



Programação de Banco de Dados com SQL

13-2

Usando Tipos de Dados



Objetivos

Nesta lição, você aprenderá a:

- Criar uma tabela usando os tipos de dados de coluna `TIMESTAMP` e `TIMESTAMP WITH TIME ZONE`
- Criar uma tabela usando os tipos de dados de coluna `INTERVAL YEAR TO MONTH` e `INTERVAL DAY TO SECOND`
- Dar exemplos de organizações e situações pessoais em que é importante saber a que fuso horário um valor de data e hora se refere
- Listar e dar um exemplo de cada um dos tipos de dados de número, data e caracteres

Finalidade

- Se você algum dia viajar para outro país, logo descobrirá que o dinheiro no seu bolso pode não valer lá.
- Se você quiser comprar alguma coisa, será necessário converter seu dinheiro para a moeda local do país.
- Esse processo de conversão é muito parecido com o modo como se lida com tipos de dados em SQL.
- Tipos de dados diferentes têm tipos de características diferentes, cujo propósito é armazenar dados com eficiência.
- Nesta lição, você vai saber mais sobre os tipos de dados e seus usos.

Visão Geral de Tipos de Dados

- Todo valor manipulado pelo Oracle tem um tipo de dados.
- O tipo de dados de um valor associa a ele um conjunto fixo de propriedades.
- Essas propriedades fazem o banco de dados tratar os valores com um tipo de dados de maneira diferente dos valores com outro tipo.

Visão Geral de Tipos de Dados

- Tipos de dados diferentes oferecem muitas vantagens
 - Colunas com um único tipo produzem resultados consistentes.
 - Por exemplo, as colunas com o tipo de dados DATE sempre produzem valores de data.
 - Você não pode inserir o tipo de dados errado em uma coluna. Por exemplo, as colunas com o tipo de dados DATE impedirão que dados do tipo NUMBER sejam inseridos.
- Por essas razões, cada coluna em um banco de dados relacional pode conter apenas um tipo de dados.
- Você não pode misturar os tipos de dados de uma coluna.

Tipos de Dados Comuns

- Os tipos de dados de coluna mais usados para os valores numéricos e de caractere encontram-se a seguir.
- Para os valores de caractere:
 - CHAR (tamanho fixo; 2.000 caracteres, no máximo)
 - VARCHAR2 (tamanho variável; 4.000 caracteres, no máximo)
 - CLOB (tamanho variável; 128 terabytes, no máximo)
- Para valores numéricos:
 - NUMBER (tamanho variável; 38 dígitos de precisão máxima)

Tipos de Dados Comuns

- Os tipos de dados de coluna mais usados para os valores binários, de data e de hora encontram-se a seguir.
- Para valores de data e hora:
 - DATE
 - TIMESTAMP
 - INTERVAL
- Para valores binários (por exemplo, multimídia: JPG, WAV, MP3 etc.):
 - RAW (tamanho variável; 2.000 bytes, no máximo)
 - BLOB (tamanho variável; 128 terabytes, no máximo)

Tipos de Dados Comuns

- Para valores de caractere, é melhor usar VARCHAR2 ou CLOB, em vez de CHAR, porque economiza espaço.
- Por exemplo, o sobrenome de um funcionário é 'Chang'.
- Em uma coluna VARCHAR2(30), apenas os cinco caracteres significativos são armazenados: C h a n g.
- Mas em uma coluna CHAR(30), 25 espaços finais seriam armazenados também, para completar o tamanho fixo de 30 caracteres.
- Os valores numéricos podem ser positivos ou negativos. Por exemplo, NUMBER(6,2) pode armazenar qualquer número entre +9999.99 e -9999.99.

Tipos de Dados DATE-TIME

- O tipo de dados DATE armazena valores que vão de séculos a segundos inteiros, mas não frações de segundo.
- '21-Aug-2003 17:25:30' é um valor válido, mas '21-Aug-2003 17:25:30.255' não.
- O tipo de dados TIMESTAMP é uma extensão de DATE e permite frações de segundo.
- Por exemplo, TIMESTAMP(3) permite três dígitos após os segundos inteiros, tornando possível armazenar valores que vão até a milissegundos.

Tipos de Dados DATE-TIME

- Exemplo de TIMESTAMP:

```
CREATE TABLE time_ex1  
( exact_time TIMESTAMP);
```

```
INSERT INTO time_ex1  
VALUES ('10-Jun-2015 10:52:29.123456');
```

```
INSERT INTO time_ex1  
VALUES (SYSDATE);
```

```
INSERT INTO time_ex1  
VALUES (SYSTIMESTAMP);
```

```
SELECT *  
FROM time_ex1;
```

EXACT_TIME
10-JUN-15 10.52.29.123456 AM
16-JUL-15 08.17.08.000000 AM
16-JUL-15 08.17.16.610293 AM

TIMESTAMP...With [LOCAL] Time Zone

- Pense no valor de hora '17:30'. Ele claramente significa "cinco e meia da tarde".
- Mas em que fuso horário?
- É cinco e meia da tarde no horário da cidade de Nova York, ou de Pequim, ou de Istambul, ou... ?
- Para as organizações globalizadas de hoje em dia, que mantêm operações em muitos países diferentes, é importante saber a que fuso horário um valor de data e hora se refere.

TIMESTAMP...With [LOCAL] Time Zone

- `TIMESTAMP WITH TIME ZONE` armazena um valor de fuso horário como um deslocamento do Tempo Universal Coordenado, ou UTC (antes conhecido como Hora de Greenwich, ou GMT).
- O valor `'21-Aug-2003 08:00:00 -5:00'` significa 8 da manhã no fuso horário UTC-5.
- Essa é a Zona de Tempo Oriental (EST) dos EUA.

TIMESTAMP...With [LOCAL] Time Zone

- Exemplo de TIMESTAMP WITH TIME ZONE:

```
CREATE TABLE time_ex2  
( time_with_offset TIMESTAMP WITH TIME ZONE);
```

```
INSERT INTO time_ex2  
VALUES (SYSTIMESTAMP);
```

```
INSERT INTO time_ex2  
VALUES ('10-Jun-2015 10:52:29.123456 AM +2:00');
```

```
SELECT *  
FROM time_ex2;
```

TIME_WITH_OFFSET
16-JUL-15 08.49.47.126056 AM -07:00
10-JUN-15 10.52.29.123456 AM +02:00

TIMESTAMP...With [LOCAL] Time Zone

- TIMESTAMP WITH LOCAL TIME ZONE é semelhante, mas tem uma diferença: quando essa coluna é selecionada em uma instrução SQL, a hora é convertida automaticamente para o fuso horário do usuário que fez a seleção.
- Exemplo de TIMESTAMP With...Time Zone:

```
CREATE TABLE time_ex3
( first_column TIMESTAMP WITH TIME ZONE,
  second_column TIMESTAMP WITH LOCAL TIME ZONE);
```

```
INSERT INTO time_ex3
  (first_column, second_column)
VALUES
  ('15-Jul-2015 08:00:00 AM -07:00', '15-Nov-2007 08:00:00');
```

Exemplo de TIMESTAMP...With Time Zone

- Ambos os valores são armazenados com um deslocamento de -7 horas (PST).
- Mas agora um usuário em Istambul executa:

```
SELECT *  
FROM time_ex3;
```

FIRST_COLUMN	SECOND_COLUMN
15-JUL-15 08.00.00.000000 AM -07:00	15-NOV-07 05.00.00.000000 PM

- O horário de Istambul está 9 horas à frente do PST. Quando são 8 horas da manhã em Los Angeles, são 5 horas da tarde em Istambul.

Tipos de Dados INTERVAL

- Eles armazenam o tempo decorrido, ou intervalo de tempo, entre dois valores de data e hora.
- INTERVAL YEAR TO MONTH armazena um período medido em anos e meses.
- INTERVAL DAY TO SECOND armazena um período medido em dias, horas, minutos e segundos.



INTERVAL YEAR...TO MONTH

- Sintaxe

```
INTERVAL YEAR [(precisão_ano)] TO MONTH
```

- `precisão_ano` é a quantidade máxima de dígitos no elemento YEAR.
- O valor padrão de `precisão_ano` é 2.



INTERVAL YEAR...TO MONTH

- Este exemplo mostra INTERVAL YEAR TO MONTH

```
CREATE TABLE time_ex4  
(loan_duration1 INTERVAL YEAR(3) TO MONTH,  
 loan_duration2 INTERVAL YEAR(2) TO MONTH);
```

```
INSERT INTO time_ex4 (loan_duration1, loan_duration2)  
VALUES (INTERVAL '120' MONTH(3),  
        INTERVAL '3-6' YEAR TO MONTH);
```

Suponha que a data de hoje é: 17-Jul-2017

```
SELECT SYSDATE + loan_duration1 AS "120 months from now",  
       SYSDATE + loan_duration2 AS "3 years 6 months from now"  
FROM time_ex4;
```

120 months from now	3 years 6 months from now
17-Jul-2027	17-Jan-2021

INTERVAL DAY...TO SECOND

- Use este tipo de dados quando precisar definir uma diferença mais precisa entre dois valores de data e hora.
- Sintaxe:

```
INTERVAL DAY [precisão_dia)] TO SECOND [(precisão_segundos_fracionais)]
```

- `precisão_dia` é a quantidade máxima de dígitos no elemento DAY.
- O valor padrão de `precisão_dia` é 2.
- `precisão_segundos_fracionais` é a quantidade de dígitos na parte fracional do campo de data e hora SECOND.
- O padrão é 6.

INTERVAL DAY...TO SECOND

- Este exemplo mostra INTERVAL DAY TO SECOND

```
CREATE TABLE time_ex5  
(day_duration1 INTERVAL DAY(3) TO SECOND,  
 day_duration2 INTERVAL DAY(3) TO SECOND);
```

```
INSERT INTO time_ex5 (day_duration1, day_duration2)  
VALUES (INTERVAL '25' DAY(2), INTERVAL '4 10:30:10' DAY TO SECOND);
```

```
SELECT SYSDATE + day_duration1 AS "25 Days from now",  
       TO_CHAR(SYSDATE + day_duration2, 'dd-Mon-yyyy hh:mi:ss')  
       AS "precise days and time from now"  
FROM time_ex5;
```

25 Days from now	precise days and time from now
11-Aug-2015	21-Jul-2015 01:13:17

Terminologia

Estes são os principais termos usados nesta lição:

- CLOB
- BLOB
- TIMESTAMP
- TIMESTAMP WITH TIMEZONE
- TIMESTAMP WITH LOCAL TIMEZONE
- INTERVAL DAY TO SECOND
- INTERVAL DAY TO MONTH

Resumo

Nesta lição, você aprendeu a:

- Criar uma tabela usando os tipos de dados de coluna `TIMESTAMP` e `TIMESTAMP WITH TIME ZONE`
- Criar uma tabela usando os tipos de dados de coluna `INTERVAL YEAR TO MONTH` e `INTERVAL DAY TO SECOND`
- Dar exemplos de organizações e situações pessoais em que é importante saber a que fuso horário um valor de data e hora se refere
- Listar e dar um exemplo de cada um dos tipos de dados de número, data e caracteres

