



ORACLE

Academy



Database Foundations

6-5

Linguagem de Controle de Transação (TCL)

ORACLE
Academy



Roteiro

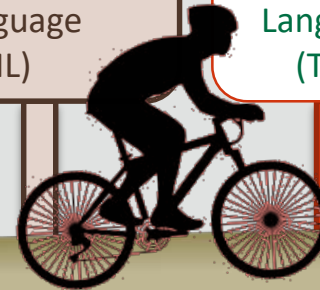
Introdução ao
Oracle
Application
Express (APEX)

Structured
Query
Language
(SQL)

Data
Definition
Language
(DDL)

Data
Manipulation
Language
(DML)

Transaction
Control
Language
(TCL)



Recuperand
o Dados com
SELECT

Restringindo
Dados com
WHERE

Classificando
Dados com
ORDER BY

Unindo Tabelas
com JOIN

Parte 6

Objetivos

- Esta lição abrange os seguintes objetivos:
 - Descrever a finalidade do TCL (Transaction Control Language)
 - Explicar as operações TCL que são necessárias para gerenciar uma transação:
 - COMMIT
 - SAVEPOINT
 - ROLLBACK
 - Descrever a necessidade de consistência de leitura



Application Express

- COMMIT, ROLLBACK e SAVEPOINT não são suportados no Oracle Application Express devido à maneira como ele gerencia as conexões com o banco de dados



Oracle APEX

Transações de Banco de Dados

- Transações consistem em instruções DML que representam uma alteração consistente nos dados
- O servidor Oracle garante a consistência dos dados com base em transações
- As transações proporcionam mais flexibilidade e controle ao alterar dados, bem como garantem a consistência dos dados em caso de falhas no processo do usuário ou do sistema

Transações de Banco de Dados

- Por exemplo, uma transferência de fundos entre duas contas deve incluir o débito em uma conta e o crédito do mesmo valor na outra. As duas ações devem falhar ou ter êxito juntas; o crédito não deve ser confirmado (commit) sem o débito.

Transações de Banco de Dados

- Uma transação de banco de dados consiste em uma das seguintes instruções:
 - Instruções DML que representam uma alteração consistente nos dados
 - Uma instrução DDL
 - Uma instrução TCL

Transações de Banco de Dados: Início e Fim

- Uma transação inicia quando a primeira instrução DML SQL é executada
- Ela termina com um dos seguintes eventos:
 - Uma instrução COMMIT ou ROLLBACK é emitida
 - Uma instrução DDL ou TCL é executada (commit automático)
 - O usuário sai do software SQL que está sendo usado
 - O sistema trava

Vantagens das Instruções COMMIT e ROLLBACK

- Com instruções COMMIT e ROLLBACK, você pode:
 - Garantir a consistência dos dados
 - Visualizar alterações nos dados antes de torná-las permanentes
 - Agrupar operações logicamente relacionadas
 - Ter controle permanente sobre as alterações nos dados

Instruções de Controle de Transações

Instrução	Descrição
COMMIT	Finaliza a transação atual tornando todas as alterações pendentes nos dados permanentes.
SAVEPOINT nome	Marca um ponto de salvamento na transação atual.
ROLLBACK	Finaliza a transação atual descartando todas as alterações pendentes nos dados.
ROLLBACK TO SAVEPOINT nome	Faz rollback da transação atual para o ponto de salvamento especificado, descartando todas as alterações e/ou pontos de salvamento que foram criados após o ponto de salvamento para o qual você está fazendo rollback. Se você omitir a cláusula TO SAVEPOINT, a instrução ROLLBACK fará rollback de toda a transação. Como os pontos de salvamento são lógicos, não há como listar os que você criou.

Instruções de Controle de Transações Explícitas

Tempo

COMMIT

Transação

```
DELETE FROM copy_employees;
```

SAVEPOINT A

```
INSERT INTO departments  
VALUES (80, 'Marketing', 400, 1900);
```

```
UPDATE employees  
SET dept_id = 110;
```

SAVEPOINT B

```
INSERT ...
```

ROLLBACK

to SAVEPOINT B

ROLLBACK

to SAVEPOINT A

ROLLBACK



Fazendo Rollback de Alterações para um Marcador

- Crie um marcador na transação atual usando a instrução **SAVEPOINT**
- Descarte as alterações pendentes fazendo rollback para esse marcador com a instrução **ROLLBACK TO SAVEPOINT**

```
UPDATE...  
SAVEPOINT update_done;
```

```
SAVEPOINT update_done succeeded.
```

```
INSERT...  
ROLLBACK TO update_done;
```

```
ROLLBACK TO succeeded.
```

Processamento Implícito de Transações

- Um commit automático ocorre nas seguintes circunstâncias:
 - Uma instrução DDL é emitida
 - Uma instrução TCL é emitida
 - Há um encerramento normal do software SQL, sem a emissão explícita das instruções COMMIT ou ROLLBACK
- Um rollback automático ocorre quando há um encerramento anormal do software SQL ou quando há uma falha no sistema para proteger a integridade do banco de dados

Estado dos Dados Antes de COMMIT ou ROLLBACK

- Toda alteração de dados feita durante uma transação será temporária se não for confirmada (commit); portanto, o estado anterior dos dados poderá ser recuperado
- A sessão atual pode examinar os resultados das operações DML usando a instrução SELECT
- Outras sessões não podem exibir os resultados das instruções DML emitidas pela sessão atual
- As linhas afetadas são bloqueadas; outras sessões não podem alterar os dados nas linhas afetadas

Estado dos Dados Depois de COMMIT

- As alterações nos dados são salvas no banco de dados
- O estado anterior dos dados é substituído
- Todas as sessões podem exibir os resultados
- Os bloqueios nas linhas afetadas são liberados; essas linhas estão disponíveis para manipulação por outras sessões
- Todos os pontos de salvamento são apagados

Fazendo Commit dos Dados

- Faça as alterações:

```
DELETE  
FROM   copy_employees  
WHERE  employee_id=113;
```

```
1 row(s) deleted.
```

```
INSERT INTO copy_departments  
VALUES (290, 'Corporate Tax', NULL, 1700);
```

```
1 row(s) inserted.
```

- Confirme (commit) as alterações:

```
COMMIT;
```

Estado dos Dados Depois de ROLLBACK

- Descarte todas as alterações pendentes usando a instrução ROLLBACK:
 - As alterações dos dados são desfeitas
 - O estado anterior dos dados é restaurado
 - Os bloqueios nas linhas afetadas são liberados

```
DELETE FROM copy_employees;  
ROLLBACK ;
```

Estado dos Dados Depois de ROLLBACK: Exemplo

- Considere uma tabela de teste contendo quatro registros:

```
DELETE FROM test;
```

4 linhas excluídas

```
ROLLBACK;
```

Rollback concluído

```
DELETE FROM test WHERE id = 100;
```

1 linha excluída

```
SELECT * FROM test WHERE id = 100;
```

Nenhuma linha selecionada

```
COMMIT;
```

Commit concluído



Rollback no Nível de Instrução

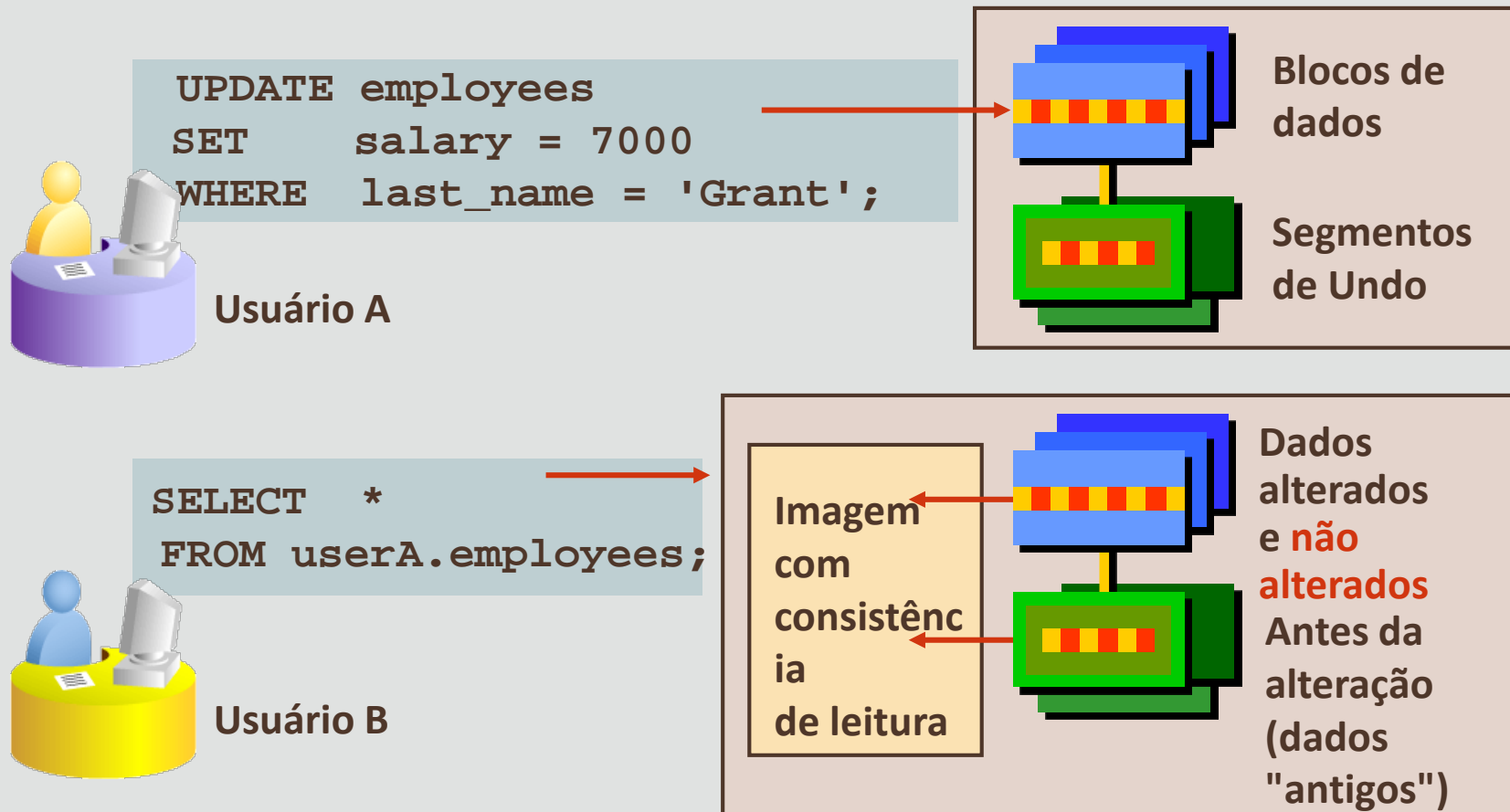
- Se uma única instrução DML falhar durante a execução, será feito rollback somente dessa instrução
- O servidor Oracle implementa um ponto de salvamento implícito
- Todas as outras alterações são mantidas
- O usuário deve encerrar as transações explicitamente executando uma instrução COMMIT ou ROLLBACK



Consistência de Leitura

- A consistência de leitura garante uma exibição consistente dos dados em todos os momentos – cada usuário vê os dados conforme existiam desde o último COMMIT
- As alterações feitas por um usuário não entram em conflito com as alterações feitas por outro
- A consistência de leitura garante que, nos mesmos dados:
 - Os leitores não aguardem os gravadores
 - Os gravadores não aguardem os leitores
 - Os gravadores aguardem outros gravadores

Implementando a Consistência de Leitura



Observação: para obter uma explicação mais detalhada da consistência de leitura, leia as anotações do slide

Resumo

- Nesta lição, você deverá ter aprendido a:
 - Descrever a finalidade do TCL (Transaction Control Language)
 - Explicar as operações TCL que são necessárias para gerenciar uma transação:
 - COMMIT
 - SAVEPOINT
 - ROLLBACK
 - Descrever a necessidade de consistência de leitura





ORACLE

Academy

