

Soluções para Questões de Triggers, Views, e Subconsultas

1. Crie um trigger que impeça a inserção de livros com o mesmo Título na tabela Livro.

```
Solução: CREATE TRIGGER trg_no_duplicate_title ON Livro
AFTER INSERT
AS
BEGIN
    IF EXISTS (SELECT * FROM Livro WHERE Titulo IN (SELECT Titulo FROM
INSERTED))
    BEGIN
        RAISEERROR('Não é permitido inserir livros com o mesmo título.', 16, 1);
        ROLLBACK TRANSACTION;
    END
END
```

2. Crie um trigger que atualize automaticamente o ano de publicação na tabela Livro para o ano atual quando um novo livro é inserido.

```
Solução: CREATE TRIGGER trg_update_year ON Livro
AFTER INSERT
AS
BEGIN
    UPDATE Livro SET AnoPublicacao = YEAR(GETDATE()) WHERE LivroID IN (SELECT
LivroID FROM INSERTED);
END
```

3. Crie um trigger que exclua automaticamente registros da tabela LivroAutor quando o livro correspondente é excluído da tabela Livro.

```
Solução: CREATE TRIGGER trg_delete_livroautor ON Livro
AFTER DELETE
AS
BEGIN
    DELETE FROM LivroAutor WHERE LivroID IN (SELECT LivroID FROM DELETED);
END
```

4. Crie um trigger que atualize o número total de livros em uma categoria específica na tabela Categoria sempre que um novo livro é inserido nessa categoria.

```
Solução: CREATE TRIGGER trg_update_category_total ON Livro
AFTER INSERT
AS
```

```

BEGIN
    UPDATE Categoria SET TotalLivros = (SELECT COUNT(*) FROM Livro WHERE
CategoriaID = INSERTED.CategoriaID)
    WHERE CategoriaID = (SELECT CategoriaID FROM INSERTED);
END

```

5. Crie um trigger que restrinja a exclusão de categorias na tabela Categoria se houver livros associados a essa categoria.

Solução: CREATE TRIGGER trg_restrict_delete_categoria ON Categoria
INSTEAD OF DELETE
AS
BEGIN
 IF EXISTS (SELECT * FROM Livro WHERE CategoriaID IN (SELECT CategoriaID
FROM DELETED))
 BEGIN
 RAISEERROR('Não é possível excluir categoria com livros associados.', 16,
1);
 END
 ELSE
 BEGIN
 DELETE FROM Categoria WHERE CategoriaID IN (SELECT CategoriaID
FROM DELETED);
 END
END

6. Crie um trigger que registre em uma tabela de auditoria sempre que um livro for atualizado na tabela Livro.

Solução: CREATE TRIGGER trg_audit_update ON Livro
AFTER UPDATE
AS
BEGIN
 INSERT INTO AuditLog (LivroID, DataHora, Acao) VALUES (SELECT LivroID,
GETDATE(), 'Atualização')
 FROM INSERTED;
END

7. Crie um trigger que calcule automaticamente o número total de livros escritos por um autor na tabela Autor sempre que um novo livro é associado a esse autor na tabela LivroAutor.

Solução: CREATE TRIGGER trg_update_author_total ON LivroAutor
AFTER INSERT
AS
BEGIN

```

UPDATE Autor SET TotalLivros = (SELECT COUNT(*) FROM LivroAutor WHERE
AutorID = INSERTED.AutorID)
WHERE AutorID = (SELECT AutorID FROM INSERTED);
END

```

8. Crie um trigger que restrinja a atualização do ISBN na tabela Livro para impedir que ele seja alterado.

Solução: CREATE TRIGGER trg_restrict_isbn_update ON Livro
 INSTEAD OF UPDATE
 AS
 BEGIN
 IF UPDATE(ISBN)
 BEGIN
 RAISEERROR('Alteração de ISBN não permitida.', 16, 1);
 END
 ELSE
 BEGIN
 UPDATE Livro SET ... WHERE LivroID IN (SELECT LivroID FROM
 INSERTED);
 END
 END

9. Crie um trigger que limite o número de livros escritos por um autor na tabela LivroAutor para um máximo de 5 livros por autor.

Solução: CREATE TRIGGER trg_limit_books_per_author ON LivroAutor
 BEFORE INSERT
 AS
 BEGIN
 IF (SELECT COUNT(*) FROM LivroAutor WHERE AutorID = INSERTED.AutorID) >=
 5
 BEGIN
 RAISEERROR('Autor já possui o máximo de 5 livros.', 16, 1);
 ROLLBACK TRANSACTION;
 END
 END

10. Crie um trigger que atualize automaticamente o campo total_livros na tabela Categoria sempre que um novo livro daquela categoria for inserido na tabela Livro.

Solução: CREATE TRIGGER trg_update_total_books ON Livro
 AFTER INSERT
 AS
 BEGIN

```
UPDATE Categoria SET total_livros = (SELECT COUNT(*) FROM Livro WHERE
CategoriaID = INSERTED.CategoriaID)
WHERE CategoriaID = (SELECT CategoriaID FROM INSERTED);
END
```

11. Crie um trigger que, quando um livro for excluído da tabela Livro, verifique se há outros livros escritos pelo mesmo autor e, se não, remova automaticamente o autor da tabela Autor.

Solução: CREATE TRIGGER trg_delete_author_if_no_books ON Livro
AFTER DELETE
AS
BEGIN
DELETE FROM Autor WHERE AutorID IN (SELECT AutorID FROM LivroAutor
WHERE LivroID IN (SELECT LivroID FROM DELETED) GROUP BY AutorID HAVING
COUNT(*) = 0);
END

12. Crie um trigger que limite o número de livros em uma categoria específica na tabela Categoria para um máximo de 100 livros e não permita mais inserções além desse limite.

Solução: CREATE TRIGGER trg_limit_books_per_category ON Livro
BEFORE INSERT
AS
BEGIN
IF (SELECT COUNT(*) FROM Livro WHERE CategoriaID = INSERTED.CategoriaID)
>= 100
BEGIN
RAISEERROR('Limite de 100 livros por categoria atingido.', 16, 1);
ROLLBACK TRANSACTION;
END
END

13. Crie um trigger que atualize automaticamente o campo nacionalidade na tabela Autor para 'Desconhecida' sempre que um autor for excluído da tabela Autor.

Solução: CREATE TRIGGER trg_update_nationality ON Autor
AFTER DELETE
AS
BEGIN
UPDATE Autor SET Nacionalidade = 'Desconhecida' WHERE AutorID IN (SELECT
AutorID FROM DELETED);
END

14. Crie um trigger que registre automaticamente todas as exclusões de livros na tabela LogExclusaoLivros com detalhes sobre o livro excluído, data e hora.

```
Solução: CREATE TRIGGER trg_log_deletion ON Livro
AFTER DELETE
AS
BEGIN
    INSERT INTO LogExclusaoLivros (LivroID, Titulo, DataHora) VALUES (SELECT
LivroID, Titulo, GETDATE() FROM DELETED);
END
```

15. Crie um trigger que, quando um livro for atualizado na tabela Livro, verifique se o novo título contém a palavra 'proibido' e, se sim, reverta o título anterior.

```
Solução: CREATE TRIGGER trg_revert_prohibited_title ON Livro
AFTER UPDATE
AS
BEGIN
    IF (SELECT Titulo FROM INSERTED) LIKE '%proibido%'
    BEGIN
        UPDATE Livro SET Titulo = (SELECT Titulo FROM DELETED) WHERE
LivroID IN (SELECT LivroID FROM INSERTED);
    END
END
```

16. Crie um trigger que, quando um livro for inserido na tabela Livro, verifique se a editora está na lista de editoras proibidas e, se estiver, impeça a inserção.

```
Solução: CREATE TRIGGER trg_restrict_publisher ON Livro
BEFORE INSERT
AS
BEGIN
    IF (SELECT EditoraID FROM INSERTED) IN (SELECT EditoraID FROM
EditorasProibidas)
    BEGIN
        RAISEERROR('Editora proibida.', 16, 1);
        ROLLBACK TRANSACTION;
    END
END
```

17. Crie um trigger que atualize automaticamente o campo total_livros_escritos na tabela Autor sempre que um livro for associado a esse autor na tabela LivroAutor.

Solução: CREATE TRIGGER trg_update_total_books_written ON LivroAutor
AFTER INSERT
AS
BEGIN
 UPDATE Autor SET total_livros_escritos = (SELECT COUNT(*) FROM LivroAutor
WHERE AutorID = INSERTED.AutorID)
 WHERE AutorID = (SELECT AutorID FROM INSERTED);
END

18. Crie um trigger que registre automaticamente todas as atualizações de livros na tabela LogAtualizacaoLivros com detalhes sobre o livro atualizado, data e hora.

Solução: CREATE TRIGGER trg_log_update ON Livro
AFTER UPDATE
AS
BEGIN
 INSERT INTO LogAtualizacaoLivros (LivroID, Titulo, DataHora) VALUES (SELECT
LivroID, Titulo, GETDATE() FROM INSERTED);
END