



XML Documentation

Documentação dos Cadastros de WMS LanLimp

- Este documento descreve como ocorre as principais rotinas de cadastros no WMS da LanLimp dentro de cada rotina prevista no fluxo de processo de entrada de mercadorias.

Elaborado por



Elaborado para





XML Documentation

SUMÁRIO

1. APRESENTAÇÃO	03
2. OBJETIVOS	03
3. Nome da Unidade de Programa identificada.....	03
Cadastro de cliente Unit untCadCliente	03
4. XML Documentation	04

XML Documentation

1. Apresentação

Dia após dia, nós programadores escrevemos códigos e mais códigos, seja implementando uma nova rotina, ou dando manutenção em processos existentes (o que é muito comum, principalmente em sistemas legados).

Já parou para pensar quanto tempo “perdemos” tentando decifrar um código, para então poder dar a manutenção adequada? Entender o “jeito”, o raciocínio e ideia que outro programador teve para implementar determinada rotina, é muito complicado, principalmente quando o código é muito grande, poluído ou muito avançado para o nosso nível.

Muitas vezes essa situação ocorre com o nosso próprio código. Convenhamos, é muito difícil, se não impossível, lembrarmos de cada detalhe que utilizamos na escrita dos códigos. Para isso existem os comentários. Um recurso muito utilizado, que permite descrever rotinas, regras de negócios, ou até mesmo justificar o porque de tal código ser daquela maneira.

2. Objetivo

XML Documentation

O XML Documentation nada mais é do que um padrão muito utilizado para documentar códigos fontes. Este padrão não é algo exclusivo, na verdade surgiu em outras linguagens e devido ao seu grande aceite entre os desenvolvedores, foi introduzido ao RAD Studio também.

3. Nome da Unidade de Programa Identificada

O nome do arquivo fonte que contém as linhas de códigos responsável por realizar a tarefa de Cadastro de empresas é: **untCadCliente**

XML Documentation

4. XML Documentation

Tags

Segue a lista das possíveis *tags* que podem ser utilizadas para formatar a documentação do código fonte.

summary: Descrição da função (cabeçalho);

para: Criação de parágrafos;

C: Deixa o conteúdo com a fonte *courier*, semelhante ao da IDE;

code: Semelhante a *tag* acima, porém pula uma linha antes e depois do conteúdo. Exemplo: `<code>Item := nil</code>;`

remarks: Utilizado para fazer uma observação sobre o uso do código;

param: Utilizado para documentar os parâmetros de uma função. Exemplo: `<param name="NomeDoParametro">Exemplo...</param>;`

see: Referência a um tipo, variável, parâmetro, classe, etc... Exemplo: `<see cref="Referencia"/>;`

returns: Descreve o retorno da função;

exception: Descreve as exceções que podem ser retornadas ao utilizar a função. Exemplo: `<exception cref="EExceptionTypeName"></exception>`. O texto que vai entre ">" e "<" é a descrição, da exceção;

permission: Descreve as permissões que devem ter para utilizar a rotina. Exemplo: `<permission cref="PermissionType"></permission>`. Semelhante a *tag exception*, o que tiver entre ">" e "<" é a descrição.

<file name>	"untCadCliente"
<instruction name>	<summary> Formulário para manutenção do cadastro de clientes. <para> A partir deste formulário é possível executar os procedimentos inerentes a uma manutenção de cadastro, herdando da classe TfrmTpCadastro os seguintes eventos: Consultar, Incluir, Alterar, Excluir, Imprimir ou Exportar Lista. </para> </summary>
<type>	"PAS "
<caller>	caller=" untCadCliente "
<summary>	A partir deste formulário é possível executar os procedimentos inerentes a uma manutenção de cadastro, herdando

XML Documentation

	da classe TfrmTpCadastro os seguintes eventos: Consultar, Incluir, Alterar, Excluir, Imprimir ou Exportar Lista.
<param>	name="Sender" type="TObject"> <para>Parâmetro de entrada referente ao objeto no qual o procedimento está sendo acionado.</para> </param> Procedimento com a finalidade de validar a pré-existência do CNPJ informado pelo usuário e confirmar a manutenção realizada no cadastro de clientes. <param name="Sender" type="TObject"> <para>Parâmetro de entrada referente ao objeto no qual o procedimento está sendo acionado.</para> </param>
<returns>	
<exception>	
<remarks>	<instruction name="qrClienteBeforeOpen"> <summary> Procedimento com a finalidade de montar filtro para seleção dos dados do cliente. </summary> <param name="DataSet" type="TDataSet"> <para>Parâmetro de entrada referente ao objeto no qual o procedimento está sendo acionado.</para> </param> <remarks> Informações complementares: <para>- Filtra por CNPJ, IdCliente ou Razão Social, dependendo do valor informado pelo usuário no campo de pesquisa.</para> <para>- Filtra os clientes "ativos" ou "inativos", dependendo da seleção realizada pelo usuário no checkbox "cbAtivo".</para> </remarks>