## PROC SQL Features: ANSI or SAS Enhancement?

Category	Enhancements	ANSI
Statements	VALIDATE	All others.
		Note: COMMIT, ROLLBACK, GRANT
		statements are not supported by PROC SQL.
Operators	CONTAINS (?), SOUNDS LIKE (=*), EQ, NE, GT, LT,	=, ==, <>, !=, <=, >=, <, >, +, -, *, /, %, IN,
	GE, LE	LIKE, BETWEEN-AND, EXISTS, OR, AND, and
		keywords NOT, ANY and ALL
Functions	Functions PROC SQL supports SAS functions not associated with	Proc SQL supports the following ANSI standard
	arrays. Only functions listed to the right are ANSI.	functions:
	A few examples of PROC SQL supporting a SAS function	ABS, AVG, BTRIM, CEILING, COALESCE,
	instead of the ANSI equivalent:	COUNT, FLOOR, MAX, MIN, MOD, POWER,
	1. Character Variable Length:	ROUND, SIGN, SQRT, SUBSTRING, SUM
	ANSI: CHARACTER_LENGTH(column),	
	PROC SQL: LENGTH(column) [a SAS function]	
	2. Trim leading/trailing/both blanks	
	ANSI: BTRIM( <both,leading,trailing>"" FROM string)</both,leading,trailing>	
	PROC SQL: STRIP(column) [trims both ends]	
	TRIM(column) [trims trailing blanks]	
	LEFT(column) [trims leading blanks]	
Modifiers	CALCULATED, FORMAT=, INFORMAT=, LABEL=,LENGTH=	ALL, CORR, DISTINCT, ESCAPE
Metadata	DICTIONARY.TableName	INFORMATION_SCHEMA.TableName
	Note: Similar metadata is available for DATA STEP	
	processing as SASHELP views (for example,	
	SASHELP.VTABLE contains the information in	
	DICTIONARY.TABLES)	
Miscellaneous	LIBNAME table qualifier	SCHEMA table qualifier
	(SELECT X from <b>LIBNAME.</b> TABLE_NAME)	(SELECT X from <b>SCHEMA.</b> TABLE_NAME),
	SCHEMA table qualifier	Column order reference (ORDER BY 2),
	(SELECT X from <b>SCHEMA.</b> TABLE_NAME),	'*' in the Count function
	Column order reference (ORDER BY 2),	(SELECT Count(*) from TABLE)
	'*' in the Count function	'*' as a select wildcard (SELECT * from TABLE)
	(SELECT Count(*) from TABLE)	
	'*' as a select wildcard (SELECT * from TABLE)	
PROC SQL	STIMER, ERRORSTOP, SORTSEQ=, NOEXEC, FEEDBACK	
Options		