

Load: Load Envelopes

Load

Functions

afx_msg VARIANT	OSLoadUI::CreateLoadEnvelop (const VARIANT FAR &varEnvNo, const VARIANT FAR &varEnvType, const VARIANT FAR &varLoadCaseList)
Creates a Load Envelop with specified primary load case(s) and envelop type.	
afx_msg VARIANT	OSLoadUI::AddLoadCasesToEnvelop (const VARIANT FAR &varEnvNo, const VARIANT FAR &varLoadCaseList)
Adds a list of primary load case(s) to an existed load envelop.	
afx_msg VARIANT	OSLoadUI::RemoveLoadCasesFromEnvelop (const VARIANT FAR &varEnvNo, const VARIANT FAR &varLoadCaseList)
Removes a list of primary load case(s) from an existed load envelop.	
afx_msg VARIANT	OSLoadUI::DeleteLoadEnvelop (const VARIANT FAR &varEnvNo)
Deletes a specified load envelop.	
afx_msg long	OSLoadUI::GetEnvelopeCount ()
Returns number of Envelopes defined.	
afx_msg long	OSLoadUI::GetLoadEnvelopeDetails (long EnvNo, long &EnvelopeType, long &NumberofLoadsInEnvelope)
Gets the type of Envelope and number of primary load case(s) present in the entered load envelope.	
afx_msg long	OSLoadUI::GetLoadListfromLoadEnvelope (long EnvNo, VARIANT &varLoadCaseList)
Gets the list of primary load case reference Ids present in the load envelope passed.	
afx_msg long	OSLoadUI::GetEnvelopeIDs (VARIANT &varLoadCaseList)
Gets the list of Loads Envelope IDs present in the staad file.	

Detailed Description

These functions are related to Load Envelopes

Function Documentation

◆ AddLoadCasesToEnvelop()

```
VARIANT OSLoadUI::AddLoadCasesToEnvelop ( const VARIANT FAR & varEnvNo,  
                                         const VARIANT FAR & varLoadCaseList )
```

Adds a list of primary load case(s) to an existed load envelop.

Parameters

[in] **varEnvNo** Load Envelop reference ID.
[in] **varLoadCaseList** (Primary) load case(s) reference ID(s) VARIANT array.

Return values

1 OK.
0 General error.

C++ Syntax

```
// Add a list of primary load case(s) into Load Envelop #2  
VARIANT RetVal = OSLoadUI::AddLoadCasesToEnvelop(2, varLoadCaseList);
```

VBA Syntax

```
' Add a list of primary load case(s) into Load Envelop #2  
Dim RetVal As VARIANT = OSLoadUI.AddLoadCasesToEnvelop(2, varLoadCaseList)
```

See also

[OSLoadUI::CreateLoadEnvelop](#)
[OSLoadUI::RemoveLoadCasesFromEnvelop](#)
[OSLoadUI::DeleteLoadEnvelop](#)

◆ [CreateLoadEnvelop\(\)](#)

```
VARIANT OSLoadUI::CreateLoadEnvelop ( const VARIANT FAR & varEnvNo,
                                         const VARIANT FAR & varEnvType,
                                         const VARIANT FAR & varLoadCaseList )
```

Creates a Load Envelop with specified primary load case(s) and envelop type.

Parameters

[in] varEnvNo Load Envelop reference ID(Type: Integer/Long).

[in] varEnvType Type of the load envelop(Type: Integer/Long):

Value	Load Envelop Type
0	NONE
1	STRESS
2	SERVICEABILITY
3	COLUMN
4	CONNECTION
5	STRENGTH
6	TEMPORARY

[in] varLoadCaseList (Primary) load case(s) reference ID(s) VARIANT array(Type: Integer/Long or Integer/Long Array). For additional information, please refer to TR.40 (TR.40 Load Envelope) of the Technical Reference manual.

Return values

1/True OK.

0/False General error.

C++ Syntax

```
// Add a list of primary load case(s) into #2 load envelop (type of STRESS).
VARIANT RetVal = OSLoadUI::CreateLoadEnvelop(2, 1, varLoadCaseList);
```

VBA Syntax

```
Option Explicit
Sub Main
    Dim objOpenStaad As Object
    Dim stdFile As String
    Set objOpenStaad = GetObject(, "StaadPro.OpenSTAAD")
    objOpenStaad.GetSTAADFfile stdFile, "TRUE"
    If stdFile="" Then
        MsgBox "Bad"
        Set objOpenStaad = Nothing
        Exit Sub
    End If
    Dim LoadCaseList(1) As Integer
    Dim EnvType As Integer
```

```
Dim EnvNo As Integer
LoadCaseList(0) = 1
LoadCaseList(1) = 3
EnvType = 1
EnvNo = 4
Dim res As Boolean
'Add a list of primary load case(s) into #4 load envelop (type of STRESS).
res = objOpenStaad.Load.CreateLoadEnvelop(EnvNo,EnvType,LoadCaseList)
If res Then
    MsgBox"CreateLoadEnvelop Success"
Else
    MsgBox"CreateLoadEnvelop With Error"
End If
Set objOpenStaad = Nothing
End Sub
```

See also

- [OSLoadUI::AddLoadCasesToEnvelop](#)
- [OSLoadUI::RemoveLoadCasesFromEnvelop](#)
- [OSLoadUI::DeleteLoadEnvelop](#)

◆ [DeleteLoadEnvelop\(\)](#)

VARIANT OSLoadUI::DeleteLoadEnvelop (const VARIANT FAR & **varEnvNo**)

Deletes a specified load envelop.

Parameters

[in] **varEnvNo** Load Envelop reference ID.

Return values

1 OK.

0 General error.

C++ Syntax

```
// Delete load envelop #2
VARIANT RetVal = OSLoadUI::DeleteLoadEnvelop(2);
```

VBA Syntax

```
' Delete load envelop #2
Dim RetVal As VARIANT = OSLoadUI.DeleteLoadEnvelop(2)
```

See also

[OSLoadUI::CreateLoadEnvelop](#)
[OSLoadUI::AddLoadCasesToEnvelop](#)
[OSLoadUI::RemoveLoadCasesFromEnvelop](#)

◆ [GetEnvelopeCount\(\)](#)

```
long OSLoadUI::GetEnvelopeCount( )
```

Returns number of Envelopes defined.

Return values

Long/Integer Total Number of load Envelopes present.

C++ Syntax

```
long RetVal = OSLoadUI::GetEnvelopeCount();
```

VBA Syntax

```
Option Explicit
Sub Main
    Dim objOpenStaad As Object
    Dim stdFile As String
    Dim nEnvelopes As Long
    Set objOpenStaad = GetObject(, "StaadPro.OpenSTAAD")
    objOpenStaad.GetSTAADFile stdFile, "TRUE"

    If stdFile="" Then
        MsgBox"Bad"
        Set objOpenStaad = Nothing
        Exit Sub
    End If
    nEnvelopes = objOpenStaad.Load.GetEnvelopeCount()
    MsgBox"Number of Envelopes: '" & nEnvelopes & "'."
    Set objOpenStaad = Nothing
End Sub
```

See also

[OSLoadUI::CreateLoadEnvelop](#)
[OSLoadUI::AddLoadCasesToEnvelop](#)
[OSLoadUI::RemoveLoadCasesFromEnvelop](#)
[OSLoadUI::GetLoadEnvelopeDetails](#)
[OSLoadUI::GetLoadListfromLoadEnvelope](#)

◆ [GetEnvelopeIDs\(\)](#)

```
long OSLoadUI::GetEnvelopeIDs ( VARIANT & varLoadCaseList )
```

Gets the list of Loads Envelope IDs present in the staad file.

Parameters

[out] **varEnvelopeIDList** Envelope ID(s) VARIANT array(Integer/Long Array). Use [OSLoadUI::GetEnvelopeCount](#) to set size of Array.

Return values

- 1 OK.
- 1 Unsuccessful.

C++ Syntax

```
long RetVal = OSLoadUI::GetEnvelopeIDs(varEnvelopeIDList);
```

- **VBA Syntax**

```
Option Explicit
Sub Main
    Dim objOpenStaad As Object
    Dim stdFile As String
    Dim varRetVal As Long

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

    Dim nSize As Long
    nSize = objOpenStaad.Load.GetEnvelopeCount()
    Dim IDList() As Long
    ReDim IDList(nSize - 1)
    varRetVal = objOpenStaad.Load.GetEnvelopeIDs(IDList)
    ' process the return value
    If (varRetVal = 1) Then
        MsgBox"1st envelope ID is '" & IDList(0) & "'."
    Else
        MsgBox"unsuccessful"
    End If
    Set objOpenStaad = Nothing
End Sub
```

See also

- [OSLoadUI::CreateLoadEnvelop](#)
- [OSLoadUI::AddLoadCasesToEnvelop](#)
- [OSLoadUI::RemoveLoadCasesFromEnvelop](#)

[**OSLoadUI::GetEnvelopeCount**](#)

[**OSLoadUI::GetLoadEnvelopeDetails**](#)

- ◆ [**GetLoadEnvelopeDetails\(\)**](#)

```
long OSLoadUI::GetLoadEnvelopeDetails ( long EnvNo,
                                         long & EnvelopeType,
                                         long & NumberofLoadCasesInEnvelope )
```

Gets the type of Envelope and number of primary load case(s) present in the entered load envelope.

Parameters

[in] **EnvNo** Load Envelope reference ID.(Long)

[out] **EnvelopeType** Type of Load Envelope.(Long)

Value	Load Envelope Type
0	NONE
1	STRESS
2	SERVICEABILITY
3	COLUMN
4	CONNECTION
5	STRENGTH
6	TEMPORARY

[out] **NumberofLoadCasesInEnvelope** Number of Load cases in Envelope.(Long)

Return values

1 Successful.

-1 Unsuccessful.

C++ Syntax

```
long RetVal = OSLoadUI::GetLoadEnvelopeDetails(2, nType, nLoads);
```

VBA Syntax

```
Option Explicit
Sub Main
    Dim objOpenStaad As Object
    Dim stdFile As String
    Dim varReturnVal As Long

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

    Dim nType As Long, nLoads As Long
    varReturnVal = objOpenStaad.Load.GetLoadEnvelopeDetails(2, nType, nLoads)
```

```
' process the return value
If (varRetVal = 1) Then
    MsgBox"Type Of Envelope: '" & nType & "'."
    MsgBox"Number of Load cases in Envelope: '" & nLoads & "'."
Else
    MsgBox"unsuccessful"
End If
Set objOpenStaad = Nothing
End Sub
```

See also

[OSLoadUI::CreateLoadEnvelop](#)
[OSLoadUI::AddLoadCasesToEnvelop](#)
[OSLoadUI::RemoveLoadCasesFromEnvelop](#)
[OSLoadUI::GetEnvelopeCount](#)
[OSLoadUI::GetLoadListfromLoadEnvelope](#)

◆ GetLoadListfromLoadEnvelope()

```
long OSLoadUI::GetLoadListfromLoadEnvelope ( long EnvNo,
                                            VARIANT & varLoadCaseList )
```

Gets the list of primary load case reference IDs present in the load envelope passed.

Parameters

[in] **EnvNo** Load Envelope reference ID(Type: Integer/Long).
 [out] **varLoadCaseList** (Primary) load case(s) reference ID(s) VARIANT array(Type: Integer/Long Array). See the function [OSLoadUI::GetLoadEnvelopeDetails](#) to obtain the total number of Load cases present in the load Envelope passed.

Return values

1 Successful.

-1 Unsuccessful.

C++ Syntax

```
long RetVal = OSLoadUI::GetLoadListfromLoadEnvelope(2, varLoadCaseList);
```

VBA Syntax

```
Option Explicit
Sub Main
    Dim objOpenStaad As Object
    Dim stdFile As String
    Dim varRetVal As Long

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

    Dim nType As Long, nLoads As Long
    varRetVal = objOpenStaad.Load.GetLoadEnvelopeDetails(2, nType, nLoads)
    Dim LoadList() As Long
    ReDim LoadList(nLoads - 1)
    varRetVal = objOpenStaad.Load.GetLoadListfromLoadEnvelope(2, LoadList)
    ' process the return value
    If (varRetVal = 1) Then
        MsgBox "1st load in envelope is " & LoadList(0) & "."
    Else
        MsgBox "unsuccessful"
    End If
    Set objOpenStaad = Nothing
End Sub
```

See also

```
OSLoadUI::CreateLoadEnvelop  
OSLoadUI::AddLoadCasesToEnvelop  
OSLoadUI::RemoveLoadCasesFromEnvelop  
OSLoadUI::GetEnvelopeCount  
OSLoadUI::GetLoadEnvelopeDetails
```

◆ RemoveLoadCasesFromEnvelop()

```
VARIANT OSLoadUI::RemoveLoadCasesFromEnvelop ( const VARIANT FAR & varEnvNo,  
                                              const VARIANT FAR & varLoadCaseList )
```

Removes a list of primary load case(s) from an existed load envelop.

Parameters

[in] **varEnvNo** Load Envelop reference ID.
[in] **varLoadCaseList** (Primary) load case(s) reference ID(s) VARIANT array.

Return values

1 OK.
0 General error.

C++ Syntax

```
// Remove a list of primary load case(s) into Load Envelop #2  
VARIANT RetVal = OSLoadUI::RemoveLoadCasesFromEnvelop(2, varLoadCaseList);
```

VBA Syntax

```
' Remove a list of primary load case(s) into Load Envelop #2
Dim RetVal As VARIANT = OSLoadUI.RemoveLoadCasesFromEnvelop(2, varLoadCaseList)
```

See also

OSLoadUI::CreateLoadEnvelop
OSLoadUI::AddLoadCasesToEnvelop
OSLoadUI::DeleteLoadEnvelop

© Copyright Bentley Systems, Inc. For more information, see <http://www.bentley.com>.