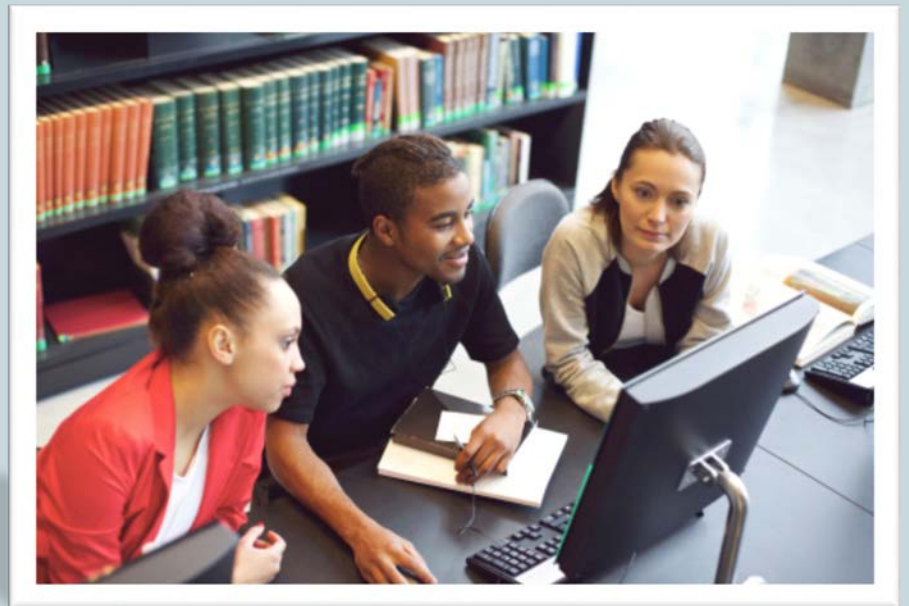




Programação de Banco de Dados com SQL

17-3

Expressões Regulares



Objetivos

Esta lição abrange os seguintes objetivos:

- Descrever expressões regulares
- Usar expressões regulares para pesquisar, relacionar e substituir strings em instruções SQL
- Construir e executar expressões regulares e constraints de verificação

Finalidade

- Às vezes, você precisa localizar ou substituir uma porção de texto em uma coluna, string de texto ou documento.
- Você já sabe como fazer correspondência de padrões simples usando LIKE e curingas.
- Às vezes, é possível que você precise buscar strings de texto muito complexas, como extrair todas as URLs de uma porção de texto.
- Em outros momentos, pode lhe ser solicitado fazer uma pesquisa mais complexa, como localizar todas as palavras cujo segundo caractere seja uma vogal.

Finalidade

- As expressões regulares compõem um método para descrever padrões simples e complexos, para pesquisa e manipulação.
- Elas são amplamente usadas na área de computação e não se limitam ao Oracle.
- A implementação do Oracle das expressões regulares é uma extensão do POSIX (Portable Operating System for UNIX) e, por isso, elas são totalmente compatíveis com o padrão POSIX, controlado pelo IEEE (Institute of Electrical and Electronics Engineers).

Expressões Regulares

- O uso das expressões regulares se baseia na utilização de metacaracteres.
 - Metacaracteres são caracteres especiais que possuem um significado especial, como um caractere curinga, um caractere repetido, um caractere não correspondente ou uma série de caracteres.
 - Você pode usar vários símbolos de metacaracteres predefinidos na correspondência de padrões.
 - Os próximos slides listam alguns dos metacaracteres e fornecem uma explicação breve sobre cada um.

Metacaracteres

Símbolo	Descrição
. (ponto)	Corresponde a qualquer caractere no conjunto de caracteres suportados, exceto NULL
?	Corresponde a zero ou uma ocorrência
*	Corresponde a zero ou mais ocorrências
+	Corresponde a um ou mais ocorrências
()	Expressão de agrupamento, tratada como uma subexpressão
\	Caractere de escape
 	Operador de alternância para especificar correspondências alternativas
^/\$	Corresponde ao início/fim da linha
[]	Expressão entre colchetes para uma lista de correspondências equivalente a qualquer uma das expressões representadas na lista

Exemplos de Expressões Regulares

- Uma expressão regular simples é muito semelhante à pesquisa com curinga que você já conhece.
- Vamos ver um exemplo: vamos usar o operador ponto para buscar a letra 'a' seguida por qualquer caractere e pela letra 'c'.
- Como uma expressão regular, isso seria feito assim: 'a.c'.
- A mesma expressão como uma pesquisa com curinga SQL padrão seria: WHERE coluna LIKE 'a_c'.

Exemplos de Expressões Regulares

- Qual das strings abaixo corresponderiam a 'a.c'?
1. 'ABC',
 2. 'abc',
 3. 'aqx',
 4. 'axc',
 5. 'aBc',
 6. 'abC'
 7. 'Amc'
 8. 'amrc'



Exemplos de Expressões Regulares

- As strings em vermelho corresponderiam à string de pesquisa 'a.c'
- 'ABC', 'abc', 'aqx', 'axc', 'aBc', 'abC' 'Amc' 'amrc'
- Os outros exemplos não são válidos porque têm o caractere na posição errada ou um uso diferente de letras maiúsculas e minúsculas (letra maiúscula, e não letra minúscula, como especificado na string de pesquisa).

Funções de Expressão Regular

- O Oracle fornece um conjunto de funções SQL que você pode usar para pesquisar e manipular strings com o apoio de expressões regulares.
- Você pode usar essas funções em qualquer tipo de dados que contenha caracteres, como CHAR, CLOB e VARCHAR2.
- Uma expressão regular deve estar entre aspas simples.

Funções de Expressão Regular

Nome	Descrição
REGEXP_LIKE	Semelhante ao operador LIKE, mas realiza correspondência de expressões regulares, em vez de correspondência de padrões simples
REGEXP_REPLACE	Pesquisa um padrão de expressão regular e o troca por uma string substituta
REGEXP_INSTR	Pesquisa uma determinada string para um padrão de expressão regular e retorna a posição onde a correspondência foi encontrada
REGEXP_SUBSTR	Pesquisa um padrão de expressão regular em uma determinada string e retorna a substring correspondente
REGEXP_COUNT	Retorna o número de vezes que um padrão aparece em uma string. Você especifica a string e o padrão. Você também pode especificar a posição inicial e as opções de correspondência (por exemplo, c para case sensitivity (distinção entre maiúsculas e minúsculas)).

Exemplos de Funções de Expressão Regular

- Suponha que você foi solicitado a listar todos os funcionários com o nome Stephen ou Steven.

```
SELECT first_name, last_name  
FROM employees  
WHERE REGEXP_LIKE(first_name, '^Ste(v|ph)en$');
```

- Com as expressões regulares, você poderia simplesmente usar a função REGEXP_LIKE e a string de pesquisa: '^Ste(v|ph)en\$'
 - "^" especifica o início da string que está sendo pesquisada
 - "S" maiúsculo, seguido por
 - "t" minúsculo, seguido por
 - "e" minúsculo, seguido por

Exemplos de Funções de Expressão Regular

- Com as expressões regulares, você poderia simplesmente usar a função REGEXP_LIKE e a string de pesquisa: '^Ste(v|ph)en\$'
 - "(" inicia uma subexpressão
 - "v" minúsculo
 - "|" especifica OR
 - "p" minúsculo seguido por "h" minúsculo
 - ")" encerra o grupo de opções,
 - "e" minúsculo
 - "n" minúsculo
 - "\$" especifica o final da string que está sendo pesquisada

Exemplos de Funções de Expressão Regular

- Exemplo:

```
SELECT first_name, last_name  
FROM employees  
WHERE REGEXP_LIKE(first_name, '^Ste(v|ph)en$');
```

- Resultado:

FIRST_NAME	LAST_NAME
Steven	King

Exemplos de Funções de Expressão Regular

- A função REPLACE de expressão regular substituirá um padrão de string por outro.
- Este exemplo busca um "H" seguido por qualquer vogal e os substitui por dois símbolos "**".

```
SELECT last_name, REGEXP_REPLACE(last_name, '^H(a|e|i|o|u)', '**')  
  AS "Name changed"  
FROM employees;
```

LAST_NAME	Name changed
Abel	Abel
Davies	Davies
...	...
Hartstein	**rtstein
Higgins	**ggins
...	...

Exemplos de Funções de Expressão Regular

- A função COUNT de expressão regular retorna o número de vezes que um padrão aparece em uma string.
- Este exemplo busca a subexpressão "ab":

```
SELECT country_name, REGEXP_COUNT(country_name, '(ab)')  
  AS "Count of 'ab'"  
FROM wf_countries  
WHERE REGEXP_COUNT(country_name, '(ab)')>0;
```

COUNTRY_NAME	Count of 'ab'
Republic of Zimbabwe	1
Arab Republic of Egypt	1
Great Socialist Peoples Libyan Arab Jamahiriya	1
Kingdom of Saudi Arabia	1
Syrian Arab Republic	1
Gabonese Republic	1
United Arab Emirates	1

Expressões Regulares em Constraints de Verificação

- As expressões regulares também poderiam ser usadas como parte do código da aplicação para garantir que somente dados válidos fossem armazenados no banco de dados.
- É possível incluir uma chamada a uma função de expressão regular em uma constraint CHECK, por exemplo.

Expressões Regulares em Constraints de Verificação

- Então, se quisesse garantir que os endereços de e-mail sem '@' não fossem capturados em uma tabela no seu banco de dados, você poderia simplesmente adicionar a seguinte constraint de verificação:

```
ALTER TABLE employees  
ADD CONSTRAINT email_addr_chk  
CHECK(REGEXP_LIKE(email, '@'));
```

- Isso garantiria que todos os endereços de e-mail incluíssem o símbolo "@".

Expressões Regulares em Constraints de Verificação

- Usando expressões regulares, você pode verificar o formato dos endereços de e-mail mais detalhadamente, para saber se são válidos.
- Um endereço de e-mail válido teria um ou mais caracteres além de @, seguido por um ou mais caracteres além de um . (ponto), seguido por um ou mais caracteres.

```
CREATE TABLE my_contacts  
(first_name VARCHAR2(15),  
last_name VARCHAR2(15),  
email VARCHAR2(30) CHECK(REGEXP_LIKE(email, '.*@.*\..*')));
```

Expressões Regulares em Constraints de Verificação

- Definições de sintaxe:

.+ significa um ou mais caracteres

@ o símbolo @

\. . (um **ponto**) (aqui, a barra invertida é um caractere de escape)

```
CREATE TABLE my_contacts
(first_name VARCHAR2(15),
last_name VARCHAR2(15),
email VARCHAR2(30) CHECK(REGEXP_LIKE(email, '.*@.*\..*'));
```

Terminologia

Estes são os principais termos usados nesta lição:

- Expressões Regulares
- Metacaracteres
- Subexpressões

Resumo

Nesta lição, você deverá ter aprendido a:

- Descrever expressões regulares
- Usar expressões regulares para pesquisar, relacionar e substituir strings em instruções SQL
- Construir e executar expressões regulares e constraints de verificação

