet your specific needs
- Beneficial for efficiency in programming

Richann Watson
Experis
richann.watson@experis.com

Karl Miller
inVentiv Health
karl.miller@inventivhealth.com