#### Banco de Dados II

Prof. Helton de Andrade

#### Revisão Ultima Aula

Criando banco de Dados

Código:

Create database 2BDes;

Acessando Banco de Dados:

Use 2Bdes;

#### Criando Primeira Tabela

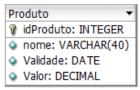
Código:
CREATE TABLE CLIENTE
(COD\_CLI INT,
NOME\_CLI VARCHAR(40));

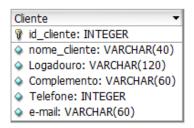
### Inserindo dados

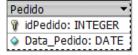
```
INSERT INTO CLIENTE VALUES (1,'ALEXANDRE DA SILVA'), (2,'ANA CLAUDIA DA SILVA'), (3,'BRUNO DA SILVA'), (4,'DEBORA SILVA'), (5,'TATIANA DA SILVA');
```

#### Criando Tabela com Chaves

## Criando Tabelas com Chaves







#### Criando tabela sem chave primaria

```
CREATE TABLE Cliente (
 id cliente INTEGER UNSIGNED NOT NULL
AUTO INCREMENT,
 nome_cliente VARCHAR(40) NULL,
 Logadouro VARCHAR(120) NULL,
 Complemento VARCHAR(60) NULL,
 Telefone INTEGER UNSIGNED NULL,
 email VARCHAR(60) NULL,
 PRIMARY KEY(id cliente)
);
```

#### Comando SQL

```
CREATE TABLE Pedido (
idPedido INTEGER NOT NULL AUTO_INCREMENT,
Data_Pedido DATE NULL,
PRIMARY KEY(idPedido)
);
CREATE TABLE Produto (
idProduto INTEGER NOT NULL AUTO_INCREMENT,
nome VARCHAR(40) NULL,
Validade DATE NULL,
Valor DECIMAL NULL,
PRIMARY KEY(idProduto)
);
```

#### Comando SQL

- CREATE TABLE Pedido (
- idPedido INTEGER UNSIGNED NOT NULL AUTO\_INCREMENT,
- Data\_Pedido DATE NULL,
- );

#### Cláusula CONSTRAINT

Um restrição é semelhante a um índice, embora também possa ser usado para estabelecer uma relação com outra tabela.

Use a cláusula CONSTRAINT nas instruções ALTER TABLE e CREATE TABLE para criar ou excluir restrições.

#### Criando Tabelas com Relacionamento

```
CREATE TABLE Pessoa
(IDPessoa integer primary key AUTO_INCREMENT,
 Nome varchar(255),
 Endereco varchar(255),
 Cidade varchar(255)
CREATE TABLE Carro
 ID_Carro integer PRIMARY KEY AUTO_INCREMENT,
 Nome varchar(255),
 Marca varchar(255),
 ID_Pessoa integer,
 CONSTRAINT fk_PesCarro FOREIGN KEY (ID_Pessoa) REFERENCES
  Pessoa (IDPessoa)
```

#### Inserindo dados nas Tabelas

IDPessoa	Nome	Endereço	Cidade
1	Maria	Rua Guarani, 002	Diadema
2	Joaquina	Rua Toro, 001	Diadema
3	João	Rua Lico Maia, 005	Diadema
4	José	Rua Bonifácio, 10	Santo André
5	Lucia	Rua Sete de Setembro, 07	São Bernardo
6	Oscar	Rua Prestes Maia, 55	São Caetano

#### Inserindo dados nas Tabelas

ID_carro	Nome	Marca	ld_pessoa
1	Fox	volkswagen	1
2	Onix	chevrolet	3
3	Ká	Ford	4
4	Corolla	Toyota	2
5	Up	Fiat	2

#### **Comandos DML**

#### Comandos DML

 Os comados DML (Data Manupilation Language) são utilizados para realizar consultas, inclusões, exclusões e alterações de dados presentes em registros.

#### Comandos

- Select
- Insert
- Update
- Delete

• São os comandos DML disponíveis para realizar as tarefas mencionadas.

#### Select

 Dentre as principais instruções oferecidas pela linguagem SQL, há aquelas que permitem realizar consultas aos dados presentes em uma tabela, a instrução select é a principal delas, uma vez que é responsável por retornar dados para aplicações ou, ainda, para outras instruções SQL.

# Sintase da instrução SELECT

- Select (colunas)
- From (tabelas)
- Whare(condição)
- Group by (grupo)
- Having (codições de grupo)
- Order by (classificar colunas)
- Limit (limites)

Exibindo todas as colunas

Select \* from pessoas

SELECT

Exibindo algumas colunas

• Select nome, endereço from pessoas

SELECT

# Exibindo algumas colunas usando Alias

 A exibição de colunas de uma determinada tabela é um procedimento que também pode ser realizado por meio de Alias. Neste caso, o alias é um apelido atribuído a uma coluna, o qual será visualizado no momento da exibição de seus valores. Esse alias permite não apenas renomear colunas, mas tmbém expressões em uma instrução SELECT. Vale destacar que o nome será exibido no resultado da novo consulta.

Exibindo algumas colunas usando Alias

 Select nome as 'Nome Pessoa', Endereço as 'Endereço da Pessoa' from Pessoa

#### Criando uma Coluna Virtual

 A instrução Select também pode ser utilizada para exibir os dados de uma tabela e, ainda, apresentar uma coluna que fisicamente não existe. Para compreendermos este aspecto de forma adequada, consideremos a tabela Clientes, na qual a coluna TIPO Cliente não existe.

#### Criando uma Coluna Virtual

 Select nome as 'Nome Pessoa', Endereço as 'Endereço da Pessoa', 'Pessoa Física' AS 'Tipo\_cliente' from Pessoa

Com a instrução SELECT, podemos exibir o resultado de um cálculo sem que o valor obtido seja necessariamente gravado na tabela. Para compreender este conceito de forma adequada, primeiramente criaremos uma tabela com o nome treinamento e com os campos cod treinamento, treinamento e valor.

```
Create table treinamento (cod_trei int,nome_Treinamento varchar (30),Valor decimal(10,2));
Insert into treinamento values
```

(1,'WORD','720.00'),

(2,'EXCEL','850.00'),

(3,'POWERPOINT','550.00');

Após inserir os dados na tabela, devemos considerar a seguinte situação:

Precisamos analisar os valores dos treinamentos com acréscimo de 10%, porém, desejamos que esses valores não sejam alterados e gravados na tabela.

Para tanto, devemos escrever o comando SELECT da seguinte forma;

SELECT TREINAMENTO, VALOR \*1.1 AS 'VALOR COM AUMENTO' FROM TREINAMENTO;

### Dúvidas >

