

SQL Top 20 Regular Expressions

Joachim Forster
Lisa Dräxlmaier GmbH
Vilsbiburg

Schlüsselworte

Regular, Expression, SQL, PL/SQL

Einleitung

Beim Umgang mit Daten ist man täglich damit konfrontiert, Muster in Datenfeldern zu finden und mit diesen Mustern auch Prüfungen, Ersetzungen, Extrahierungen oder Zählungen vorzunehmen. Im Vergleich zu den traditionellen Zeichen-Funktionen wie etwa like, replace oder translate ist das Verstehen der Syntax von Regular Expressions jedoch wesentlich komplizierter. In der kurzen Vortragszeit werden möglichst viele "real world"-Problemstellungen aufgezeigt und Lösungen mit "REGEXP"-Funktionen präsentiert. Es wird außerdem erklärt, wann es aus Gründen der Machbarkeit, Lesbarkeit oder Performance Sinn macht, traditionelle Funktionen zu bevorzugen.

Regular Expressions sind keine Erfindungen von SQL Datenbanken, sondern haben erst relativ spät Einzug in die Datenbankwelt gefunden.

In vielen Programmiersprachen, Betriebssystemen und Texteditoren werden Regular Expressions schon seit den Anfängen der Softwareentwicklung als Werkzeug zum "Suchen" bzw. "Suchen und Ersetzen" verwendet.

Wer also in anderen Programmierumgebungen bereits Übung im Umgang mit "REGEXE's" gesammelt hat, kann dieses Wissen auch in SQL verwenden. Im Detail unterscheiden sich die Implementierungen und unterstützten Features von REGEXP Funktionalitäten erheblich. Die Einbettung in die jeweilige Programmiersprache (in unserem Fall SQL und PL/SQL) muss der Programmierer erlernen.

Im Vortrag wird zuerst auf allgemeine Grundlagen eingegangen, es werden die in Oracle SQL zur möglichen "REGEXP" Metazeichen aufgelistet und deren Bedeutung kurz erklärt.

Eine zweite Übersicht erläutert vordefinierte Zeichenklassen.

Schließlich werden die 5 in SQL zur Verfügung stehenden "REGEXP"-Funktionen mit ihren Parametern tabellarisch dargestellt.

Mit diesem Wissen werden dann 20 Anwendungsfälle aufgezeigt. Das beginnt beim einfachen Zählen der Kommas in einem Text (... wie ging das gleich nochmal in Zeiten vor "REGEXP_COUNT", wie schneidet die "alte" Technik beim Performance Vergleich ab ?) bis zum Zerlegen von (vielen) CSV Datensätzen in einzelne Werte. Bekannte Muster wie eine Prüfung auf "IS_NUMERIC" dürfen in dieser Aufzählung natürlich auch nicht fehlen.

Falls die "REGEXP"-Formeln nicht alle gleich verstanden werden, können die mit Demodaten ausgestatteten SQL's natürlich auch sehr schnell selbst nachgetestet werden.

Zwei Buchempfehlungen (... und eine Buchwarnung) sowie interessante Links runden den Vortrag ab.

Regular Expression Functions and Parameters in Oracle SQL and PL/SQL

FUNCTION NAME	RETURN TYPE	SRC STR	REXP PATTERN	REPLACE STR	POSITION	OCCURRENCE	RETURN PARAM	MODIFIER	SUB EXPRESSION
		STR	STR	STR	INT >= 1	INT >= 1(0)	INT (0,1)	STR icnmx	INT [0-9]
REGEXP_LIKE	BOOL	1	2					3*	
REGEXP_INSTR	INT	1	2		3*	4*	5*	6*	7*
REGEXP_SUBSTR	STR	1	2		3*	4* 0=err		5*	6*
REGEXP_REPLACE	STR	1	2	3*	4*	5* 0=all		6*	
REGEXP_COUNT	INT	1	2		3*			4*	

* = optional

List of SQL Top-20 Regular Expressions

TOP	Title
1.1:	Ascii Matrix (NSL_CHARACTERSET = WE8ISO8859P15)
1.2:	List Characters of (Named) Character Classes
2.1:	REGEXP_COUNT
2.2:	REGEXP_REPLACE (REPLACESTR = #)
2.3:	REGEXP_SUBSTR
2.4:	REGEXP_INSTR
2.5:	REGEXP_LIKE(_1) (boolean datatype not supported in Oracle)
3:	Count characters, words, sentences, lines, ... in test
4:	Extract Token by Position (including NULL Values)
5:	Extract Prefix and Suffix (nonmatching, greedy, lazy)
6:	Extract Pah, File and Extension Name
7:	Value for Key in KVP String
8:	Key Value Pairs to CSV Header and Data
9:	CSV Row to Table
10:	CSV Rows to Table
11.1:	OR_LIKE
11.2:	OR_LIKE_TO_REXP Pattern Translation function
12.1:	Is string upper or lower case
12.2:	Is upper ? ... Lots of errors can happen ... so test carefully
13:	Is / make numeric (NLS_NUMERIC_CHARACTERS = ',.')
14:	Check Valid Oracle Identifier (quoted and nonquoted)
15:	Remove consecutive duplicates
16:	Wrap Text
17:	Convert camelCase to UPPER_CASE code style and vice versa
18:	Remove Comments
19:	...
20:	...

Kontaktadresse:

Joachim Forster
Lisa Dräxlmaier GmbH
Landshuter Straße 100
D-84137 Vilsbiburg

E-Mail joachim.forster@draexlmaier.com
Internet: www.draexlmaier.com