Geometry: Node

Geometry

Functions

afx_msg void	**OSGeometryUI::CreateNode** (const VARIANT FAR &nNodeNo, const VARIANT FAR &fCoordX, const VARIANT FAR &fCoordY, const VARIANT FAR &fCoordZ) Creates a node with specified coordinates in the structure with the number specified in *nNodeNo*.
afx_msg VARIANT	**OSGeometryUI::AddNode** (const VARIANT FAR &fCoordX, const VARIANT FAR &fCoordY, const VARIANT FAR &fCoordZ) Adds a node with specified coordinates in current model and returns the node number ID automatically assigned with.
afx_msg void	**OSGeometryUI::DeleteNode** (const VARIANT FAR &nNodeNo) Delete a specified node.
afx_msg VARIANT	**OSGeometryUI::GetLastNodeNo** () Returns the node number ID of the last node in the model.
afx_msg VARIANT	**OSGeometryUI::GetNoOfSelectedNodes** () Returns the number of selected node(s).
afx_msg void	**OSGeometryUI::GetSelectedNodes** (VARIANT FAR &naNodeNos, VARIANT FAR &nIsSorted) Returns a list of selected node(s).
afx_msg void	**OSGeometryUI::GetNodeCoordinates** (const VARIANT FAR &nNodeNo, VARIANT FAR &fCoordX, VARIANT FAR &fCoordY, VARIANT FAR &fCoordZ) Returns the coordinates of the specified node.
afx_msg void	**OSGeometryUI::SetNodeCoordinate** (const VARIANT FAR &nNodeNo, const VARIANT FAR &fCoordX, const VARIANT FAR &fCoordY, const VARIANT FAR &fCoordZ) Sets or replaces the coordinate of the *nNodeNo* node.
afx_msg VARIANT	**OSGeometryUI::GetNodeNumber** (const VARIANT FAR &fCoordX, const VARIANT FAR &fCoordY, const VARIANT FAR &fCoordZ) Returns the number ID of the node at specified coordinates.
afx_msg VARIANT	**OSGeometryUI::GetNodeDistance** (const VARIANT FAR &nNodeNoA, const VARIANT FAR &nNodeNoB) Returns the distance between two specified nodes.
afx_msg VARIANT	**OSGeometryUI::SelectMultipleNodes** (const VARIANT FAR &naNodeNos) Selects multiple node(s) in current model.
afx_msg VARIANT	**OSGeometryUI::SelectNode** (const VARIANT FAR &nNodeNo) Selects the specified node in current model.
afx_msg void	**OSGeometryUI::AddMultipleNodes** (const VARIANT FAR &faCoordinates)

	Adds multiple nodes with specified node coordinates array.
afx_msg void	**OSGeometryUI::CreateMultipleNodes** (const VARIANT FAR &nNodeIdArray, const VARIANT FAR &dCoordArray) Create multiple nodes with specified node number ID(s).
afx_msg VARIANT	**OSGeometryUI::GetNodeCount** () Returns the total number of nodes in the current model. Note: Count of nodes can vary depending upon the flag set for consideration of hidden entities (see **OSGeometryUI::SetFlagForHiddenEntities**).
afx_msg void	**OSGeometryUI::GetNodeList** (VARIANT FAR &nNodeList) Returns the list of all the node number ID(s) in the current model. Note: List of nodes can be different depending upon the flag set for consideration of hidden entities (see **OSGeometryUI::SetFlagForHiddenEntities**).
afx_msg VARIANT	**OSGeometryUI::GetNodeIncidence** (const VARIANT FAR &nNodeNo, VARIANT FAR &fCoordX, VARIANT FAR &fCoordY, VARIANT FAR &fCoordZ) Return the coordinates of the specified node.
afx_msg void	**OSGeometryUI::ClearNodeSelection** () Unselect all the node item(s).
afx_msg void	**OSGeometryUI::SetNodeUniqueID** (const VARIANT FAR &nNodeNo, const VARIANT FAR &szName) Assigns an unique string ID (GUID) to specified node.
afx_msg VARIANT	**OSGeometryUI::GetNodeUniqueID** (const VARIANT FAR &nNodeNo) Returns the unique string ID (GUID) for specified node.
afx_msg VARIANT	**OSGeometryUI::GetNodeIncidence_CIS2** (const VARIANT FAR &nNodeNo, VARIANT FAR &szName, VARIANT FAR &fCoordX, VARIANT FAR &fCoordY, VARIANT FAR &fCoordZ) Return the coordinates of the specified node.
afx_msg VARIANT	**OSGeometryUI::IsOrphanNode** (const VARIANT FAR &varNodeNo) Returns whether the specified node is orphan node or not?
afx_msg VARIANT	**OSGeometryUI::MergeNodes** (const VARIANT FAR &varNodeNo, const VARIANT FAR &nNodeIdArray) Merges multiple nodes to a single node. All members and elements which were connected to the nodes will be reconnected to the specified node to retain.

Detailed Description

These functions are related to operations of creating, adding, getting and deleting node(s).

Function Documentation

◆ AddMultipleNodes()

void OSGeometryUI::AddMultipleNodes (const VARIANT FAR & faCoordinates) private

Adds multiple nodes with specified node coordinates array.

Parameters

[in] **faCoordinates** VARIANT array of m * 3 dimension containing coordinates of nodes: [X_i, Y_i, Z_i].

C++ Syntax

```
// Add multiple nodes.
OSGeometryUI::AddMultipleNodes(faCoordinates);
```

VBA Syntax

```
' Add multiple nodes.
OSGeometryUI.AddMultipleNodes(faCoordinates)
```

See also

OSGeometryUI::CreateNode

OSGeometryUI::AddNode

OSGeometryUI::DeleteNode

◆ AddNode()

VARIANT OSGeometryUI::AddNode (const VARIANT FAR & fCoordX,
const VARIANT FAR & fCoordY,
const VARIANT FAR & fCoordZ) private

Adds a node with specified coordinates in current model and returns the node number ID automatically assigned with.

Parameters

[in] **fCoordX** Nodal coordinate X in GLOBAL.

[in] **fCoordY** Nodal coordinate Y in GLOBAL.

[in] **fCoordZ** Nodal coordinate Z in GLOBAL.

Return values

<Val> Node number ID assigned to this created node.

0 OK.

-2004 Unable to add Node.

C++ Syntax

```
double fCoordX = 3.0;
double fCoordY = 2.0;
double fCoordZ = 3.0;
// Add a node (3.0,2.0,3.0).
VARIANT nNodeNo = OSGeometryUI::AddNode(fCoordX, fCoordY, fCoordZ);
```

VBA Syntax

```
Dim fCoordX As Double = 3.0
Dim fCoordY As Double = 2.0
Dim fCoordZ As Double = 3.0
' Add a node (3.0,2.0,3.0).
Dim nNodeNo As VARIANT = OSGeometryUI.AddNode(fCoordX, fCoordY, fCoordZ)
```

Remarks

The difference between **OSGeometryUI::CreateNode** and **OSGeometryUI::AddNode** is the former has an option to label the node with any user-defined number.

Note

If geometry update flag is set, this function will update the node coordinate values.

See also

OSGeometryUI::CreateNode

OSGeometryUI::DeleteNode

OSGeometryUI::AddMultipleNodes

◆ ClearNodeSelection()

void OSGeometryUI::ClearNodeSelection ( ) private

Unselect all the node item(s).

C++ Syntax

```
// Clear Node Selection
OSGeometryUI::ClearNodeSelection();
```

VBA Syntax

```
' Clear Node Selection
OSGeometryUI.ClearNodeSelection()
```

See also

OSGeometryUI::SelectNode

OSGeometryUI::SelectMultipleNodes

OSGeometryUI::GetSelectedNodes

◆ CreateMultipleNodes()

void OSGeometryUI::CreateMultipleNodes (const VARIANT FAR & nNodeIdArray,
const VARIANT FAR & dCoordArray) private

Create multiple nodes with specified node number ID(s).

Parameters

[in] **nNodeIdArray** Integer array of 1 dimension containing m node IDs:[ID_i]

[in] **dCoordArray** Double array of 2 dimensions containing M*3 elements i.e node coordinate: [X_i, Y_i, Z_i].

C++ Syntax

```
// Create Multiple Nodes - Setting Identical check for Node Entity as 0 (=FALSE). So,
       while creating multiple nodes in the subsequent API call, it will not check for
// existing identical nodes while creating nodes resulting in faster operation. If, check
       is set to 1 (=TRUE), it will check for identical nodes
// in the subsequent add multiple node API which will take longer time.

int nNodeIdArray[] = {91, 92, 93};
int dCoordArray[][3] = {{3, 5, 15}, {15, 25, 27}, {47, 27, 29}};
OSGeometryUI::SetCheckForIdenticalEntity(STAADEntityType.NodeEntity, 0);
OSGeometryUI::CreateMultipleNodes(nNodeIdArray, dCoordArray);
```

C# Syntax

```
// Create Multiple Nodes - Setting Identical check for Node Entity as 0 (=FALSE). So,
       while creating multiple nodes in the subsequent API call, it will not check for
// existing identical nodes while creating nodes resulting in faster operation. If, check
       is set to 1 (=TRUE), it will check for identical nodes
// in the subsequent add multiple node API which will take longer time.

int[] nNodeIdArray = new int[3] {91, 92, 93};
int[,] dCoordArray = new int[3, 3] {{3, 5, 15}, {15, 25, 27}, {47, 27, 29}};
object objectNodeCoordArray = nNodeCoordArray as object;
object objectNodeIdArray = nNodeIdArray as object;
m_OStd.Geometry.SetCheckForIdenticalEntity(STAADEntityType.NodeEntity, 1);
int RetValue = m_OStd.Geometry.CreateMultipleNodes(nNodeIdArray, dCoordArray);
```

VBA Syntax

```
// Create Multiple Nodes - Setting Identical check for Node Entity as 0 (=FALSE). So,
       while creating multiple nodes in the subsequent API call, it will not check for
// existing identical nodes while creating nodes resulting in faster operation. If, check
       is set to 1 (=TRUE), it will check for identical nodes
// in the subsequent add multiple node API which will take longer time.

Dim nNodeIdArray(1) As Integer
nNodeIdArray(0) = 91
nNodeIdArray(1) = 92
Dim dCoordArray(1,2) As Integer
dCoordArray(0,0) = 3
dCoordArray(0,1) = 5
dCoordArray(0,2) = 15
```

```
dCoordArray(1,0) = 15
dCoordArray(1,1) = 25
dCoordArray(1,2) = 27
objOpenStaad.Geometry.SetCheckForIdenticalEntity(0, 1);
objOpenStaad.Geometry.CreateMultipleNodes(nNodeIdArray,dCoordArray)
```

See also

COSGeometry::SetCheckForIdenticalEntity

COSGeometry::CreateNode

CreateNode()

void OSGeometryUI::CreateNode (const VARIANT FAR & nNodeNo,
const VARIANT FAR & fCoordX,
const VARIANT FAR & fCoordY,
const VARIANT FAR & fCoordZ) private

Creates a node with specified coordinates in the structure with the number specified in *nNodeNo*.

Parameters

[in] **nNodeNo** Node number ID to be assigned to the newly created node.

[in] **fCoordX** Nodal coordinate X in GLOBAL.

[in] **fCoordY** Nodal coordinate Y in GLOBAL.

[in] **fCoordZ** Nodal coordinate Z in GLOBAL.

C++ Syntax

```
double fCoordX = 3.0;
double fCoordY = 2.0;
double fCoordZ = 3.0;
// Add a node (3.0,2.0,3.0), call it node # 10.
OSGeometryUI::CreateNode(10, fCoordX, fCoordY, fCoordZ);
```

VBA Syntax

```
Dim fCoordX As Double = 3.0
Dim fCoordY As Double = 2.0
Dim fCoordZ As Double = 3.0
' Add a node (3.0,2.0,3.0), call it node # 10.
OSGeometryUI.CreateNode(10, fCoordX, fCoordY, fCoordZ)
```

Remarks

The difference between **OSGeometryUI::CreateNode** and **OSGeometryUI::AddNode** is the former has an option to label the node with any user-defined number.

See also

OSGeometryUI::AddNode

OSGeometryUI::DeleteNode

OSGeometryUI::AddMultipleNodes

◆ DeleteNode()

void OSGeometryUI::DeleteNode (const VARIANT FAR & nNodeNo) private

Delete a specified node.

Parameters

[in] **nNodeNo** Node number ID.

C++ Syntax

```
long nNodeNo = 25;
//Delete node #25
OSGeometryUI::DeleteNode(nNodeNo);
```

VBA Syntax

```
Dim nNodeNo As Long = 25
' Delete node #25.
OSGeometryUI.DeleteNode(nNodeNo)
```

See also

OSGeometryUI::CreateNode

OSGeometryUI::AddNode

OSGeometryUI::AddMultipleNodes

◆ GetLastNodeNo()

VARIANT OSGeometryUI::GetLastNodeNo ( ) private

Returns the node number ID of the last node in the model.

Return values

<Val> The number of the highest node number ID in the model .

-1 General error.

C++ Syntax

```
// Get last node #.
VARIANT LastNodeNo = OSGeometryUI::GetLastNodeNo();
```

VBA Syntax

```
' Get last node #.
Dim LastNodeNo As VARIANT = OSGeometryUI.GetLastNodeNo()
```

◆ GetNodeCoordinates()

void OSGeometryUI::GetNodeCoordinates (const VARIANT FAR & nNodeNo,
VARIANT FAR & fCoordX,
VARIANT FAR & fCoordY,
VARIANT FAR & fCoordZ) private

Returns the coordinates of the specified node.

Parameters

[in] **nNodeNo** Node number ID.

[out] **fCoordX** Nodal coordinate X in GLOBAL.

[out] **fCoordY** Nodal coordinate Y in GLOBAL.

[out] **fCoordZ** Nodal coordinate Z in GLOBAL.

C++ Syntax

```
// Get coordinate of node # 10 to dX, dY and dZ.
OSGeometryUI::GetNodeCoordinates(10, &fCoordX, &fCoordY, &fCoordZ);
```

VBA Syntax

```
' Get coordinate of node # 10 to dX, dY and dZ.
OSGeometryUI.GetNodeCoordinates(10, &fCoordX, &fCoordY, &fCoordZ)
```

See also

OSGeometryUI::GetNodeIncidence

◆ GetNodeCount()

VARIANT OSGeometryUI::GetNodeCount ( )

private

Returns the total number of nodes in the current model.

Note: Count of nodes can vary depending upon the flag set for consideration of hidden entities (see **OSGeometryUI::SetFlagForHiddenEntities**).

Return values

<Val> The total number of node(s).

C++ Syntax

```
// Count for the nodes.
VARIANT lNodeCount = OSGeometryUI::GetNodeCount();
```

VBA Syntax

```
' Count for the nodes.
Dim lNodeCount As VARIANT = OSGeometryUI.GetNodeCount()
```

See also

OSGeometryUI::GetNodeNumber

OSGeometryUI::GetNodeList

OSGeometryUI::GetFlagForHiddenEntities

OSGeometryUI::SetFlagForHiddenEntities

◆ GetNodeDistance()

VARIANT OSGeometryUI::GetNodeDistance (const VARIANT FAR & nNodeNoA,
const VARIANT FAR & nNodeNoB) private

Returns the distance between two specified nodes.

Parameters

[in] **nNodeNoA** Number ID of one node (Type: Long).

[in] **nNodeNoB** Number ID of one of the other node (Type: Long).

Return values

<Val> The distances in double.

-1 General error.

-2001 Cannot find Node < **nNodeNoA** > or < **nNodeNoB** >.

C++ Syntax

```
// Get the distance between node 2 and 3.
VARIANT NodeDist = OSGeometryUI::GetNodeDistance(2, 3);
```

VBA Syntax

```
' Get the distance between node 2 and 3.
Option Explicit

Sub Main
    Dim objOpenStaad As Object
    Dim stdFile As String
    Dim nNodeNoA As Long
    Dim nNodeNoB As Long
    Dim NodeDist As Double

    Set objOpenStaad = GetObject(,"StaadPro.OpenSTAAD")
    objOpenStaad.GetSTAADFile stdFile, "TRUE"
    If stdFile="" Then
        MsgBox"Bad"
         Set objOpenStaad = Nothing
         Exit Sub
     End If

    nNodeNoA = 2
    nNodeNoB = 3
    NodeDist = objOpenStaad.Geometry.GetNodeDistance(nNodeNoA, nNodeNoB)

    Set objOpenStaad = Nothing
End Sub
```

◆ GetNodeIncidence()

VARIANT OSGeometryUI::GetNodeIncidence (const VARIANT FAR & nNodeNo,
VARIANT FAR & fCoordX,
VARIANT FAR & fCoordY,
VARIANT FAR & fCoordZ) private

Return the coordinates of the specified node.

Parameters

[in] **nNodeNo** Node number ID (Type: Long).

[out] **fCoordX** Nodal coordinate X in GLOBAL (Type: Double).

[out] **fCoordY** Nodal coordinate Y in GLOBAL (Type: Double).

[out] **fCoordZ** Nodal coordinate Z in GLOBAL (Type: Double).

Return values

0 OK.

-1 General error.

-2001 Cannot find Node < **nNodeNo** >.

C++ Syntax

```
// Get coordinate of node # 10 to fCoordX, fCoordY and fCoordZ, and store the returned
       error code
VARIANT RetVal = OSGeometryUI::GetNodeIncidence(10, &fCoordX, &fCoordY, &fCoordZ);
```

VBA Syntax

```
' Get coordinate of node # 10 to fCoordX, fCoordY and fCoordZ, and store the returned
       error code
Option Explicit

Sub Main
    Dim objOpenStaad As Object
    Dim stdFile As String
    Dim nNodeNo As Long
    Dim fCoordX As Double
    Dim fCoordY As Double
    Dim fCoordZ As Double
    Dim RetVal as Long

    Set objOpenStaad = GetObject(,"StaadPro.OpenSTAAD")
    objOpenStaad.GetSTAADFile stdFile, "TRUE"
    If stdFile="" Then
        MsgBox"Bad"
         Set objOpenStaad = Nothing
         Exit Sub
     End If

    nNodeNo = 10
    RetVal = objOpenStaad.Geometry.GetNodeIncidence(nNodeNo, fCoordX, fCoordY, fCoordZ)

    Set objOpenStaad = Nothing
```

```
End Sub
```

See also

OSGeometryUI::GetNodeCoordinates

◆ GetNodeIncidence_CIS2()

VARIANT OSGeometryUI::GetNodeIncidence_CIS2 (const VARIANT FAR & nNodeNo,
VARIANT FAR & szName,
VARIANT FAR & fCoordX,
VARIANT FAR & fCoordY,
VARIANT FAR & fCoordZ) private

Return the coordinates of the specified node.

Parameters

[in] **nNodeNo** Node number ID.

[out] **szName** (LPCTSTR) unique string ID.

[out] **fCoordX** Nodal coordinate X in GLOBAL.

[out] **fCoordY** Nodal coordinate Y in GLOBAL.

[out] **fCoordZ** Nodal coordinate Z in GLOBAL.

Return values

0 OK.

-1 General error.

-2001 Cannot find Node < **nNodeNo** >.

C++ Syntax

```
// Get coordinate of node # 10 to fCoordX, fCoordY and fCoordZ, and store the returned
       error code
VARIANT RetVal = OSGeometryUI::GetNodeIncidence_CIS2(10, &szName, &fCoordX, &fCoordY,
       &fCoordZ);
```

VBA Syntax

```
' Get coordinate of node # 10 to fCoordX, fCoordY and fCoordZ, and store the returned
       error code
Dim RetVal As VARIANT = OSGeometryUI.GetNodeIncidence_CIS2(10, &szName, &fCoordX,
       &fCoordY, &fCoordZ)
```

See also

OSGeometryUI::GetNodeCoordinates

◆ GetNodeList()

void OSGeometryUI::GetNodeList (VARIANT FAR & nNodeList) private

Returns the list of all the node number ID(s) in the current model.
Note: List of nodes can be different depending upon the flag set for consideration of hidden entities (see **OSGeometryUI::SetFlagForHiddenEntities**).

Parameters

[out] **nNodeList** VARIANT array of LONG type, for storing returned node number ID(s).

C++ Syntax

```
// Get nodes list
OSGeometryUI::GetNodeList(&nNodeList);
```

VBA Syntax

```
' Get nodes list.
OSGeometryUI.GetNodeList(&nNodeList)
```

See also

OSGeometryUI::GetNodeCount

OSGeometryUI::GetNodeNumber

OSGeometryUI::GetFlagForHiddenEntities

OSGeometryUI::SetFlagForHiddenEntities

◆ GetNodeNumber()

VARIANT OSGeometryUI::GetNodeNumber (const VARIANT FAR & fCoordX,
const VARIANT FAR & fCoordY,
const VARIANT FAR & fCoordZ) private

Returns the number ID of the node at specified coordinates.

Parameters

[in] **fCoordX** New coordinate X in GLOBAL. (Type: Double)

[in] **fCoordY** New coordinate Y in GLOBAL. (Type: Double)

[in] **fCoordZ** New coordinate Z in GLOBAL. (Type: Double)

Return values

<Val> Node number ID. (Type: Long)

-2001 Cannot find Node with coordinates < **fCoordX** >, < **fCoordY** > and < **fCoordZ** >.

-1 General error.

C++ Syntax

```
// Get the node number of (3.0, 2.0, 3.0).
VARIANT nNodeNo = OSGeometryUI::GetNodeNumber(5, 3.0, 2.0, 3.0);
```

VBA Syntax

```
Option Explicit

Sub Main
    Dim objOpenStaad As Object
    Dim stdFile As String
    Dim X As Double
    Dim Y As Double
    Dim Z As Double
    Dim NodeNo As Long

    Set objOpenStaad = GetObject(,"StaadPro.OpenSTAAD")
    objOpenStaad.GetSTAADFile stdFile, "TRUE"
    If stdFile="" Then
        MsgBox"Bad"
         Set objOpenStaad = Nothing
         Exit Sub
     End If
    X = 5.0
    Y = 2.0
    Z = 3.0
    NodeNo = objOpenStaad.Geometry.GetNodeNumber(X, Y, Z)
    MsgBox"Node Number: '" & NodeNo & "'."
    Set objOpenStaad = Nothing
End Sub
```

See also

OSGeometryUI::GetNodeCount

OSGeometryUI::GetNodeList

◆ GetNodeUniqueID()

VARIANT OSGeometryUI::GetNodeUniqueID (const VARIANT FAR & nNodeNo) private

Returns the unique string ID (GUID) for specified node.

Parameters

[in] **nNodeNo** Node number ID.

Return values

<VARIANT> Unique string ID for specified node.

The API would return an empty string if specified node < **nNodeNo** > is *not* found

C++ Syntax

```
// Get the unique ID of node #3.
VARIANT szName = OSGeometryUI::GetNodeUniqueID(3);
```

VBA Syntax

```
' Get the unique ID of node #3.
Dim szName As VARIANT = OSGeometryUI.GetNodeUniqueID(3)
```

See also

OSGeometryUI::SetNodeUniqueID

◆ GetNoOfSelectedNodes()

VARIANT OSGeometryUI::GetNoOfSelectedNodes ( ) private

Returns the number of selected node(s).

Return values

The number of selected node(s).

C++ Syntax

```
// Counts for the total number of node(s) selected.
VARIANT NoOfSelectedNodes = OSGeometryUI::GetNoOfSelectedNodes();
```

VBA Syntax

```
' Counts for the total number of node(s) selected.
Dim NoOfSelectedNodes As VARIANT = OSGeometryUI.GetNoOfSelectedNodes()
```

See also

OSGeometryUI::SelectNode

OSGeometryUI::ClearNodeSelection

OSGeometryUI::SelectMultipleNodes

OSGeometryUI::GetSelectedNodes

◆ GetSelectedNodes()

void OSGeometryUI::GetSelectedNodes (VARIANT FAR & naNodeNos,
VARIANT FAR & nIsSorted) private

Returns a list of selected node(s).

Parameters

[out] **naNodeNos** Returned selected node number ID(s) VARIANAT array.

[in] **nIsSorted** The order of the selection(s): in sorted order (= 1 or TRUE), in the order of selection (= 0 or FALSE).

C++ Syntax

```
// Get selected node list in the order of selection.
OSGeometryUI::GetSelectedNodes(&naNodeNos, 0);
```

VBA Syntax

```
' Get selected node list in the order of selection.
OSGeometryUI.GetSelectedNodes(&naNodeNos, 0)
```

See also

OSGeometryUI::SelectNode

OSGeometryUI::ClearNodeSelection

OSGeometryUI::SelectMultipleNodes

OSGeometryUI::GetNoOfSelectedNodes

◆ IsOrphanNode()

VARIANT OSGeometryUI::IsOrphanNode (const VARIANT FAR & varNodeNo) private

Returns whether the specified node is orphan node or not?

Parameters

[in] **varNodeNo** Number ID of the node.

Return values

1 True;

0 False;

C++ Syntax

```
// specified node #3 is orphan?
VARIANT RetVal = OSGeometryUI::IsOrphanNode(3);
```

VBA Syntax

```
' specified node #3 is orphan?
Dim RetVal As VARIANT = OSGeometryUI.IsOrphanNode(3)
```

◆ MergeNodes()

VARIANT OSGeometryUI::MergeNodes (const VARIANT FAR & varNodeNo,
const VARIANT FAR & nNodeIdArray) private

Merges multiple nodes to a single node. All members and elements which were connected to the nodes will be reconnected to the specified node to retain.

Parameters

[in] **varNodeNo** node number to be assigned to the merged node (must be present in nNodeIdArray) (type - Long/Integer).

[in] **nNodeIdArray** array of node numbers to be merged (must have more than one node) (type - Long/Integer).

Return values

1 if merging is successful.

0 if merging is unsuccessful.

VBA Syntax

```
' Merge Node # 1.
Option Explicit

Sub Main
    Dim objOpenStaad As Object
    Dim stdFile As String
    Dim varNodes(1) As Long
    Dim nResult As long

    Set objOpenStaad = GetObject(,"StaadPro.OpenSTAAD")
    objOpenStaad.GetSTAADFile stdFile, "TRUE"
    varNodes(0) = 1
    varNodes(1) = 4
    nResult = objOpenStaad.Geometry.MergeNodes(1,varNodes)

 ' process the return value
       If nResult > 0 Then
         MsgBox"Merge node is successful"
       Else
         MsgBox"Merge node is unsuccessful"
       End If

       Set objOpenStaad = Nothing
End Sub
```

◆ SelectMultipleNodes()

VARIANT OSGeometryUI::SelectMultipleNodes (const VARIANT FAR & naNodeNos) private

Selects multiple node(s) in current model.

Parameters

[in] **naNodeNos** Number number ID(s) VARIANT array (type - Long/Integer).

Return values

0 Failed

1 Succeeded

C++ Syntax

```
// Select multiple nodes.
OSGeometryUI::SelectMultipleNodes(naNodeNos);
```

VBA Syntax

```
Option Explicit
    Sub Main
        Dim objOpenStaad As Object
        Dim stdFile As String
        'launch STAAD.Pro application and open "Verification Models\02 Trusses\Roof Truss
       Axial Forces.STD" file from the Samples folder
        Set objOpenStaad = GetObject(,"StaadPro.OpenSTAAD")
        objOpenStaad.GetSTAADFile stdFile, "TRUE"
        If stdFile="" Then
            MsgBox"Bad"
            Set objOpenStaad = Nothing
            Exit Sub
        End If
        Dim Objs(1) As Long
        Objs(0)=5
        Objs(1)=10
        Dim bRes As Boolean
        bRes = objOpenStaad.Geometry.SelectMultipleNodes( Objs)
        If bRes Then
            MsgBox"Success"
        Else
            MsgBox"Failed"
        End If
    End Sub
```

See also

OSGeometryUI::SelectNode

OSGeometryUI::ClearNodeSelection

OSGeometryUI::GetNoOfSelectedNode

OSGeometryUI::GetSelectedNodes

◆ SelectNode()

VARIANT OSGeometryUI::SelectNode (const VARIANT FAR & nNodeNo) private

Selects the specified node in current model.

Parameters

[in] **nNodeNo** Node number ID.

C++ Syntax

```
// Select node # 3.
OSGeometryUI::SelectNode(3);
```

VBA Syntax

```
' Select node # 3.
OSGeometryUI.SelectNode(3)
```

See also

OSGeometryUI::SelectMultipleNodes

OSGeometryUI::ClearNodeSelection

OSGeometryUI::GetNoOfSelectedNode

OSGeometryUI::GetSelectedNodes

◆ SetNodeCoordinate()

void OSGeometryUI::SetNodeCoordinate (const VARIANT FAR & nNodeNo,
const VARIANT FAR & fCoordX,
const VARIANT FAR & fCoordY,
const VARIANT FAR & fCoordZ) private

Sets or replaces the coordinate of the *nNodeNo* node.

Parameters

[in] **nNodeNo** Node number ID.

[in] **fCoordX** New coordinate X in GLOBAL.

[in] **fCoordY** New coordinate Y in GLOBAL.

[in] **fCoordZ** New coordinate Z in GLOBAL.

C++ Syntax

```
// Change the coordinate of node #5 to be (3.0, 2.0, 3.0).
OSGeometryUI::SetNodeCoordinate(5, 3.0, 2.0, 3.0);
```

VBA Syntax

```
' Change the coordinate of node #5 to be (3.0, 2.0, 3.0).
OSGeometryUI.SetNodeCoordinate(5, 3.0, 2.0, 3.0)
```

◆ SetNodeUniqueID()

void OSGeometryUI::SetNodeUniqueID (const VARIANT FAR & nNodeNo,
const VARIANT FAR & szName) private

Assigns an unique string ID (GUID) to specified node.

Parameters

[in] **nNodeNo** Number ID of the node to be assigned.

[in] **szName** (LPCTSTR) unique string ID.

C++ Syntax

```
// Set "NODESP" to node #3.
OSGeometryUI::SetNodeUniqueID(3, (LPCTSTR)"NODESP");
```

VBA Syntax

```
' Set "NODESP" to node #3.
OSGeometryUI.SetNodeUniqueID(3, "NODESP")
```

See also

OSGeometryUI::GetNodeUniqueID