SQL

* SQL – Structured Query Language
  + Linguagem baseada no inglês
  + Pode processar um conjunto de linhas por vez
  + Serve para a comunicação com o banco de dados relacional
* T-SQL
  + SQL (Structured Query Language) – Linguagem estruturada de pesquisa\consulta\manipulação
  + Fundamentada no modelo relacional
* Vantagens
  + Independência de fabricante
  + Portabilidade entre sistemas
  + Comandos em inglês
  + Consulta interativa
  + Manipulação dinâmica
* Desvantagens
  + Não atende todos os modelos relacionais
* Categorias\Subdivisões
  + DMI – Data Manipulation Language
    - INSERT Table 1 Values (1,’Row #1’)
  + DCL – Data Control Language
    - GRANT CONNECT TO guest;
    - GRANT CONTROL SERVER TO user\_name;
  + DDL – Data Definition Language
    - CREATE USER user\_name;
  + TCL – Transactional Control Language
    - COMMIT TRANSACTION Inner2;
  + DDL – Data Query Language
    - SELECT ProductID, Name, ListPrice FROM Production.Product
* Objetos do SQL SERVER
  + Database
    - Banco de Dados – Todos os objetos serão criados dentro dele.
  + Table
  + DATA TYPE e USER DEFINED DATA TYPE
    - Tipos de dados que serão usados para o recebimento das informações
  + VIEW
    - Cria e Grava Consultas
  + INDEX
    - Objetos responsáveis pela otimização de acesso aos dados de uma tabela
  + Procedure
    - É onde criamos os comandos de programação para executar
  + Trigger
    - Bloco de comandos que é criado sobre uma tabela e ativados no momento da execução dos comandos update, insert e\ou delete.
  + Function
    - Cria-se funções
* Bancos do Sistema
  + MASTER
    - É responsável por efetuar o registro de todas as informações do sistema
  + TEMPDB
    - É responsável por manter todo o tipo de armazenamento temporário, inclusive as tabelas temporárias
  + MODEL
    - É utilizado como modelo para criar todos os outros bancos de dados
  + MSDB
    - É utilizado pelo SQL Agent com a funcionalidade de registrar operadores e programar execução de Jobs e alerts.
* Tipos de dados numéricos
  + Tinyint – de 0 a 225(1 byte)
  + Smallint – (2 bytes)
  + Int – (4 bytes)
  + Bigint – (8 bytes)
  + Decimal (p,s) – 10^38 + 1 até 10^38 – 1
    - P = precisão; S=Escala
  + Numeric, Float, Real
  + Bit – valor de nº inteiro com o valor 1 ou 0
  + Binary(n)
  + Varbinary(n)
  + Varbinary(max) – **2^31-1**
* Tipos de dados Data e hora
  + Date – data entre 01\01\01 e 31\12\9999 – 3bytes
  + Datetime – Data e hora entre 01\01\1753 e 31\12\9999, com exatidão de 3.33 milisegundos. (8bytes)
  + Smalldatetime – Data e hora entre 01\01\1900 e 060\06\06, com exaltidão de 1 segundo.
* Tipos de dados String
  + Char(n) – Comprimento fixo de no máximo 8000 caracteres. Tem uma melhor performance, é mais rápido.
    - Ex: nome\_cliente char(100) – mas foi utilizado 50. Gastará o tamanho 100 e nada será devolvido.
  + Varchar – Comprimento fixo de no máximo 8000 caracteres. É mais lente, mas economiza nos dígitos
    - Ex: nome\_cliente char(100) – mas foi utilizado 50. Te devolverá os 50 characteres não utilizados.

Como criar tabelas:

<NOMECAMPO1><TipodeDados>

1. CREATE TABLE Tipo(Codigo INT NOT NULL PRIMARY KEY, Nome Varchar(30) NOT NULL)
2. Criação de uma Tabela com chave primária composta de 2 campos:

CREATE TABLE TotalFaturado(

Mês INTEGER NOT NULL,

Ano INTEGER NOT NULL,

Valor DECIMAL (12,2),

PRIMARY KEY(Mês, Ano)

)

1. Criação de uma tabela com chave estrangeira

Create TABLE Secao(

Codigo INT NOT NULL PRIMARY KEY,

Descricao VARCHAR(20) NOT NULL,

CodDepto INT FOREIGN KEY REFERENCES

Departamento (Codigo)

)

1. Criação de uma tabela com restrição de Preenchimento de Campos

CREATE TABLE Animal(

Codigo INT NOT NULL PRIMARY KEY,

Nome VARCHAR(40) NOT NULL,

Sexo char(1) check((upper(sexo) = ‘M’ or upper(sexo) = ‘F’))

)

Restrições:

-Not null: especifica que uma coluna não pode aceitar um valor nulo

-Default: atribui um valor padrão para uma coluna da tabela

-Unique: força a coluna a ter valores únicos, ou seja, não pode haver 2 registros com um mesmo valor

-Check: limita os valores que serão recebidos por uma coluna, através do teste de uma condição. Salário dec(7,2) check(salário <=20000.00)

-Primary key – chave primária

-Foregin key – chave estrangeria

-Identity – indica que um campo terá valor único incremental automaticamente gerado pelo sistema toda vez que um novo registro for incluído.

Ex: id\_num int IDENTITY(1,1) primary key

|  |
| --- |
| Pessoa |

|  |
| --- |
|  |

Herança

|  |
| --- |
| F  CPF  RG |

|  |
| --- |
| J  CNPJ  IE |

SETA AZUL: ESPECIALIZAÇÃO

SETA LARANJA: GENERALIZAÇÃO