

BÖLÜM- 14: REGULAR EXPRESSIONS

- SQL içerisinde Regular Expression kullanımı
- Regular Expressions içerisinde Meta karakter kullanımı
- REGEXP_LIKE
- REGEXP_INSTR
- REGEXP_SUBSTR
- REGEXP_REPLACE
- REGEXP_COUNT
- ...

REGEXP_LIKE

- Standart LIKE operatörü ile çok benzerlik gösterir fakat daha gelişmiş sonuç üretme yapısına sahiptir.
- `select * from HR.JOBS where job_title like '%re%';`
- `select * from HR.JOBS where regexp_like(job_title, 're');`

JOB_ID	JOB_TITLE	MIN_SALARY	MAX_SALARY
AD_PRES	President	20080	40000
AD_VP	Administration Vice President	15000	30000
SA_REP	Sales Representative	6000	12008
MK_REP	Marketing Representative	4000	9000
HR_REP	Human Resources Representative	4000	9000
PR_REP	Public Relations Representative	4500	10500

1: 1 MSDEVECI@TESTDB Modified

REGEXP_LIKE

- S ile başlayan pozisyonları normal like ve regexp_like ile yapalım
- `select * from HR.JOBS where job_title like 'S%';`
- `select * from HR.JOBS where regexp_like(job_title, '^S');`

JOB_ID	JOB_TITLE	MIN_SALARY	MAX_SALARY
SA_MAN	Sales Manager	10000	20080
SA_REP	Sales Representative	6000	12008
ST_MAN	Stock Manager	5500	8500
ST_CLERK	Stock Clerk	2008	5000
SH_CLERK	Shipping Clerk	2500	5500

4: 1	MSDEVECI@TESTDB	Modified
------	-----------------	----------

REGEXP_LIKE

- İlk karakteri S ve 3.karakteri l olan pozisyonları listeleyelim
- ```
select * from HR.JOBS where job_title like 'S_l%';
```
- ```
select * from HR.JOBS where regexp_like(job_title, '^S.l');
```

JOB_ID	JOB_TITLE	MIN_SALARY	MAX_SALARY
SA_MAN	Sales Manager	10000	20080
SA_REP	Sales Representative	6000	12008

7: 1 Row 1 of 2 total rows MSDEVECI@TESTDB Modified

REGEXP_LIKE

- Standart LIKE operatörünün yapamadığı bazı işlemleri REGEXP_LIKE tek komutla yapabilir.
- 2.harfi a,b yada c olanları getir gibi
- `select * from HR.JOBS where regexp_like(job_title, '^.[abc]');`

JOB_ID	JOB_TITLE	MIN_SALARY	MAX_SALARY
FI_ACCOUNT	Accountant	4200	9000
AC_MGR	Accounting Manager	8200	16000
SA_MAN	Sales Manager	10000	20080
SA_REP	Sales Representative	6000	12008
MK_MAN	Marketing Manager	9000	15000
MK_REP	Marketing Representative	4000	9000

10: 1 | Row 1 of 6 total rows | MSDEVCI@TESTDB | Modified

REGEXP_INSTR

- Standart INSTR operatörü ile çok benzerlik gösterir fakat daha gelişmiş sonuç üretme yapısına sahiptir.
- Pattern : Buluncak değer
- Position : Aranmaya başlanacak pozisyon
- Occurence : Kaçınıcısi bulunacak
- Return Option : 0 = bulunca dur, 1 = bir sonrakiki bul
- Parameters : Arama şekli (i,c,n,m,x)

```
regexp_instr (string, pattern)
regexp_instr (string, pattern, position)
regexp_instr (string, pattern, position, occurence)
regexp_instr (string, pattern, position, occurence, return-option)
regexp_instr (string, pattern, position, occurence, return-option, parameters)
```

REGEXP_INSTR

- `select first_name,last_name,regexp_instr(first_name, 'a') pozisyon1, instr(first_name, 'a') pozisyon2 from hr.employees;`

FIRST_NAME	LAST_NAME	POZISYON1	POZISYON2
Donald	OConnell	4	4
Douglas	Grant	6	6
Jennifer	Whalen	0	0
Michael	Hartstein	5	5
Pat	Fay	2	2
Susan	Mavris	4	4

- *--Boş olmayan karakterle başlayan 4. karakterin pozisyonu*
- `select regexp_instr('500 oracle parkway, redwood shores, ca',`
- `'[^]+', 1, 4) "regexp_instr" from dual;`

regexp_instr
21

REGEXP_INSTR

- *e harfinin 2. karaktderden sonra 2. kez bulunduğun pozisyonunu ver, büyük-küçük harf duyarlılığı olmasın*
- `select regexp_instr('Regular Expressions', 'e', 2, 2, 0, 'i')`
pozisyon `from dual;`

POZISYON
9

21: 1 Row 1 of 107 total rows MSDEVECI@TESTDB

REGEXP_SUBSTR

- Standart SUBSTR operatörü ile çok benzerlik gösterir fakat daha gelişmiş sonuç üretme yapısına sahiptir.
- `select first_name, substr(first_name, 2, 2) str1, regexp_substr(first_name, '..', 2) str2`
- `from hr.employees;`

	FIRST_NAME	STR1	STR2
▶	Donald	on	on
	Douglas	ou	ou
	Jennifer	en	en
	Michael	ic	ic
	Pat	at	at

25: 1 | Row 1 of 107 total rows | MSDEVECI@TESTDB

REGEXP_SUBSTR

- --İki virgül arasındaki değer (, Redwood Shores,)
- `select regexp_substr('500 Oracle Parkway, Redwood Shores, CA', '^[^,]+,') sonuc from dual;`

SONUC
, Redwood Shores,

28: 1 | Row 1 of 107 total rows | MSDEVECI@TESTDB

REGEXP_REPLACE

- Standart REPLACE operatörü ile çok benzerlik gösterir fakat daha gelişmiş sonuç üretme yapısına sahiptir.
- Aynı zamanda TRANSLATE komutunun bir kısım yeteneklerini de içerir.



REGEXP_REPLACE

- select first_name, replace(first_name,'o','a') a1,
regexp_replace(first_name, 'o', 'a') a2,
translate(first_name, 'oa','***') b1,
regexp_replace(first_name,'[abc]', '*') b2,
translate(first_name, 'abc','xyz') c1,
regexp_replace(first_name,'[abc]', 'xyz') c2,
regexp_replace(first_name,'an', '*') d1
from hr.employees;

FIRST_NAME	A1	A2	B1	B2	C1	C2	D1
Donald	Danald	Danald	D*n*ld	Don*ld	Donxld	Donxyzld	Donald
Douglas	Dauglas	Dauglas	D*ugl*s	Dougl*s	Douglxs	Douglxyzs	Douglas
Jennifer	Jennifer	Jennifer	Jennifer	Jennifer	Jennifer	Jennifer	Jennifer
Michael	Michael	Michael	Mich*el	Mi*h*el	Mizhxel	Mixyzhxyzel	Michael
Pat	Pat	Pat	P*t	P*t	Pxt	Pxyzt	Pat
Susan	Susan	Susan	Sus*n	Sus*n	Susxn	Susxyzn	Sus*

REGEXP_COUNT

- 11G ile birlikte gelen, kullanımı kolay ve çok yetnekli bir fonksiyondur. Aynı zamanda performaslı çalışır.
- `select first_name,`
`regexp_count(first_name, 'a') a1, --a'ların sayısı`
`regexp_count(first_name, 'a', 5) b1, --5. karakterden sonrakiler`
`regexp_count(first_name, 'a', 1, 'i') c1, --büyük-küçük harfe bakmfirst_namean`
`regexp_count(first_name, '[abc]') a2, --A,B,C toplam sayısı`
`regexp_count(first_name, '[^]+') a3, --kelime sayısı`
`regexp_count(first_name, '!.') a4 --tüm karkter sayısı`
`from hr.employees;`

FIRST_NAME	A1	B1	C1	A2	A3	A4
Donald	1	0	1	1	1	6
Douglas	1	1	1	1	1	7
Jennifer	0	0	0	0	1	8
Michael	1	1	1	2	1	7
Pat	1	0	1	1	1	3
Susan	1	0	1	1	1	5

43: 1 Row 1 of 107 total rows MSDEVECI@TESTDB