Postgres PL/pgSQL String and Formating Functions

Source: http://www.postgresql.org/docs/8.1/static/functions-string.html
http://www.postgresql.org/docs/8.1/static/functions-formatting.html
http://www.postgresql.org/docs/8.1/static/functions-array.html
http://www.postgresql.org/docs/8.1/static/functions-aggregate.html
http://www.postgresql.org/docs/9.1/static/functions-string.html
Practical http://www.adderpit.com/practical-postgresql/book1.htm

STRING FUNCTIONS

Function	**Return Type**	**Description**	**Example**	**Result**
string || string	text	String concatenation	'Post' || 'greSQL'	PostgreSQL
char_length(string)	int	Number of chars in string	char_length('jose');	4
lower(string)	text	Convert string to lower case	lower('TOM')	tom
overlay(string placing string from int [for int])	text	Replace substring	overlay('Txxxxas' placing 'hom' from 2 for 4)	Thomas
position(substring in string)	int	Location of specified substring	position('om' in 'Thomas')	3
substring(string [from int] [for int])	text	Extract substring	substring('Thomas' from 2 for 3)	hom
trim([leading | trailing | both] [characters] from string)	text	Remove the longest string containing only the chars (space defaults) form start/end of string	trim(both 'x' from 'xTomxx')	Tom
upper(string)	text	Convert string to uppercase	upper('tom')	TOM
initcap(text)	text	Convert first letter of each word to uppercase and rest to lower case	initcap('hi THOMAS')	Hi Thomas
length(string text)	int	Number of characters in string	length('jose')	4
md5(string text)	text	Calculates the MD5 hash	md5('abc')	900150983cd24fb0d6963f7d28e17f72
strpos(string, substring)	int	Location of specified substr	strpo('high','ig')	2
substr(string, from [, count])	int	Extract substring	substr('alphabet',3,2)	ph
replace(string text, from text, to text)	text	Replace all occurrences in string of substring from with substring to	replace('abcdefabcdef', 'cd', 'XX')	abXXefabXXef

e.g. UPDATE dataelement SET name=replace(name,'IPD2012 ','DTC2012 ') WHERE

dataelement.name LIKE 'DTC2012 %';

FORMATTING FUNCTIONS

Function	Return Type	Description	Example
to_char(timestamp, text)	text	convert time stamp to string	to_char(current_timestamp, 'HH12:MI:SS')

Array Functions and Operators

Operator	Description	Example	Result
=	equal	ARRAY[1.1,2.1,3.1]::int[] = ARRAY [1,2,3]	t
<>	not equal	ARRAY[1,2,3] <> ARRAY [1,2,4]	t
<	less than	ARRAY[1,2,3] < ARRAY [1,2,4]	t
>	greater than	ARRAY[1,4,3] > ARRAY [1,2,4]	t
<=	less than or equal	ARRAY[1,2,3] <= ARRAY [1,2,3]	t
>=	greater than or equal	ARRAY[1,4,3] >= ARRAY [1,4,3]	t
||	array-to-array concatenation	ARRAY[1,2,3] || ARRAY [4,5,6]	{1,2,3,4,5,6}
||	array-to-array concatenation	ARRAY[1,2,3] || ARRAY[[4,5,6], [7,8,9]]	{{1,2,3},{4,5,6},{7,8,9}}
||	element-to-array concatenation	3 || ARRAY [4,5,6]	{3,4,5,6}
||	array-to-element concatenation	ARRAY[4,5,6] || 7	{4,5,6,7}

Operator	Return type	Description	Example	Result
array_cat (anyarray, anyarray)	anyarray	concatenate two arrays	array_cat(ARRAY[1,2,3], ARRAY [4,5])	{1,2,3,4,5}
array_append (anyarray, anyelement)	anyarray	append an element to the end of an array	array_append(ARRAY[1,2], 3)	{1,2,3}
array_prepend (anyelement, anyarray)	anyarray	append an element to the beginning of an array	array_prepend(1, ARRAY [2,3])	{1,2,3}
array_dims (anyarray)	text	returns a text representation of array's dimensions	array_dims(ARRAY[[1,2,3], [4,5,6]])	[1:2][1:3]
array_to_string (anyarray, text)	text	concatenates array elements using provided delimiter	array_to_string(ARRAY[1, 2, 3], '~^~')	1~^~2~^~3
string_to_array (text, text)	text[]	splits string into array elements using provided delimiter	string_to_array('xx~^~yy~^~zz', '~^~')	{xx,yy,zz}