



LABORATÓRIO DE BANCO DE DADOS

IBD100

IBD100 ÁLGEBRA RELACIONAL + SQL

João Maurício Hypólito
prof.oao.hypolito@gmail.com

Programa

- Histórico da AR
 - Origem
 - Conceitos fundamentais
 - Variações da proposta original
 - Símbolos adaptados
- Histórico da SQL
 - O projeto inicial
 - A evolução → SQUARE
 - Voltando ao SEQUEL
 - Atraindo a atenção e evoluindo para SQL.
- As operações de AR → SQL
 - Onde praticar? Base PostgreSQL
 - Os padrões SQL
 - Os módulos de linguagens dos SGBD e os Padrões SQL

- Histórico da AR – Origem

- 1970 – Edgard F. Codd apresenta o Modelo Relacional e muda a forma de desenvolver a modelagem de dados. Conceito: Teoria dos conjuntos. TABELA = {tuplas}
- Para processar as tabelas ele apresenta a Álgebra Relacional
- As tuplas são ‘elementos’ dos ‘conjuntos’ (tabelas), então as operações de elementos de conjuntos expressam um raciocínio de processamento de tuplas das tabelas.

- Histórico da AR – Conceitos fundamentais:

- Tabela={ tuplas }
- Tabelas são definidas por seus campos.
 - Entre tabelas com igual ESQUEMA é possível fazer: $\cup$, $-$ e $\cap$
 - ESQUEMA igual (campos com mesmo nome, mesma ordem, mesmo tipo de dado e mesmo tamanho na tabela) $\rightarrow$ tabelas União Compatível
- Operações: Seleção, Projeção, Produto Cartesiano e Junção.
- Permitem a combinação de dados das tabelas.

IBD100-ÁLGEBRA RELACIONAL + SQL

- Histórico da AR – Variações da Proposta Original

- A Álgebra Relacional foi apresentada com uma notações de símbolos gregos: π σ X , $\bowtie$, etc.
- Esta notação de símbolos não é adequada para edição de operações usando editores de texto simples. Por isso, usuários fizeram a proposta de símbolos em texto simples para as operações da AR.
- U passa a ser U,
- $-$ passa a ser – e
- $\cap$ passa a ser ^
- π – projeção – passa a ser escrita com o nome da tabela seguido de [] delimitando os nomes de campos.
- σ – seleção – passa a ser escrita com o nome da tabela seguido da condição escrita dentro de []
- X – produto cartesiano – passa a ser escrita com os nomes das tabelas escritos de cada lado do 'X'.
- $\bowtie$ – Junção - passa a ser escrita com os nomes das tabelas de cada lado de [] e com a condição dentro dos [].

- Assim as operações podem ser rerepresentadas assim:

Operação	Símbolo adaptados
Seleção	Tab[condição]
Projeção	Tab[lista de campos]
Produto Cartesiano	Tab1 X Tab2
Junção	Tab1[condição]Tab2
União	Tab1 U Tab2
Subtração	Tab1 – Tab2
Intersecção	Tab1 ^ Tab2

IBD100-ÁLGEBRA RELACIONAL + SQL

- Depois que desenvolveram o SQL veio a necessidade apresentarem os símbolos para junção aberta. Na simbologia fundamental bastaria retirar as barras desenhadas nos lados do símbolo $\bowtie$.
- Mas esta escrita ficaria complicada na simbologia adaptada. Então isso foi resolvido usando simplesmente o símbolo $[[]]$ no lugar de $[]$
- Histórico da AR
Símbolos adaptados para as operações.

Operação	Símbolo adaptados
Seleção	Tab $[[condição]]$
Projeção	Tab $[[\text{lista de campos}]]$
Produto Cartesiano	Tab1 X Tab2
Junção	Tab1 $[[condição]]$ Tab2
União	Tab1 U Tab2
Subtração	Tab1 – Tab2
Intersecção	Tab1 ^ Tab2
Junção aberta pela direita	Tab1 $[[condição]]$ Tab2
Junção aberta pela esquerda	Tab1 $][condição]]$ Tab2
Junção completa (aberta pelos dois lados)	Tab1 $][condição]]$ Tab2

IBD100-ÁLGEBRA RELACIONAL + SQL

- Histórico do SQL.
 - 1970: E.F. Codd – "A Relational Model of Data Large Shared Banks" → Modelo Relacional $\Rightarrow$ Álgebra Relacional $\therefore$ indica uma Linguagem de Pesquisa;
 - Apresenta a Álgebra Relacional
um conjunto de símbolos que permite uniformizar o entendimento sobre as operações que podemos realizar sobre as tuplas das tabelas
 - Consequência da Álgebra Relacional:
 - Criação da Linguagem de Manipulação para pesquisa dos dados nos Banco de Dados
 - 1974: Donald D. Chamberlin e Raymond Boyce (pesquisadores do "IBM San Jose Research Center"): artigo sugerindo a forma da linguagem de consulta estruturada. A linguagem foi chamada de SEQUEL;
 - 1975: Chamberlin, Boyce, King e Hammer apresentam a SQUARE (com símbolos matemáticos), semelhante a SEQUEL (mas esta tinha termos em inglês);
 - 1976: Chamberlin apresenta a SEQUEL2 que passa a ser usada como linguagem de consulta para o banco de dados que se pesquisava na IBM, o sistema R: criam-se grupos de usuários, a linguagem começa a evoluir rapidamente ;

IBD100-ÁLGEBRA RELACIONAL + SQL

- Histórico do SQL.
 - 1977: O SystemR-IBM (com SEQUEL/2) torna-se operacional + Oracle (Relacional)
 - 1980: Chamberlin resume as experiências dos usuários com a linguagem (seu nome a esta altura já era SQL – Structured Query Language).
 - 1983: IBM lança o DB2. Outros produtos no mercado: Sybase, Ingres, Progress), SQL é padrão de fato;
 - 1986: SQL é homologada como linguagem padrão ANSI para tratamento de dados em BD Relacionais basicamente a linguagem desenvolvida para o SystemR-IBM
- Os padrões da linguagem
 - 1987: Padrão ANSI é aceito pelo ISO (ANSI-SQL:86)
 - 1989: SQL (ANSI-SQL:89) – incorpora comandos para junções abertas e completas
 - 1992: Os comitês ISO e ANSI apresentam SQL2 (SQL:1992) incorpora características de tratamento de integridade (de CP e CE), tabelas temporárias, novas funções, expressões nomeadas, valores únicos, instrução CASE, etc.
 - 1999: SQL:1999 – implementação de expressões regulares, recursos de orientação a objetos, queries recursivas, triggers, novos tipos de dados (boolean, LOB, array e outros), novos predicados etc.
 - 2003: SQL:2003: - inclui suporte básico ao padrão XML, sequências padronizadas, instrução MERGE, colunas com valores auto-incrementais etc.
 - 2006: SQL:2006: - Não inclui mudanças significativas para as funções e comandos SQL. Reforça a interação SQL $\leftrightarrow$ XML

IBD100-ÁLGEBRA RELACIONAL + SQL

• A Evolução dos SGBDs

- Os SGBDs são o resultado do amadurecimento das necessidades empresariais para suporte no desenvolvimento de Sistemas de Informação
- Sistemas de Informação são um conjunto de computadores e outros dispositivos que tem a finalidade de fornecer meios de entrada, armazenamento, processamento e apresentação de saídas de dados que permitam estabelecer parâmetros que melhorem a **tomada de decisão** pelas organizações (empresas, escolar, etc).
- OS SGBD devem dar suporte ao projeto do SI, portanto devem permitir:
 - Criar os arquivos que armazenam os dados e administrar estes arquivos
 - Implementar os programas aplicativos (que contém as regras de negócio das organizações).
- No grupo de linguagens os SGBDs devem ter os comandos em três categorias:

Categoria	Descrição
DDL - <i>Data Defination Language</i>	Permite que sejam caracterizados a estrutura dos arquivos onde se gravam os dados da organização. É possível definir o tamanho dos campos, do arquivo como um todo e seus índices associados
DCL - <i>Data Control Language</i>	Linguagem de Definição de Usuários e Grupos de usuários e suas atribuições de acesso aos Dado
DML - <i>Data Manipulation Language</i>	Linguagem de Manipulação de Dados. É o grupo de comandos que permite incluir uma nova linha em uma tabela, pesquisar uma linha, alterar alguns dos valores de seus campos ou mesmo remover a linha da tabela

IBD100-ÁLGEBRA RELACIONAL + SQL

- Os padrões da linguagem e os comandos (em categorias) dos SGBDs
 - De 1975 até meados da década de 1990 o ANSI estudava o SQL e indicava o caminho que os SGBD iriam trilhar.
 - O ritmo de desenvolvimento dos padrões SQL andou “atrás” do desenvolvimento dos SGBD a partir da década de 1990.
 - Isso fez com que os padrões posteriores à 1992 (SQL:1992) passassem a ter baixa adesão pelos SGBDs.
 - Mesmo os SGBDs lançados após 1992 passaram a adotar o SQL:1992 como seu SQL referencial, embora pudessem ter mais funções.
 - Desta forma os SGBD passaram a ter três blocos de linguagens:
 - A linguagem Nativa (incorporando funções específicas do SGBD)
 - A linguagem padrão ANSI (na maioria das vezes: SQL:1992)
 - A linguagem SQL-Nativa – que têm comandos no ‘formato’ SQL mas com a incorporação de funções do SGBD.

IBD100-ÁLGEBRA RELACIONAL + SQL

- Então os comandos SQL 'distribuídos' nas categorias de comandos dos SGBDs são:

DDL:

CREATE
ALTER
DROP

DCL:

GRANT
REVOKE

DML:

SELECT
INSERT
DELETE
UPDATE

Todas as Operações
da AR começam com
este comando SQL

- Para prática é necessário instalar o SGBD PostgreSQL:
 - BAIXAR O SCRIPT que cria as tabelas para teste de SQL (site e Teams).
 - Acesse o aplicativo pgAdmin 4
 - CRIAR uma base de dados (sugestão de nome: **baseibd100**). Nesta base de dados...
 - EXECUTAR o script para criar as tabelas
- Vamos fazer esta prática depois de exercitar algumas recuperações de dados em exercícios teóricos com 'SQL' escritos mesmo À MÃO (mesmo).

As operações de AR são:

SELEÇÃO

PROJEÇÃO

PRODUTO CARTESIANO

JUNÇÃO: NATURAL (FECHADA)

ABERTA DIREITA

ABERTA ESQUERDA

ABERTA DIR $\leftrightarrow$ ESQ (COMPLETA)

UNIÃO

SUBTRAÇÃO

INTERSECÇÃO



LABORATÓRIO DE BANCO DE DADOS

IBD100