

ORACLE

Academy

Database Programming with SQL

16-1

Trabalhando com Sequências

ORACLE
Academy



Objetivos

- Esta lição abrange os seguintes objetivos:
 - Listar pelo menos três características úteis de uma sequência
 - Gravar e executar uma instrução SQL que crie uma sequência
 - Consultar o dicionário de dados usando `USER_SEQUENCES` para confirmar uma definição de sequência
 - Aplicar as regras para usar `NEXTVAL` e gerar números sequenciais exclusivos em uma tabela
 - Listar as vantagens e desvantagens de armazenar no cache os valores da sequência
 - Dizer três motivos pelos quais pode haver lacunas em uma sequência

Finalidade

- Já pensou como seria entediante ter de incluir em um banco de dados os nomes das 30.000 pessoas que se inscreveram na Maratona de Londres e garantir que ninguém recebesse o mesmo número de identificação?
- E se você fosse almoçar e, quando voltasse, outra pessoa tivesse incluído algumas das inscrições dos atletas?
- Como você saberia por onde recomeçar?

Finalidade

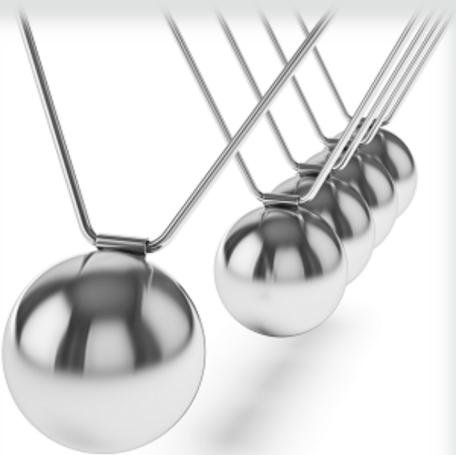
- Felizmente, a linguagem SQL tem um processo para gerar automaticamente números exclusivos, que elimina a preocupação com os detalhes de números duplicados
- O processo de numeração é tratado por meio de um objeto de banco de dados chamado SEQUENCE

O Objeto de Sequência

- Você já sabe como criar dois tipos de objetos de banco de dados: TABLE e VIEW
- O terceiro objeto de banco de dados é SEQUENCE
- SEQUENCE é um objeto compartilhável usado para gerar automaticamente números exclusivos
- Por se tratar de um objeto compartilhável, vários usuários podem acessá-lo
- Normalmente, as sequências são usadas para criar um valor de chave primária

O Objeto de Sequência

- Como você se lembra, as chaves primárias devem ser exclusivas de cada linha. A sequência é gerada e incrementada (ou diminuída) por uma rotina Oracle interna
- Esse objeto economiza tempo porque reduz a quantidade de códigos que você precisa gravar



O Objeto de Sequência

- Os números da sequência são armazenados e gerados de maneira independente às tabelas
- Por isso, a mesma sequência pode ser usada para várias tabelas
- Para criar uma sequência:

```
CREATE SEQUENCE sequence
    [INCREMENT BY n]
    [START WITH n]
    [{MAXVALUE n | NOMAXVALUE} ]
    [{MINVALUE n | NOMINVALUE} ]
    [{CYCLE | NOCYCLE} ]
    [{CACHE n | NOCACHE} ];
```

Sintaxe de Sequência

```
CREATE SEQUENCE sequence
    [INCREMENT BY n]
    [START WITH n]
    [{MAXVALUE n | NOMAXVALUE}]
    [{MINVALUE n | NOMINVALUE}]
    [{CYCLE | NOCYCLE}]
    [{CACHE n | NOCACHE}];
```

Sequence (Sequência)	é o nome do gerador de sequência (objeto)
INCREMENT BY n	especifica o intervalo entre os números da sequência, onde n é um número inteiro (se esta cláusula for omitida, a sequência incrementa-se em 1)
START WITH n	especifica o primeiro número da sequência a ser gerado (se esta cláusula for omitida, a sequência começa com 1)

Sintaxe de Sequência

```
CREATE SEQUENCE sequence  
    [INCREMENT BY n]  
    [START WITH n]  
    [ {MAXVALUE n | NOMAXVALUE} ]  
    [ {MINVALUE n | NOMINVALUE} ]  
    [ {CYCLE | NOCYCLE} ]  
    [ {CACHE n | NOCACHE} ] ;
```

MAXVALUE n	especifica o valor máximo que a sequência pode gerar
NOMAXVALUE	especifica um valor máximo de 10^{27} para uma sequência crescente e -1 para uma sequência decrescente (padrão)
MINVALUE n	especifica o valor mínimo da sequência

Sintaxe de Sequência

```
CREATE SEQUENCE sequence  
    [INCREMENT BY n]  
    [START WITH n]  
    [{MAXVALUE n | NOMAXVALUE}]  
    [{MINVALUE n | NOMINVALUE}]  
    [{CYCLE | NOCYCLE}]  
    [{CACHE n | NOCACHE}];
```

NOMINVALUE	especifica um valor mínimo de 1 para uma sequência crescente e -(10^26) para uma sequência decrescente (padrão)
CYCLE NOCYCLE	especifica se a sequência continua gerando valores após atingir seu valor máximo ou mínimo (NOCYCLE é a opção padrão)

Sintaxe de Sequência

```
CREATE SEQUENCE sequence
    [INCREMENT BY n]
    [START WITH n]
    [{MAXVALUE n | NOMAXVALUE} ]
    [{MINVALUE n | NOMINVALUE} ]
    [{CYCLE | NOCYCLE} ]
    [{CACHE n | NOCACHE} ] ;
```

CACHE n | NOCACHE

especifica quantos valores o servidor Oracle pré-aloca e mantém na memória (por padrão o servidor Oracle armazena 20 valores no cache). Se houver uma falha no sistema, os valores serão perdidos.

Criando uma Sequência

- Na sequência criada para os atletas da Maratona de Londres, os números vão se incrementar em 1, começando pelo número 1
- Nesse caso, começar a sequência com 1 provavelmente é o melhor ponto de partida

```
CREATE SEQUENCE runner_id_seq
  INCREMENT BY 1
  START WITH 1
  MAXVALUE 50000
  NOCACHE
  NOCYCLE;
```

Criando uma Sequência

- É uma tradição que o melhor atleta do grupo de elite use número 1
- Em outros casos, como IDs de departamentos e IDs de funcionários, o número inicial pode ser atribuído de maneira diferente
- Como haverá pelo menos 30.000 atletas, o valor máximo da sequência foi definido muito acima do número esperado de atletas

```
CREATE SEQUENCE runner_id_seq
    INCREMENT BY 1
    START WITH 1
    MAXVALUE 50000
    NOCACHE
    NOCYCLE;
```

Criando uma Sequência

- A opção NOCACHE impede que os valores na sequência sejam armazenados em cache na memória, o que, caso ocorra uma falha no sistema, evita que os números pré-alocados e mantidos na memória sejam perdidos

```
CREATE SEQUENCE runner_id_seq
INCREMENT BY 1
START WITH 1
MAXVALUE 50000
NOCACHE
NOCYCLE;
```

Criando uma Sequência

- A opção NOCYCLE impede que a numeração recomece a partir do 1 se o valor 50.000 for excedido
- Não use a opção CYCLE se a sequência for usada para gerar valores de chave primária, a não ser que haja um mecanismo confiável que exclua linhas antigas mais rápido do que as novas são adicionadas

```
CREATE SEQUENCE runner_id_seq
INCREMENT BY 1
START WITH 1
MAXVALUE 50000
NOCACHE
NOCYCLE;
```

Confirmando Sequências

- Para verificar se uma sequência foi criada, consulte `USER_OBJECTS` no dicionário de dados
- Para ver todas as configurações de `SEQUENCE`, consulte `USER_SEQUENCES` no dicionário de dados, como mostrado abaixo
- Liste os nomes de valor na instrução `SELECT`, como mostrado abaixo

```
SELECT sequence_name, min_value, max_value, increment_by,  
last_number  
FROM user_sequences;
```

Confirmando Sequências

- Se NOCACHE for especificado, a coluna last_number na consulta exibirá o próximo número disponível da sequência
- Se CACHE for especificado, a coluna last_number exibirá o próximo número disponível na sequência, que não foi armazenado em cache na memória

```
SELECT sequence_name, min_value, max_value, increment_by,  
last_number  
FROM user_sequences;
```

Pseudocolunas NEXTVAL e CURRVAL

- A pseudocoluna NEXTVAL é usada para extrair números sucessivos de uma sequência especificada
- Você deve qualificar NEXTVAL com o nome da sequência
- Quando você referencia sequence.NEXTVAL, um novo número da sequência é gerado e o número atual é colocado em CURRVAL

Pseudocolunas NEXTVAL e CURRVAL

- O exemplo abaixo insere um novo departamento na tabela DEPARTMENTS
- Ele usa a sequência DEPARTMENTS_SEQ para gerar um novo número de departamento, da seguinte forma:

```
INSERT INTO departments
    (department_id, department_name, location_id)
VALUES  (departments_seq.NEXTVAL, 'Support', 2500);
```

Pseudocolunas NEXTVAL e CURRVAL

- Vamos supor que agora você queira contratar funcionários para o novo departamento
- A instrução INSERT a ser executada para todos os funcionários novos pode incluir o seguinte código:

```
INSERT INTO employees
    (employee_id, department_id, ...)
VALUES (employees_seq.NEXTVAL, dept_deptid_seq .CURRVAL,
...);
```

- Observação: o exemplo acima supõe que uma sequência chamada EMPLOYEES_SEQ já foi criada para gerar novos números de funcionário

Pseudocolunas NEXTVAL e CURRVAL

- A pseudocoluna CURRVAL no exemplo abaixo é usada para se referir a um número da sequência que o usuário atual acabou de gerar
- NEXTVAL deve ser usado para gerar um número da sequência na sessão do usuário atual antes que CURRVAL possa ser referenciado
- Você deve qualificar CURRVAL com o nome da sequência

Pseudocolunas NEXTVAL e CURRVAL

- Quando sequence.CURRVAL for referenciado, o último valor gerado pelo processo do usuário será retornado

```
INSERT INTO employees
    (employee_id, department_id, ...)
VALUES (employees_seq.NEXTVAL, dept_deptid_seq.CURRVAL, ...);
```

Usando uma Sequência

- Depois que você cria uma sequência, ela gera números sequenciais para serem usados nas suas tabelas
Referencie os valores da sequência usando as pseudocolunas NEXTVAL e CURRVAL
- Você pode usar NEXTVAL e CURRVAL nos seguintes contextos:
 - A lista SELECT de uma instrução SELECT que não faça parte de uma subconsulta
 - A lista SELECT de uma subconsulta em uma instrução INSERT
 - A cláusula VALUES de uma instrução INSERT
 - A cláusula SET de uma instrução UPDATE

Usando uma Sequência

- Você não pode usar NEXTVAL e CURRVAL nos seguintes contextos:
 - A lista SELECT de uma view
 - Uma instrução SELECT com a palavra-chave DISTINCT
 - Uma instrução SELECT com as cláusulas GROUP BY, HAVING ou ORDER BY
 - Uma subconsulta em uma instrução SELECT, DELETE ou UPDATE
 - A expressão DEFAULT em uma instrução CREATE TABLE ou ALTER TABLE

Usando uma Sequência

- Para continuar com o exemplo da Maratona de Londres, foi criada uma tabela para os atletas:

```
CREATE TABLE runners
(runner_id NUMBER(6,0) CONSTRAINT runners_id_pk PRIMARY KEY,
 first_name VARCHAR2(30),
 last_name VARCHAR2(30));
```

Usando uma Sequência

- Em seguida, criamos a sequência que gerará valores para a coluna da chave primária runner_id

```
CREATE SEQUENCE runner_id_seq
  INCREMENT BY 1
  START WITH 1
  MAXVALUE 50000
  NOCACHE
  NOCYCLE ;
```

Usando uma Sequência

- O uso da sintaxe abaixo permitiria que novos participantes fossem inseridos na tabela de atletas
- O número de identificação do atleta seria gerado com a recuperação de NEXTVAL da sequência

```
INSERT INTO runners
    (runner_id, first_name, last_name)
VALUES  (runner_id_seq.NEXTVAL, 'Joanne', 'Everely');
```

```
INSERT INTO runners
    (runner_id, first_name, last_name)
VALUES  (runner_id_seq.NEXTVAL, 'Adam', 'Curtis');
```

Usando uma Sequência

- Para confirmar se a sequência funcionou corretamente, consultamos a tabela:

```
SELECT runner_id, first_name, last_name  
FROM runners;
```

RUNNER_ID	FIRST_NAME	LAST_NAME
1	Joanne	Everely
2	Adam	Curtis

Usando uma Sequência

- Para exibir o valor atual para runners_id_seq, CURRVAL é usado
- Observe o uso da tabela DUAL neste exemplo
- O Oracle Application Express não executará esta consulta, mas você deve entender como ela funciona

```
SELECT runner_id_seq.CURRVAL  
FROM dual;
```

Usando uma Sequência

- O armazenamento em cache das sequências na memória fornece acesso mais rápido aos valores da sequência
- O cache é preenchido na primeira vez que você se refere à sequência
- Cada solicitação do próximo valor da sequência é recuperado da sequência no cache
- Depois que o último valor da sequência é usado, a próxima solicitação pela sequência traz outro cache de sequências para a memória
- 20 é a quantidade padrão de números de sequência no cache

Números Não Sequenciais

- Embora os geradores de sequência emitam números sequenciais sem lacunas, essa ação ocorre de modo independente a um commit ou rollback de banco de dados
- Lacunas (números não sequenciais) podem ser geradas por:
 - Rollback de uma instrução que contenha uma sequência; o número é perdido
 - Uma falha no sistema. Se a sequência armazena em cache os valores na memória e ocorre uma falha no sistema, esses valores são perdidos
 - A mesma sequência ser usada para várias tabelas. Se você fizer isso, cada tabela poderá conter lacunas nos números sequenciais

Exibindo o Próximo Valor

- Se a sequência foi criada com NOCACHE, é possível exibir o próximo valor disponível, sem incrementá-lo, consultando a tabela USER_SEQUENCES

```
SELECT sequence_name, min_value, max_value, last_number AS  
"Next number"  
FROM USER_SEQUENCES  
WHERE sequence_name = 'RUNNER_ID_SEQ';
```

SEQUENCE_NAME	MIN_VALUE	MAX_VALUE	Next number
RUNNER_ID_SEQ	1	50000	3

Modificando uma Sequência

- Assim como com os outros objetos de banco de dados que você criou, uma sequência também pode ser alterada usando a instrução ALTER SEQUENCE
- E se mais de 50.000 atletas tivessem se inscrito na Maratona de Londres e você precisasse adicionar mais números?
- A sequência poderia ser alterada para aumentar o MAXVALUE sem modificar a ordem numérica existente

```
ALTER SEQUENCE runner_id_seq
  INCREMENT BY 1
  MAXVALUE 999999
  NOCACHE
  NOCYCLE ;
```

Modificando uma Sequência

- Uma validação é realizada quando você altera uma sequência
- Por exemplo, um novo MAXVALUE inferior ao número atual da sequência não pode ser executado

```
ALTER SEQUENCE runner_id_seq
    INCREMENT BY 1
    MAXVALUE 90
    NOCACHE
    NOCYCLE ;
```

ERROR at line 1:

ORA-04009: MAXVALUE cannot be made to be less than the current value
(Erro na linha 1: ORA-04009: MAXVALUE não pode ser definido como
inferior ao valor atual)

Diretrizes de ALTER SEQUENCE

- Algumas diretrizes se aplicam quando se executa uma instrução ALTER SEQUENCE
- São elas:
 - Para modificar a sequência, você deve ser o proprietário ou ter o privilégio ALTER para ela
 - Somente os números futuros da sequência são afetados pela instrução ALTER SEQUENCE
 - A opção START WITH não pode ser modificada usando ALTER SEQUENCE. A sequência deve ser eliminada e recriada para recomeçar em um número diferente

Removendo uma Sequência

- Para remover uma sequência do dicionário de dados, use a instrução DROP SEQUENCE
- Você deve ser o proprietário da sequência ou ter privilégios DROP ANY SEQUENCE para removê-la
- Após removida, a sequência não pode mais ser referenciada

```
DROP SEQUENCE runner_id_seq;
```

Terminologia

- Estes são os principais termos usados nesta lição:
 - CACHE/NOCACHE
 - CREATE SEQUENCE
 - CURRVAL
 - CYCLE/NOCYCLE
 - INCREMENT BY
 - MAXVALUE
 - MINVALUE

Terminologia

- Estes são os principais termos usados nesta lição:
 - NEXTVAL
 - NOMAXVALUE
 - NOMINVALUE
 - Sequências
 - START WITH

Resumo

- Nesta lição, você deverá ter aprendido a:
 - Listar pelo menos três características úteis de uma sequência
 - Gravar e executar uma instrução SQL que crie uma sequência
 - Consultar o dicionário de dados usando `USER_SEQUENCES` para confirmar uma definição de sequência
 - Aplicar as regras para usar `NEXTVAL` e gerar números sequenciais exclusivos em uma tabela
 - Listar as vantagens e desvantagens de armazenar no cache os valores da sequência
 - Dizer três motivos pelos quais pode haver lacunas em uma sequência

ORACLE

Academy