

## 1. Introdução

A primeira versão da linguagem SQL, chamada SEQUEL (Structured Query English Language), surgiu em 1974 nos laboratórios da IBM (Califórnia). Entre 1976 e 1977 ela foi revisada e ampliada, tendo então o seu nome alterado para SQL.

Devido ao sucesso da nova forma de consulta e manipulação de dados dentro de um ambiente de banco de dados, sua utilização tornou-se cada vez maior. Vários SGBD's atuais utilizam o SQL como a linguagem padrão para o acesso às bases de dados. Entre eles podemos citar:

- ✓ DB2 da IBM
- ✓ ORACLE da Oracle Corporation;
- ✓ RDB da Digital
- ✓ SYBASE da Sybase INC
- ✓ SQL Server da Microsoft
- ✓ Ingres da Computer Associates

Em 1982 o American National Standard Institute (ANSI) tornou a SQL a linguagem padrão para a manipulação de dados em ambiente relacional.

## **2. A Linguagem SQL**

A linguagem SQL pode ter vários enfoques:

### **Linguagem interativa de consulta (query AdHoc)**

Através de comandos SQL os usuários podem montar consultas poderosas, sem a necessidade da criação de um programa, podendo utilizar ferramentas front-end para a montagem de relatórios.

### **Linguagem de programação para acesso às bases de dados**

Comandos SQL embutidos em programas de aplicação (escritos em C, C++, Java, Visual Basic e etc) acessam os dados armazenados em uma base de dados relacional.

### **Linguagem de administração de banco de dados**

O responsável pela administração do banco de dados (DBA) pode utilizar comandos SQL para realizar tarefas relacionadas com a manutenção dos schemas do banco de dados.

### **Linguagem de consulta em ambiente cliente/servidor**

Os programas sendo processados nos computadores dos clientes (front ends) usam comandos SQL para se comunicarem, através de uma rede, com um SGBD sendo processado em uma máquina servidora (back end);

### **Linguagem para bancos de dados distribuídos**

A linguagem SQL é também a linguagem padrão para a manipulação de dados em uma base de dados distribuída.

### **Linguagem de definição de dados (DDL)**

Permite ao usuário a definição da estrutura e organização dos dados armazenados, e das relações existentes entre eles.

### **Linguagem de manipulação de dados (DML)**

Permite a um usuário, ou a um programa de aplicação, a execução de operações de inclusão, remoção, seleção ou atualização de dados previamente armazenados na base de dados.

### **Controle de acesso**

Protege os dados de manipulações não autorizadas.

### **Integridade dos dados**

Auxilia no processo de definição da integridade dos dados, protegendo contra corrupções e inconsistências geradas por falhas do sistema de computação, ou por erros nos programas de aplicação.

### **3. Vantagens e Desvantagens da Linguagem SQL**

Podemos apontar as seguintes vantagens no uso da linguagem SQL:

#### **Independência de fabricante**

A linguagem SQL é adotada por praticamente todos os SGBD's relacionais existentes no mercado, além de ser uma linguagem padronizada (ANSI). Com isso, pelo menos em tese, posso mudar de SGBD sem me preocupar em alterar os programas de aplicação.

#### **Portabilidade entre plataformas de hardware e software**

Pode ser utilizada tanto em máquinas Intel rodando Windows, passando por workstations RISC rodando UNIX, até mainframes rodando sistemas operacionais proprietários.

#### **Redução dos custos com treinamento**

Com base no item anterior, as aplicações podem se movimentar de um ambiente para o outro sem que seja necessária uma reciclagem da equipe de desenvolvimento.

#### **Usa inglês estruturado de alto nível**

O SQL é formado por um conjunto bem simples de sentenças em inglês, oferecendo um rápido e fácil entendimento.

#### **Permite consultas interativas**

Permite aos usuários acesso fácil e rápido aos dados a partir de um front end que permita a edição e a submissão de comandos SQL.

## **Múltiplas visões dos dados**

Permite ao criador do banco de dados levar diferentes visões dos dados aos diferentes usuários.

## **Definição dinâmica dos dados**

Através da linguagem SQL pode-se alterar, expandir ou incluir, dinamicamente, as estruturas dos dados armazenados, com máxima flexibilidade.

Porém, existem também algumas desvantagens no uso da linguagem SQL:

Críticas (segundo C.J. Date)

- ✓ Falta de ortogonalidade nas expressões, funções embutidas, variáveis indicadoras, referência a dados correntes, constante NULL, conjuntos vazios, e etc;
- ✓ Definição formal da linguagem após sua criação;
- ✓ Discordância com as linguagens hospedeiras (geralmente procedurais e orientadas para registros e não para conjuntos);
- ✓ Falta de algumas funções;
- ✓ Não dá suporte a alguns aspectos do modelo relacional (join explícito, domínios, e etc.)

#### **4. Tabelas dos Exemplos**

As tabelas a seguir serão usadas nos exemplos que se seguem:

CLIENTE (cod\_cli, nome\_cli, endereco, cidade, cep, uf)

VENDEDOR (cod\_vend, nome\_vend, sal\_fixo, faixa\_comiss)

PEDIDO ( num\_ped, prazo\_entr, cd\_cli, cd\_vend)

ITEM\_PEDIDO (no\_ped, cd\_prod, qtd\_ped)

PRODUTO (cod\_prod, unid\_prod, desc\_prod, val\_unit)

## 5. Criação e Destruição de Tabelas

O comando CREATE TABLE é usado para criar uma tabela. A sua forma geral é:

```
CREATE TABLE <nome_tabela>
    (<descrição das colunas>,
     <descrição das chaves>);
```

onde:

**<nome\_tabela>** dever ser substituído pelo nome da tabela a ser criada.

**<descrição das colunas>** deve ser substituída pela relação das colunas da tabela e seus respectivos tipos de dados (por exemplo, smallint, char, varchar, integer, number, float e etc).

**<descrição das chaves>** deve ser substituída pela lista das colunas que são tratadas como chaves estrangeiras.

Algumas colunas podem receber o valor NULL (nulo), e a coluna definida como chave primária, além de não poder receber NULL, deve ser uma coluna UNIQUE (sem repetições; isto é, chave primária)

### Script de Criação das Tabelas dos Exemplos

```
create table cliente
(
    cod_cli          smallint    not null,
    nome_cli         varchar(40) not null,
    endereco         varchar(40) null,
    cidade           varchar(20) null,
    cep              char(08)     null,
    uf               char(02)     null,
    primary key (cod_cli));

create table vendedor
(
    cod_vend         smallint    not null,
    nome_vend        varchar(40) not null,
    sal_fixo         number(9,2) not null,
```

```

        faixa_comiss char(01)    not null,
        primary key (cod_vend));

create table produto
(
    cod_prod      smallint    not null,
    unid_prod     char(03)    not null,
    desc_prod     varchar(20) not null,
    val_unit      number(9,2) not null,
    primary key (cod_prod));

create table pedido
(
    num_ped      smallint    not null,
    prazo_entr   smallint    not null,
    cd_cli       smallint    not null
REFERENCES CLIENTE (cod_cli),
    cd_vend      smallint    not null
REFERENCES VENDEDOR (cod_vend),
    primary key (num_ped));

create table item_pedido
(
    no_ped      smallint    not null
REFERENCES PEDIDO (num_ped),
    cd_prod     smallint    not null
REFERENCES PRODUTO (cod_prod),
    qtd_ped     float       not null);

```

Para excluirmos uma tabela existente devemos usar o comando DROP TABLE. A sua forma geral é:

DROP TABLE <nome\_tabela>;

onde:

<nome\_tabela> dever ser substituído pelo nome da tabela a ser excluída.

## Exemplos

```

drop table item_pedido;
drop table pedido;
drop table vendedor;
drop table produto;
drop table cliente;

```



## 6. Executando Consultas sobre as Tabelas

### 6.1 Selecionando Colunas Específicas de uma Tabela

```
SELECT <lista_de_colunas>  
      FROM <nome_tabela>;
```

Problema:

Listar todos os produtos com as respectivas descrições, unidades e valores unitários.

```
select desc_prod,unid_prod,val_unit  
      from produto;
```

DESC_PROD	UNI	VAL_UNIT
Chapa de Aco	kg	2,5
Cimento	kg	4,5
parafuso 3.0X10.5 mm	kg	2
Fio plastico	m	,2
Solvente PRW	l	5

5 linhas selecionadas.

Problema:

Listar os nomes dos clientes, as cidade e os estados onde eles estão localizados.

```
select nome_cli,cidade,uf  
      from cliente;
```

NOME_CLI	CIDADE	UF
Supermercado Carrefour	rio de janeiro	rj
Supermercado Baratao	rio de janeiro	rj
Supermercado Arariboia	niteroi	rj
UFF	niteroi	rj
CSN	volta redonda	rj
Pegeout	resende	rj
Ind. Quimicas Paulistas	sao paulo	sp
Ford Caminhoes	sao paulo	sp
Riocel Celulose	guaiba	rs
Elevadores Sur	guaiba	rs