

Programação de Banco de Dados com SQL

18-1: Transações de Banco de Dados

Atividades Práticas

Objetivos

- Definir as instruções COMMIT, ROLLBACK e SAVEPOINT com base em sua relação com as transações de dados
- Listar três vantagens das instruções COMMIT, ROLLBACK e SAVEPOINT
- Explicar por que é importante, da perspectiva de negócios, ser capaz de controlar o fluxo de processamento de transações

Vocabulário

Identifique a palavra do vocabulário correspondente a cada definição a seguir.

	Termina a transação atual, tornando todas as alterações de dados pendentes permanentes
	Permite que o usuário descarte as alterações feitas no banco de dados
	Cria um marcador em uma transação, que a divide em partes menores
	Garante uma exibição consistente dos dados para todos os usuários a qualquer momento
	Mecanismos que impedem a interação destrutiva entre transações que acessam o mesmo recurso que pode ser concedido ao usuário
	Um conjunto de instruções DML que constituem uma unidade lógica de trabalho

Tente/solucione

1. Defina as instruções COMMIT, ROLLBACK e SAVEPOINT em termos de sua relação com as transações de dados.

2. Quais dados serão submetidos a commit após a execução das seguintes instruções?

```
INSERT INTO R values (5, 6);  
SAVEPOINT my_savepoint_1;  
INSERT INTO R values (7, 8);  
SAVEPOINT my_savepoint_2;  
INSERT INTO R values (9, 10);  
ROLLBACK TO my_savepoint_1;  
INSERT INTO R values (11, 12);  
COMMIT;
```

3. Crie uma instrução SQL para a tabela D_SONGS da DJs on Demand que exclua a música “All These Years”, insira uma nova música Country chamada ‘Happy Birthday Sunshine’ do “The Sunsets”, com uma duração de 4 minutos e o ID = 60. Verifique se todos os dados podem ser recuperados antes de serem feitas alterações na tabela.
4. Crie uma instrução SQL que executará um commit automático.
5. Dê dois exemplos de negócios, diferente de bancos, que dependem de processos de controle de transações. Descreva por que cada um deles precisa desse controle.