



EAD - Polo Santa Luiza – Vitória – ES

DESENVOLVIMENTO FULL STACK

RELATÓRIO DE PRÁTICA

MUNDO 3 – NÍVEL 2 – MISSÃO PRÁTICA

RPG0015 - Vamos manter as informações

CARLOS ALTOMARE CATÃO

Semestre Letivo: 2025/1 - 3º Período

Data: 2025/04

PROCEDIMENTO 2

Objetivos da Prática:

1. Identificar os requisitos de um sistema e transformá-los no modelo adequado.
2. Utilizar ferramentas de modelagem para bases de dados relacionais.
3. Explorar a sintaxe SQL na criação das estruturas do banco (DDL).
4. Explorar a sintaxe SQL na consulta e manipulação de dados (DML)
5. No final do exercício, o aluno terá vivenciado a experiência de modelar a base de dados para um sistema simples, além de implementá-la, através da sintaxe SQL, na plataforma do SQL Server.

Objetivos do Procedimento 2: Alimentando a Base

O objetivo do segundo procedimento é desenvolver habilidades na manipulação de Tabelas de um Banco de Dados Relacional, alimentando as mesmas com dados básicos do Sistema, através da utilização do *Software SSMS* (Microsoft SQL Server Management Studio).

Códigos:

Os códigos relativos a esta prática se encontram em reservatório no GitHub e podem ser acessados pelo link: https://github.com/CarlosCatao/Mundo_3_Nivel_2_Missao_Pratica/tree/main/Procedimento_2/CODIGOS%20SQL.

Resultados:

Os resultados da execução dos códigos se encontram no arquivo PDF RESULTADOS.pdf armazenado em repositório GitHub e pode ser acessado pelo link: https://github.com/CarlosCatao/Mundo_3_Nivel_2_Missao_Pratica/blob/main/Procedimento_2/RESULTADOS.pdf.

Análise e Conclusão:

Esta atividade prática explorou muito a utilização da linguagem *sql* além de expor o resultado da Modelagem a testes exaustivos com relação à sua acurácia. Certamente foi muito proveitosa.

Quais as diferenças no uso de SEQUENCE e IDENTITY?

- **SEQUENCE:** A SEQUENCE é um objeto independente que pode ser usado para gerar números sequenciais. Ela não depende de uma tabela específica e pode ser usada em diferentes tabelas. Além disso, oferece mais flexibilidade, permitindo controle sobre os valores de incremento, de início e outras opções.
- **IDENTITY:** O IDENTITY é uma propriedade de coluna que gera automaticamente números sequenciais para uma tabela específica. Ele é mais simples, mas menos flexível que a SEQUENCE, pois não pode ser reutilizado em outras tabelas e não permite controle total sobre os valores gerados (como o valor inicial ou o incremento).

Qual a importância das chaves estrangeiras para a consistência do banco?

As *chaves estrangeiras* são fundamentais para garantir a *integridade referencial* de um banco de dados. Elas garantem que os dados em tabelas relacionadas sejam consistentes, impedindo:

- A inserção de dados inválidos ou órfãos.
- A exclusão de registros que estão sendo referenciados em outras tabelas, evitando dados órfãos.

Isso garante que os dados armazenados sejam sempre consistentes e que as relações entre as entidades sejam preservadas.

Quais operadores do SQL pertencem à álgebra relacional e quais são definidos no cálculo relacional?

- **Álgebra Relacional:** A álgebra relacional fornece operadores como:
 - **SELECT (σ)**: Filtra linhas de uma tabela com base em uma condição.
 - **PROJECT (π)**: Seleciona colunas específicas de uma tabela.
 - **JOIN (\bowtie)**: Combina registros de duas ou mais tabelas com base em uma condição.
 - **UNION (\cup)**: Combina os resultados de duas consultas, removendo duplicatas.
 - **INTERSECTION (\cap)**: Retorna os registros que aparecem em ambas as consultas.
 - **DIFFERENCE ($-$)**: Retorna os registros da primeira consulta que não aparecem na segunda.
- **Cálculo Relacional:** O cálculo relacional é um estilo declarativo, onde expressamos as condições que as tuplas devem satisfazer, sem especificar um procedimento para encontrar a resposta. Ele é dividido em:
 - **Cálculo de Tuplas:** Usa variáveis para representar tuplas, aplicando filtros com condições.
 - **Cálculo de Domínio:** Usa variáveis para representar valores, filtrando de acordo com as condições.

Como é feito o agrupamento em consultas, e qual requisito é obrigatório?

O agrupamento é feito usando a cláusula GROUP BY, que permite agrupar os resultados de uma consulta por uma ou mais colunas. Ele é frequentemente combinado com funções agregadas como COUNT(), SUM(), AVG(), etc.

Requisito obrigatório: Quando se utiliza GROUP BY, todas as colunas que não são usadas em funções agregadas (como SUM, AVG, COUNT) devem ser incluídas na cláusula GROUP BY. Caso contrário, o SQL Server gerará um erro.