



---

# PERGUNTAS PARA APRESENTAÇÃO

---

Módulo de Desenvolvimento



Feito Por:

Daniel Willians Ignacio de Souza

Thomaz William Damaceno Abot

Lilia Pereira Gonçalves



29 DE MAIO DE 2015

NEW-SOFT SOLUÇÕES INOVADORS



### 1) O que é o diagrama de Atividade?

O objetivo do diagrama de atividades é mostrar o fluxo de atividades em um único processo. O diagrama mostra como uma atividade depende uma da outra. As atividades são conectadas através de arcos que mostram as dependências entre elas.

Exemplo: Descrição do exemplo: Retirando dinheiro de um caixa eletrônico (para cartões de crédito).

### 2) O que é o diagrama de Caso de uso?

O Diagrama de Casos de Uso tem objetivo auxiliar a comunicação entre os analistas e o cliente. Este diagrama descreve um cenário que mostra as funcionalidades do sistema do ponto de vista do usuário. Ele documenta o que o sistema faz do ponto de vista do usuário..

Exemplo:

Ator = Cliente Caso de uso = Atividade



### 3) O que é o diagrama de Estado?

Em um diagrama de estado, um objeto possui um comportamento e um estado. O estado de um objeto depende da atividade na qual ele está processando. Um diagrama de estado mostra os possíveis estados de um objeto e as transações responsáveis pelas suas mudanças de estado.

Exemplo:

Descrição do exemplo: Modelagem do sistema de login. Para que o usuário seja autenticado, ele deve fornecer dois valores: SSN (Social Security Number) e o PIN (Personal ID Number). Após a submissão é feita uma validação.

### 4) O que é o diagrama de Classe?

Seu objetivo é descrever os vários tipos de objetos no sistema e o relacionamento entre eles. Um diagrama de classes pode oferecer três perspectivas, cada uma para um tipo de observador diferente. São elas:

- **Conceitual**

Representa os conceitos do domínio em estudo.

Perspectiva destinada ao cliente.

Especificação

Tem foco nas principais interfaces da arquitetura, nos principais métodos, e não como eles irão ser implementados.

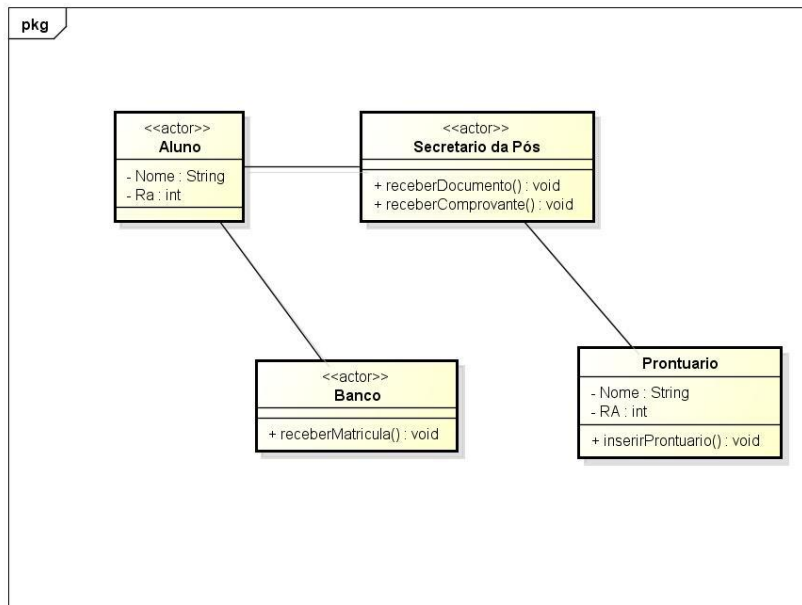
- **Perspectiva destinada as pessoas que não precisam saber detalhes de desenvolvimento,** tais como gerentes de projeto.

- **Implementação - a mais utilizada de todas**

Aborda vários detalhes de implementação, tais como navegabilidade, tipo dos atributos, etc.

Perspectiva destinada ao time de desenvolvimento

Exemplo:



### Outros Diagramas:

- **Diagramas de Pacotes:** Objetivo principal: agrupar classes em pacotes.
- **Diagrama de interação:** Objetivo captar o comportamento entre objetos dentro um único use case.

## 5) O que é UML?

UML é uma linguagem para visualização, especificação, construção e documentação de artefatos de um software em desenvolvimento.

## 6) O que é banco de dados?

Bancos de dados, ou bases de dados, são coleções de informações que se relacionam de forma que crie um sentido. São de vital importância para empresas, e há duas décadas se tornaram a principal peça dos sistemas de informação.

## 7) O que é Unicode?

O UNICODE define um padrão de codificação que pode ser utilizado com qualquer tipo de caracteres no mundo todo. Se esta não for a sua necessidade, ou seja, se for trabalhar somente com textos comuns, sem caracteres especiais (por exemplo, caracteres japoneses) ou algo do tipo, utilize os tipos de dados CHAR ou VARCHAR.

## 8) O que é INT, VARCHAR, CHAR, NVARCHAR, NCHAR e?

INT: Valores numéricos inteiros variando de -2.147.483.648 até 2.147.483.647, usado para representar um inteiro, e poderia fazer uma correlação com os números do conjunto dos naturais na matemática.

CHAR(N): Armazena N caracteres fixos (até 8.000) no formato não Unicode. Se a quantidade de caracteres armazenada no campo for menor que o tamanho total especificado em N, o resto do campo é preenchido com espaços em branco.

VARCHAR(N): Armazena N caracteres (até 8.000) no formato não Unicode. Se a quantidade de caracteres armazenada no campo for menor que o tamanho total especificado em N, o resto do campo não é preenchido.

NCHAR(N): Armazena N caracteres fixos (até 4.000) no formato Unicode. Se a quantidade de caracteres armazenada no campo for menor que o tamanho total especificado em N, o resto do campo é preenchido com espaços em branco.

NVARCHAR(N): Armazena N caracteres (até 4.000) no formato Unicode. Se a quantidade de caracteres armazenada no campo for menor que o tamanho total especificado em N, o resto do campo não é preenchido.

Um *número inteiro* é um número sem uma parte fracionária, um número que você pode usar para contar coisas (embora inteiros também pode ser negativo). Os matemáticos podem distinguir entre os números naturais e números cardinais, e linguistas pode distinguir entre números cardinais e ordinais, mas estas distinções não nos interessam aqui.

Um *número real* é, para os nossos propósitos, simplesmente um número com uma parte fracionária. Desde que os computadores não costumam implementar números reais exatamente, não é necessário ou mesmo significativo para distinguir entre os números racionais, irracionais, e transcendentais.

**9) Qual comando é usado para criar um banco de DB?**

O comando CREATE DATABASE cria um banco de dados.

**10) Qual comando é usado para usar um banco de DB?**

Use (Nome do Banco de Dados) / GO

**11) Qual comando é usado para trocar o esquema?**

Create schema (Nome do seu Schema) authorization dbo/go

**12) Qual comando é usado para criar uma tabela no DB?**

create table (Nome da sua tabela)

**13) Qual comando é usado para sair de uma tela do programa?**

This.Close();

**14) Qual comando é usado para sair do programa?**

Application.Exit();

**15) Qual a diferença do comando que sai do programa e o que sai de uma tela do programa?**

O comando que sai do programa é o Application.Exit(); Ele sai definitivamente do programa, enquanto o comando this.Close(); somente fecha a sua tela atual e volta para a anterior .

**16) Qual comando é usado para excluir o bando de dados?**

Para excluir um banco de dados, usa-se o comando DROP DATABASE, seguido do nome do banco de dados que deseja apagar.

Exemplo: DROP DATABASE Padaria/go

**17) Qual comando é usado para excluir uma tabela no bando de dados?**

Para excluir uma tabela do banco de dados, usa-se o comando DROP Table, seguido do nome do banco de dados que deseja apagar.

drop table padaria.funcionario

### **18) Como chegaram à conclusão que o programa iria funcionar assim?**

Chegamos na conclusão porque nós visitamos uma padaria e analisamos o seu funcionamento.

### **19) O programa funciona na internet ou no desktop?**

O programa funciona pelo desktop.

### **20) O que é necessário para que o seu programa rode em outros computadores?**

É necessário que tenha o SQL e para te-lo é necessário no mínimo 6 GB de espaço disponível no disco rígido, Virtualização, Software rede, PowerShell e .netFramework. Também é necessário o Visual Studio

### **21) Programa multiplataforma?**

Um programa que rode em vários sistemas Operacionais.

### **22) Como fazer para ligar o banco com Visual Studio?**

Localize o "Data Sources". Se ele não estiver visível, para localizá-lo entre em "VIEW"/"OTHER WINDOWS"/"DATA SOURCES", certo?

Com o "Data Sources" aberto, clique em "ADD NEW DATA SOURCE"

Verifique se abriu a tela com o nome: "DATA SOURCE CONFIGURATION WIZARD"

Se sim, clique em "DATABASE", e NEXT

Depois em "DATASET", NEXT

Depois em "NEW CONNECTION"

Nesta tela de "ADD CONNECTION", alguns itens precisam ser selecionados.

Em "Server Name:" escolha o sua máquina como servidor. Dica, sua máquina é aquela que está escrito SQLEXPRESS, ok?

Em "LOG ON TO THE SERVER", escolha o "USE SQL SERVER AUTHENTICATION". Coloque "USER NAME" e "PASSWORD" e deixe o "SAVE MY PASSWORD" salvo, ok?

Em seguida no item "CONNECT TO A DATABASE", "SELECT OR ENTER A DATABASENAME:", escolha o database desejado.

Em seguida, "TEST CONNECTION" e depois "OK"

Na tela que apareceu, marque a opção do "YES", e NEXT.

NEXT

E na tela do "CHOOSE YOUR DATABASE OBJECTS", selecione somente as "TABLES", e depois "FINISH"

### **23) Como funciona o backup do programa?**

O backup funciona no form de backup nele você escolhe o numero da maquina, coloca a senha e em seguida manda conectar, depois de conectado seleciona o banco de dados e escolhe o lugar onde quer salvar e pronto.

### **24) Pra que serve o comando “not null”?**

É usado para o campo da tabela para ser preenchido obrigatoriamente, no caso do nome ficaria na tabela: Cli\_nm varchar(50) not null.

### **25) Como se chama o gerenciador do sgbd?**

Database Management System.

### **26) Qual a utilidade do go no SQL?**

Separar o começo e o fim do comando.

### **27) Qual motivo de mudar o schema no SQL?**

Em vez de ficar dbo.cliente fica padaria.cliente é mais fácil de organizar.

### **28) Qual utilidade do int identity(1,1) no SQL?**

É utilizado para que conte os códigos dos item da tabela. 1234...

### **29) Como fazer a inserção de dados no comando?**

insert into (Nome da tabela) values ('Constant','Welles', 'Constantin.Welles@ANHINGA.COM')

### **30) Como realizar uma pesquisa pelo comando no SQL?**

select \*  
from (Nome da sua tabela)

### **31) Como alterar tabela sem apagar?**

Alter table ami.gos  
Add primary key (ami.gos  
Go

### **32) O que é algoritmo?**

Conjunto de instruções para programa.

### **33) Quais os dois tipos de linguagem?**

Linguagem de programação de alto nível uma linguagem com um nível de abstração relativamente elevado, longe do código de máquina e mais próximo à linguagem humana. Desse modo, as linguagens de alto nível não estão diretamente relacionadas à arquitetura do computador. O programador de uma linguagem de alto nível não precisa conhecer características do processador, como instruções e registradores. Essas características são abstraídas na linguagem de alto nível.

Alguns exemplos de linguagens de alto nível: C/C++; C#; Java.

Linguagem de programação de baixo nível trata-se de uma linguagem de programação que compreende as características da arquitetura do computador. Assim, utiliza somente instruções do processador, para isso é necessário conhecer os registradores da máquina. Nesse sentido, as linguagens de baixo nível estão diretamente relacionadas com a arquitetura do computador. Um exemplo é a linguagem Assembly (não Assembler) que trabalha diretamente com os registradores do processador, manipulando dados.

### **34) O que é o .NET Framework?**

O .NET Framework é um modelo de programação de código gerenciado da Microsoft para criar aplicativos em clientes, servidores e dispositivos móveis ou incorporados ao Windows.

### **35) O que posso fazer com o .NET?**

O .NET permite desenvolver soluções como: Aplicativos WEB; Aplicativos para Servidores; Aplicativos para SmartClient; Aplicativos de Console; Aplicativos de Banco de Dados; Serviços Windows; Web Services e muito mais.

### **36) O que é o CLR?**

O Common Language Runtime (CLR) é o componente encarregado de gerenciar aplicações desenvolvidas em .NET.

### **37) Serviços disponíveis no CLR?**

Gerenciamento de Memória; Tratamento de exceções; Compilação; Segurança; Outros..

### **38) Quais Linguagens utiliza .NET?**

Microsoft Visual Basic; Microsoft C#; Microsoft C++ (Gerenciado/Não-Gerenciado) ; Microsoft J; .NET; Outras..



### **39) O que são Variáveis?**

Variáveis são espaços reservados (alocados) na memória RAM do computador, que devem possuir um nome e um tipo de dado, sendo que seu conteúdo pode sofrer alteração durante a execução do programa.

Exemplos: o Nome de um cliente, as Notas de um aluno etc.

### **40) O que são Variáveis Constantes?**

Constantes são espaços reservados (alocados) na memória RAM do computador, que devem possuir um nome e um tipo de dado, sendo que seu conteúdo NÃO pode sofrer alteração durante a execução do programa.

O valor que for definido na criação (declaração) de uma constante a acompanha durante todo o ciclo de vida do processo.

### **41) Quais são as regras para formação de variáveis?**

O nome de um identificador: não pode ser iniciado por número. Não pode conter espaços ou qualquer caractere especial, exceto o underline “\_”. Não pode ser igual ao de qualquer palavra reservada da linguagem de programação. Procure utilizar nomes curtos, e que possuam algum significado associado ao identificador. Atenção: Não é regra, mas procure sempre declarar uma variável com sua primeira letra minúscula. Será abordado mais a frente o assunto

### **42) O que são Laços de Repetição?**

Estruturas de repetição são comandos que permitirão a execução repetida de vários outros comandos descritos no algoritmo.

O comando será executado determinado número de vezes em função de uma condição que determinará o número de repetições a serem executadas.

Para isso, geralmente utilizamos uma variável que chamamos de contador.

### **43) O que é uma instância?**

Objetos são gerados a partir de classes

Uma classe define as propriedades e o comportamento dos objetos gerados por ela

Todo objeto é uma instância de uma classe

### **44) Como fazer uma Classes em C#?**

Definindo uma classe e seus atributos (características): `public class Conta {`

`public string numero;`

`public string donoDaConta;`

`public double saldo;`

`public double limite;`

}

OBS:É utilizado o modificador de acesso public devido ao fato de que em C# o padrão de todas as classes é estar atribuído com o modificador protected.

Portanto, não seria possível acessar a classe de fora dela se não adicionasse o modificador public. Falaremos mais detalhadamente sobre modificadores de acesso mais a frente.

Instanciando uma classe

```
Conta minhaConta = new Conta();
```

Através da variável minhaConta podemos acessar os atributos da classe Conta.

```
EX:minhaConta.numero = "123456";
```

#### 45) O que são Eventos

Eventos são ações que os componentes podem gerar.

A tabela a seguir consta a descrição dos eventos mais comuns:

Evento	Descrição
--------	-----------

Click	Ocorre quando o usuário clica sobre um controle ou, em alguns casos, quando a tecla ENTER é pressionada.
-------	----------------------------------------------------------------------------------------------------------

DoubleClick	Ocorre quando o usuário aplica um clique duplo.
-------------	-------------------------------------------------

KeyDown	Ocorre quando o usuário pressiona uma tecla no momento em que o controle está em foco. Ocorre antes de KeyPressKeyUp.
---------	-----------------------------------------------------------------------------------------------------------------------

KeyPress	Ocorre quando o usuário pressiona uma tecla no momento em que o controle está em foco. Ocorre depois de KeyDown antes de KeyUp.
----------	---------------------------------------------------------------------------------------------------------------------------------

KeyUp	Ocorre quando o usuário solta uma tecla no momento em que o controle está em foco.
-------	------------------------------------------------------------------------------------

#### 46) Qual a diferença entre Label e LinkLabel?

O Label é responsável por exibir um texto que não pode ser editado pelo usuário. É muito comum utilizá-lo para exibir uma mensagem referente ao status do processamento de alguma ação.

O LinkLabel apresenta um texto que é também um hyperlink.

Propriedades:

Propriedade	Descrição
-------------	-----------

BorderStyle	Define o estilo da borda que aparece em torno do label. Por padrão, não existe borda neste componente.
-------------	--------------------------------------------------------------------------------------------------------

Dock	Define se o label fica fixo nos cantos ou no centro da tela.
------	--------------------------------------------------------------

Font	Define a fonte utilizada para mostrar o texto do objeto
------	---------------------------------------------------------

Image	Define uma imagem para ser mostrada no controle Label.
-------	--------------------------------------------------------

Name	Define o nome do objeto
------	-------------------------

#### 47) Qual a diferença entre TextBox e RichTextBox?

O TextBox é um controle que possibilita a inserção de textos.

O RichTextBox aceita formatação caractere por caractere, diferente do TextBox. Propriedades:

Propriedade   Descrição

CharacterCasing   Define se o texto deve ser mantido em letras maiúsculas (UPPER), em letras minúsculas (Lower) ou da forma como o usuário inserir (Normal)

Clear()   Esse método tem como finalidade limpar o texto

MaxLength   Define a extensão máxima do texto a ser inserido no TextBox. Esse valor é definido em caracteres. Se for configurado para zero, o limite será estabelecido de acordo com a memória disponível.

PasswordChar   Define se os caracteres reais de uma senha devem ser substituídos por algum outro caractere (como o asterisco) quando exibidos na linha de texto.

ReadOnly   Define se o texto será somente leitura, evitando que seja editado.

#### 48) O que pode se fazer com um Button?

Este controle é responsável por induzir a aplicação a executar uma tarefa predefinida. Pode executar um dos seguintes três tipos de ações: Fechar uma caixa de diálogo; Abrir uma caixa de diálogo ou aplicação; Executar uma ação usando a informação inserida na caixa de diálogo.

Para utilizar este controle basta adicionar ao formulário, dar um duplo clique e adicionar o código ao evento Click.

Propriedades:

Propriedade   Descrição

Anchor   Define a posição do controle em relação às bordas do formulário

Enabled   Essa propriedade define se o botão deve permanecer habilitado ou desabilitado.

Image   Define uma imagem para ser exibida no botão.

#### 49) O que pode se fazer com um ListBox?

Responsável por exibir uma lista de opções dentre as quais o usuário pode optar, o controle ListBox é exibido como uma lista de strings. É indicado para os casos em que há um grande número de opções possíveis, permitindo ao usuário selecionar uma ou mais delas.

Propriedades:

Propriedade   Descrição

Items   Trata-se de uma coleção em que estão todos os itens da lista.

Sorted   Essa propriedade, se configurada como true, faz que itens contidos na lista sejam disponibilizados em ordem alfabética.

### 50) O que pode se fazer com um ComboBox?

Esse controle utiliza um campo de texto e uma lista. Tendo disponível esses dois elementos o usuário pode digitar ou selecionar um item da lista. Por padrão, o ComboBox aparece como um campo de texto com uma lista drop-down oculta.

Propriedades:

Propriedade    Descrição

**DropDownStyle**      Essa propriedade define o estilo do controle. Podemos configurá-lo de modo que seja exibida ou uma lista dropdown simples, o que mantém a lista sempre visível; ou uma lista dropdown em que o texto não pode ser editado pelo usuário e as opções só poderão ser visualizadas após o clique sobre uma seta; ou uma lista dropdown padrão, a qual permite edição no texto e exibe as opções somente após o clique em uma seta.

**Items**    Trata-se de uma coleção em que estão todos os itens da lista.

### 51) O que pode se fazer com um DateTimePicker?

Esse controle permite a seleção de uma data a partir de um calendário que é aberto quando o usuário clica sobre uma seta.

Propriedades:

Propriedade    Descrição

**Format**      Esta propriedade define o formato da data a ser mostrada.

**Value**    Essa propriedade contém a data selecionada.

### 52) O que pode se fazer com um TabControl?

Esse controle permite criar guias, as quais têm como objetivo organizar um formulário em diferentes partes, normalmente seguindo uma estrutura lógica.

Possui **TabPage**s, que são as guias responsáveis por agrupar os diferentes controles. Após adicionar a quantidade de guias com que se deseja trabalhar, basta arrastar os controles desejados para cada uma delas.

Propriedades:

Propriedade    Descrição

**Alignment**    Define a localização das guias dentro de **TabControl**. Por padrão, está configurada para exibir as guias no topo.

**Appearance**    Define a aparência das guias, que pode ser normal ou em alto relevo.

**Hot-Track**      Essa propriedade, quando configurada para **true**, faz com que a aparência das guias mude quando o ponteiro do mouse for posicionado sobre elas.

### 53) O que pode se fazer com um Timer?

Com a utilização do controle **Timer**, podemos executar um grupo de expressões para um período de tempo específico ou em intervalos específicos. Além disso, ele pode ser utilizado

para realizar a contagem regressiva a partir de um tempo presente, repetir uma ação em intervalos fixos ou causar um atraso em um programa.

Propriedades:

Propriedade	Descrição
-------------	-----------

Enabled	Essa propriedade habilita o funcionamento do Timer.
---------	-----------------------------------------------------

Interval	Essa propriedade define o intervalo, em milésimos de segundos, em que ocorrerá um evento.
----------	-------------------------------------------------------------------------------------------

#### **54) O que posso fazer com o MenuStrip?**

Os menus podem ser acrescentados nos programas com a utilização de um controle chamado MenuStrip.

O controle MenuStrip permite adicionar novos menus e também modificar, reordenar e excluir menus já existentes.

É possível também adicionar menus com efeitos como teclas de atalho, chaves de acesso e marcas de verificação.

Por meio de configurações disponíveis na janela Properties, torna-se possível personalizar as configurações do menu que é adicionado ao programa.

#### **55) O que posso fazer com o ToolStrip?**

Enquanto o MenuStrip representa o contêiner da estrutura do menu de um formulário, os objetos ToolStripMenuItem representam os comandos contidos na estrutura do menu, ou ainda, um menu pai para outros itens.

#### **56) O que é entidade?(relacionamento)**

O Modelo Entidade Relacionamento (também chamado Modelo ER, ou simplesmente MER), como o nome sugere, é um modelo conceitual utilizado na Engenharia de Software para descrever os objetos (entidades) envolvidos em um domínio de negócios, com suas características (atributos) e como elas se relacionam entre si (relacionamentos).

Em geral, este modelo representa de forma abstrata a estrutura que possuirá o banco de dados da aplicação. Obviamente, o banco de dados poderá conter várias outras entidades, tais como chaves e tabelas intermediárias, que podem só fazer sentido no contexto de bases de dados relacionais.

Observação: nem sempre criaremos modelos para um sistema completo, pois isso poderia resultar em um modelo muito extenso e difícil de interpretar.

### 57) O que é cardinalidade?

Cardinalidade de um Relacionamento Indica o número de instâncias de entidades que podem estar

Associadas umas às outras através de um relacionamento

### 58) O que são atributos?

Os Atributos em Programação Orientada a Objetos são os elementos que definem a estrutura de uma classe. Os atributos também são conhecidos como variáveis de classe, e podem ser divididos em dois tipos básicos: atributos de instância e de classe.

### 59) O que é modelagem de dados?

Modelagem de dados é o processo anterior a construção do banco de dados em um software.

### 60) Como obter a identificação do código?

Obtemos a identificação do código a partir do comando `int identity(1,1)primary key`.

### 61) Quantos forms tem no seu projeto?

Temos mais ou menos 27 telas e 3 classes.

### 62) Qual papel da análise de sistemas no seu projeto?

Ela é o seu planejamento do seu projeto os diagramas (um estudo antes de começar o projeto).

### 63) O que é concatenação?

Concatenar é juntar e não somar! Por exemplo se você somar 1+1 sempre dá 2, ou seja isso é soma! Quando você concatena 1+1 dá 11 ou seja Juntar!

O símbolo usado é o mesmo para soma ou seja (+) mas quando é variável numérica a própria linguagem sabe que é para somar e quando é texto (string) ela sabe que é para concatenar!

Concatenar é juntar dois valores, segue um exemplo em Java:

```
String a = "Minha";  
String b = "Casa";  
a.concat(b);  
System.out.println(a);
```

A saída deste programa será : Minha Casa.

Isso é um exemplo simples você poderia fazer a mesma coisa apenas somando as variáveis: a + b

Daria o mesmo resultado. Obs. se você for programar em Java o concatenar não funciona com variável do tipo String no lugar utilize StringBuffer.

#### 64) Por que utilizar bancos de dados informatizados?

Compacto (elimina arquivos de papéis); Rapidez; Integrado (vários aplicativos utilizam o mesmo repositório de dados); Compartilhado (vários usuários podem acessar);

#### 65) Para que serve o DER?

##### DIAGRAMA ENTIDADE RELACIONAMENTO- DER

O DER é uma ferramenta para modelagem conceitual de banco de dados amplamente utilizada no projeto de banco de dados, sendo considerado praticamente padrão para modelagem, por ser de fácil compreensão e apresentar poucos conceitos.

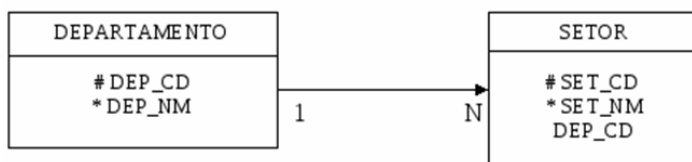
#### 66) GRAU DE CARDINALIDADE:

**1:1 (um para um):** é um tipo de relacionamento muito eficiente para evitar informações vazias, que ocupam espaço em disco e diminuem a performance do banco de dados.

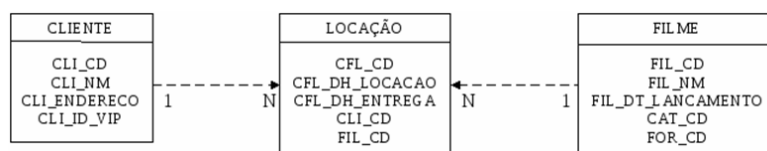


**1:N (um para muitos):** é o tipo de relacionamento mais comum e utilizado. Ele é usado quando há uma relação hierárquica entre as entidades envolvidas,

onde uma das entidades pode ser subordinada a uma de suas ocorrências



**M:N (muitos para muitos):** este tipo de relacionamento é utilizado quando no relacionamento entre duas entidades não há uma relação hierárquica ou complementar.



#### 67) O QUE SÃO RESTRIÇÕES DE INTEGRIDADE DE UM BD?

Integridade é um conceito fundamental em banco de dados, uma vez que diz respeito à correção, consistência e segurança dos dados armazenados. Os aspectos de integridade básica do modelo relacional estão associados aos conceitos de chave de acesso. Os principais tipos de chaves de acesso são:

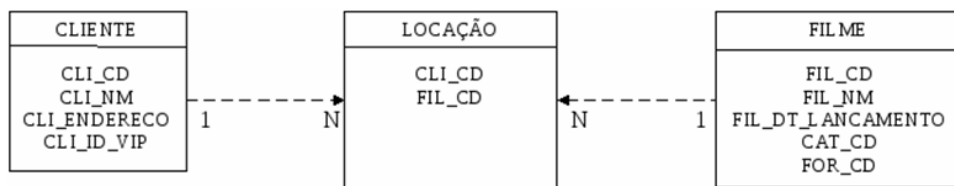
## 68) O que é Chaves Chave Primária Chave secundária, Chave estrangeira?

**Chave Primária (primary key):** é a principal chave de acesso a uma tabela. A criação dessa chave faz com que, automaticamente, a tabela seja ordenada por essa chave e que não seja permitida duplicidade em seu valor. A chave primária visa a manter a consistência e a unicidade dos dados de uma tabela. Ao escolher a chave primária, deve-se escolher um campo que Nunca vai se repetir na tabela.

Ex. de chave primária: CNPJ de um cliente poderia ser chave primária? Poderia, porque cada cliente tem um único CNPJ. Mas e no caso de clientes que não tem CNPJ? Então seria indicado criar um código com valor sequencial.

**Chave secundária:** é a chave auxiliar de acesso a uma tabela. A chave secundária também possui índices, que não podem ser repetidos.

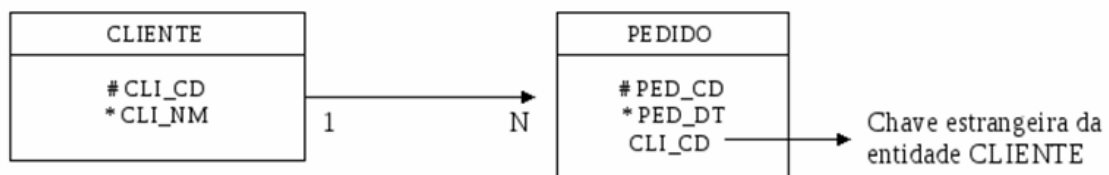
Ex:



**Chave estrangeira (foreign key):** permite o acesso e a validação de outras tabelas. Essa chave permite que se estabeleçam os relacionamentos em um banco de dados. A chave estrangeira deve ser compatível (tipo e tamanho dos campos) com sua correspondente em outra tabela. Ex: Os PEDIDOS se relaciona com os clientes pelo campo CLIENTE na tabela PEDIDO. A chave estrangeira deve ser compatível (tipo e tamanho dos campos) com sua correspondente em outra tabela.

Ex:

Os PEDIDOS se relaciona com os clientes pelo campo CLIENTE na tabela PEDIDO.



## 69) Qual a diferença de false e true?

False é falso (deixa a função falsa)

True é verdadeiro (deixa a função verdadeira)

## 70) Qual utilidade do comando textBox1.Text = "";

Textbox 1 é o text que você quer limpar e text é o que vai estar escrito.



### 71) Porque vocês resolveram fazer esse projeto?

Por que nos interessamos no fato de que poderíamos criar um sistema voltado para padarias, afim de aplicar todos os conhecimentos que iríamos adquirir no decorrer do modulo

#### Comandos aprendidos no access 2013

Qualificador	Significado
CD	Código
NM	Nome
NR	Número
VL	Valor
QT	Quantidade
TX	Taxa ou Porcentagem
DS	Descrição
SG	Sigla
DT	Data
HR	Hora
DH	Data Hora
ID	Identificador
IM	Imagem

#### Comandos utilizados no Pascal:

Begin = Inicio;

End = final;

Write = escreva;

Read = leia;

Writeln = escreva e pule linha;

Readln = leia e pule linha;

Clrscr = clear screen = limpa tela;

If then else = se então senão;

Case of = caso seja;

While = enquanto;

If = se; Then = então;

Else = senão