

NAME

Blt_TreeDeleteNode – Deletes a node and its descendants.

SYNOPSIS

```
#include <bltTree.h>
```

```
Blt_TreeNode
```

```
Blt_TreeDeleteNode(tree, node)
```

ARGUMENTS

Blt_Tree	<i>tree</i>	(in)	Tree containing the node.
----------	-------------	------	---------------------------

Blt_TreeNode	<i>node</i>	(in)	Node to be deleted.
--------------	-------------	------	---------------------

DESCRIPTION

This procedure deletes a given node and all its descendants from a tree data object.

The arguments are as follows:

<i>tree</i>	The tree containing the parent node.
-------------	--------------------------------------

<i>node</i>	Node to be deleted. The node and its descendant nodes are deleted. Each node's data values are deleted also. The reference count of the Tcl_Obj is decremented.
-------------	---

Since all tree objects must contain at least a root node, the root node itself can't be deleted unless the tree is released and destroyed. Therefore you can clear a tree by deleting its root, but the root node will remain until the tree is destroyed.

RETURNS

Always returns TCL_OK. Errors generated in a notification callbacks are backgrounded (see **Tcl_TreeCreateNotifyHandler**).

EXAMPLE

The following example deletes the root node.

```
Blt_TreeNode root;

root = Blt_TreeRootNode(token);
Blt_TreeDeleteNode(token, root);
```

NOTIFICATIONS

Blt_TreeDeleteNode can trigger tree notify events. You can be notified whenever a node is deleted by using the **Blt_TreeCreateNotifyHandler**. A callback routine is registered that will be automatically invoked whenever a node is deleted via **Blt_TreeDeleteNode** to the tree.

KEYWORDS

tree, token