Load Definition: Wind

Functions

afx_msg VARIANT	**OSLoadUI::AddWindDefinition** (const VARIANT FAR &varTypeNo, const VARIANT FAR &varTypeName) Adds a Wind Definition named "\b varTypeName" with number ID **varTypeNo**.
afx_msg VARIANT	**OSLoadUI::AddWindIntensity** (const VARIANT FAR &varTypeNo, const VARIANT FAR &varIntensity, const VARIANT FAR &varHeight) Adds to Wind Definitions Wind Intensity by giving Intensity vs. Height.
afx_msg long	**OSLoadUI::ComputeWallWindPressureProfile** (LoadingCode code, double windSpeed, BldgClass bldgclass, BldgType bldgtype, ExposureCategory expCat, BOOL bEscarpment, const VARIANT FAR &varUnitData, const VARIANT FAR &varescarpmentData, const VARIANT FAR &varbldgData, WallType wallType) Generates the wall wind pressure profile using ASCE CODE.
afx_msg long	**OSLoadUI::ComputeWallWindPressureProfileASCE72016** (double windSpeed, double heightAboveSeaLvl, BldgClass bldgclass, BldgType bldgtype, ExposureCategory expCat, BOOL bEscarpment, const VARIANT FAR &varUnitData, const VARIANT FAR &varescarpmentData, const VARIANT FAR &varbldgData, WallType wallType) Generates the wall wind pressure profile using ASCE7-2016 version only. For any versions before 2016, please use the other API ComputeWallWindPressureProfile.
afx_msg VARIANT	**OSLoadUI::AddWindDefinitionASCE7Parameters** (long nTypeNo, ASCE7_Publication code, const VARIANT FAR &windSpeed, const VARIANT FAR &heightAboveSeaLvl, ASCE7_BldgClass bldgclass, ASCE7_BldgType bldgtype, ASCE7_ExposureCategory expCat, long bEscarpment, ASCE7_WallType wallType, long isFlexible, const VARIANT FAR &varescarpmentData, const VARIANT FAR &varbldgData, const VARIANT FAR &varUnitsData, const VARIANT FAR &varFactorsUserInput, const VARIANT FAR &varFactors) Generates the Wind Definition Parameters using ASCE CODE.
afx_msg VARIANT	**OSLoadUI::AddWindExposure** (const VARIANT FAR &varTypeNo, const VARIANT FAR &varExposureFactor, const VARIANT FAR &varNodeArray) Adds Wind Exposures factor to Wind Definitions and assign to node(s).
afx_msg VARIANT	**OSLoadUI::DeleteWindDefinition** (long nTypeNo) Deletes Wind definition. All defintions will be deleted if this input is set as 0.

Detailed Description

These functions are related to Wind Definition.

Function Documentation

◆ AddWindDefinition()

VARIANT OSLoadUI::AddWindDefinition (const VARIANT FAR & varTypeNo,
const VARIANT FAR & varTypeName)

Adds a Wind Definition named "\b varTypeName" with number ID **varTypeNo**.

Parameters

[in] **varTypeNo** Wind Definition Type number ID.

[in] **varTypeName** String name of this new type ("Comment" in STAAD).

Return values

0 OK.

-1 General error.

C++ Syntax

```
// Add a Wind Definition named "WIND 1" and assigned with reference number #1
VARIANT RetVal = OSLoadUI::AddWindDefinition(1, (LPCTSTR)"WIND 1");
```

VBA Syntax

```
' Add a Wind Definition named "WIND 1" and assigned with reference number #1
Dim RetVal As VARIANT = OSLoadUI.AddWindDefinition(1, (LPCTSTR)"WIND 1")
```

◆ AddWindDefinitionASCE7Parameters()

```
VARIANT OSLoadUI::AddWindDefinitionASCE7Parameters ( long                    nTypeNo,
                                                     ASCE7_Publication       code,
                                                     const VARIANT FAR &     windSpeed,
                                                     const VARIANT FAR &     heightAboveSeaLvl,
                                                     ASCE7_BldgClass         bldgclass,
                                                     ASCE7_BldgType          bldgtype,
                                                     ASCE7_ExposureCategory  expCat,
                                                     long                    bEscarpment,
                                                     ASCE7_WallType          wallType,
                                                     long                    isFlexible,
                                                     const VARIANT FAR &     varescarpmentData,
                                                     const VARIANT FAR &     varbldgData,
                                                     const VARIANT FAR &     varUnitsData,
                                                     const VARIANT FAR &     varFactorsUserInput,
                                                     const VARIANT FAR &     varFactors )
```

Generates the Wind Definition Parameters using ASCE CODE.

Parameters

[in] **varTypeNo** Wind Definition Type number ID (Integer).

[in] **code** ASCE CODE:

Value	ASCE CODE
ASCE7_95	ACSE 7-1995
ASCE7_02	ACSE 7-2002
ASCE7_05_10	ACSE 7-2010
ASCE7_16	ACSE 7-2016

[in] **windSpeed** Wind speed (Double).

[in] **heightAboveSeaLvl** Ground Height above sea level (Double) [Required only for ASCE7-2016. For other versions give 0.0]

[in] **bldgclass** Building Classification Category:

Value	Building Classification Category
ASCE7_TypeI = 0	Category I
ASCE7_TypeII = 1	Category II
ASCE7_TypeIII = 2	Category III
ASCE7_TypeIV = 3	Category IV

[in] **bldgtype** Structure Type:

Value	Structure Type

ASCE7_Building = 0	Building Structures
ASCE7_Chimney = 1	Chimney, Tank and similar structures
ASCE7_Solidsign = 2	Solid Signs
ASCE7_Opensign = 3	Open Signs
ASCE7_Laticeframe = 4	Lattice Framework
ASCE7_Trusstower = 5	Trussed Tower

[in] **expCat** Exposure Category.

Value	Exposure Category
ASCE7_ExpA = 0	Exposure A
ASCE7_ExpB = 1	Exposure B
ASCE7_ExpC = 2	Exposure C
ASCE7_ExpD = 3	Exposure D

[in] **varEscarpment** Consider Wind Speed-up over Hills (**FALSE**) or Escarpment (**TRUE**).

[in] **wallType** Building wall to generate Wind Load on:

Value	Wall Type
ASCE7_WindWard = 0	WindWard
ASCE7_LeeWard = 1	Leeward
ASCE7_SideWall = 2	SideWall

(0 to 2 for WindWard, Leeward and SideWall, respectively).

Parameters

[in] **varIsFlexible** Consider structure is Flexible (**TRUE**) or RIGID (**FALSE**).

[in] **varescarpmentData** - Variant Array of Double of size 4 containing information describing Hills or Escarpment:

Index	Data
0	Type: 2D Ridge (0)/ 2D Escarpment (1)/ 3D Escarpment (2)
1	Height (**H**)
2	Distance upwind of crest (**Lh**)
3	Distance from the crest to the building (**x**)

[in] **varBldgData** - Variant Array of Double of size 7 containing information describing the building based on structure type.

Building Data

Index	Item
0	Enclosure Classification:-

	• [**Before 2016**] Open Building (0)/ Partially Enclosed (1)/ Enclosed Building (2) • [**2016**] Open Building (0)/ Partially Open (1)/ Partially Enclosed (2)/ Enclosed Building (3)
1	Building Height
2	Building length long the direction of Wind (**L**)
3	Building length normal to the direction of Wind (**B**)
4	Building Natural Frequency
5	Building Damping Ratio

OR **Tank** Data

Index	Item
0	Horizontal Cross-section Type:- • [**Before 2016**] Square (0)/ Square Diagonal (1)/ Hexagonal or Octagonal (2)/ Round (3) • [**2016**] Square (0)/ Square Diagonal (1)/ Hexagonal (2)/ Octagonal Non-axisymmetric (3) / Octagonal Axisymmetric (4) / Round Non-axisymmetric (5) / Round Axisymmetric (6)
1	Tank Height (**H**)
2	Least Horizontal Dimension (**W**)
3	Depth of producing elements like Spoilers and Ribs (**D'**)
4	Structure Natural Frequency
5	Structure Damping Ratio

OR |**Solid** Sign Data|

Index	Item
0	Solid Sign Height (**H**)
1	Solid Sign M Dimension (**M**)
2	Solid Sign N Dimension (**N**)
3	Structure Natural Frequency
4	Structure Damping Ratio

OR |**Open** Sign/Lattice Framework Data|

Index	Item
0	Orientation Type: Flat (0)/ Rounded (1)
1	Height (**H**)
2	Width
3	Diameter of typical round member

4	Ratio of Solid Area to Gross Area
5	Structure Natural Frequency
6	Structure Damping Ratio

OR |**Trussed** Tower Data|

Index	Item
0	Horizontal Cross Sectio Type: Triangle (0)/ Square (1)
1	Tank Height (**H**)
2	Width
3	Ratio of Solid Area to Gross Area(in percetage)
4	Structure Natural Frequency
5	Structure Damping Ratio

`[in]` **varUnitsData** - VARIANT Integer array of size 7 containing Units of data inputs:

Index	Data
0	Unit of Wind Speed {mph(VelocityUnit::mph or 0) or m/sec(VelocityUnit::metersec or 1) or cm/sec(VelocityUnit::cmsec or 2) or mm/sec(VelocityUnit::mmsec or 3) or kmph(VelocityUnit::kmph or 4) or in/sec(VelocityUnit::inchsec or 5) or ft/sec(VelocityUnit::ftsec or 6) or Yd/sec(VelocityUnit::yardsec or 7)}
1	Unit of Height above sea level {inch(LengthUnit::In or 0) or ft(LengthUnit::Ft or 1) or foot(LengthUnit::foot or 2) or cm(LengthUnit::cm or 3) or m(LengthUnit::M or 4) or mm(LengthUnit::Mm or 5) or dm(LengthUnit::Dm or 6) or Km(LengthUnit::Km or 7) or yard(LengthUnit::Yd or 8) or mile(LengthUnit::mil or 9)}[Required only for ASCE7-2016. For other versions give any length unit.]
2	[Escarpment] Unit of Height (H) {inch(LengthUnit::In or 0) or ft(LengthUnit::Ft or 1) or foot(LengthUnit::foot or 2) or cm(LengthUnit::cm or 3) or m(LengthUnit::M or 4) or mm(LengthUnit::Mm or 5) or dm(LengthUnit::Dm or 6) or Km(LengthUnit::Km or 7) or yard(LengthUnit::Yd or 8) or mile(LengthUnit::mil or 9)}
3	[Escarpment] Unit of Distance upwind of crest (Lh) {inch(LengthUnit::In or 0) or ft(LengthUnit::Ft or 1) or foot(LengthUnit::foot or 2) or cm(LengthUnit::cm or 3) or m(LengthUnit::M or 4) or mm(LengthUnit::Mm or 5) or dm(LengthUnit::Dm or 6) or Km(LengthUnit::Km or 7) or yard(LengthUnit::Yd or 8) or mile(LengthUnit::mil or 9)}

4	[Escarpment] Unit of Distance from the crest to the building (x) {inch(LengthUnit::In or 0) or ft(LengthUnit::Ft or 1) or foot(LengthUnit::foot or 2) or cm(LengthUnit::cm or 3) or m(LengthUnit::M or 4) or mm(LengthUnit::Mm or 5) or dm(LengthUnit::Dm or 6) or Km(LengthUnit::Km or 7) or yard(LengthUnit::Yd or 8) or mile(LengthUnit::mil or 9)}
5	[Building]Unit of Height/ [Tank]Unit of Height/ [Solid Sign]Unit of Height/ [Open Sign/Lattice]Unit of Height/ [Trusses Tower]Unit of Height {inch(LengthUnit::In or 0) or ft(LengthUnit::Ft or 1) or foot(LengthUnit::foot or 2) or cm(LengthUnit::cm or 3) or m(LengthUnit::M or 4) or mm(LengthUnit::Mm or 5) or dm(LengthUnit::Dm or 6) or Km(LengthUnit::Km or 7) or yard(LengthUnit::Yd or 8) or mile(LengthUnit::mil or 9)}
6	[Building]Unit of Length/ [Tank]Unit of Width/ [Solid Sign]Unit of M dimension/ [Open Sign/Lattice]Unit of Width/ [Trusses Tower]Unit of Width {inch(LengthUnit::In or 0) or ft(LengthUnit::Ft or 1) or foot(LengthUnit::foot or 2) or cm(LengthUnit::cm or 3) or m(LengthUnit::M or 4) or mm(LengthUnit::Mm or 5) or dm(LengthUnit::Dm or 6) or Km(LengthUnit::Km or 7) or yard(LengthUnit::Yd or 8) or mile(LengthUnit::mil or 9)}
7	[Building]Unit of Width/ [Tank]Unit of Depth/ [Solid Sign]Unit of N dimension/ [Open Sign/Lattice]Unit of Diameter/ [Trusses Tower]Not Applicable {inch(LengthUnit::In or 0) or ft(LengthUnit::Ft or 1) or foot(LengthUnit::foot or 2) or cm(LengthUnit::cm or 3) or m(LengthUnit::M or 4) or mm(LengthUnit::Mm or 5) or dm(LengthUnit::Dm or 6) or Km(LengthUnit::Km or 7) or yard(LengthUnit::Yd or 8) or mile(LengthUnit::mil or 9)}

`[in]` **varFactorsUserInput-** VARIANT Integer array of size 7 containing information describing whether Factors are User Input or Calculated:

Index	Data
0	Kz is User Input(1) or Calculated(0)
1	Kzt is User Input(1) or Calculated(0)
2	I is User Input(1) or Calculated(0)
3	Kd is User Input(1) or Calculated(0)
4	Ke is User Input(1) or Calculated(0) [Required only for ASCE7-2016]
5	G is User Input(1) or Calculated(0)
6	Cp is User Input(1) or Calculated(0)
7	Gcpi is User Input(1) or Calculated(0)

[in] **varFactorsUserInput-** VARIANT Integer array of size 7 containing information describing whether Factors are User Input or Calculated:

Index	Data
0	Factor Kz
1	Factor Kzt
2	Factor I
3	Factor Kd
4	Factor Ke [Required only for ASCE7-2016]
5	Factor G
6	Factor Cp
7	Factor Gcpi

Return values

0 OK.

-1 General error.

C++ Syntax

```
// Generate wind pressure profile to Wind Definitions #1 using ASCE7-2010, assuming Wind
        speed 85, Category I, Exposure Category B, Building Structure, Consider Wind over
        2D Escarpment, Windward.
long RetVal = COSLoad::AddWindDefinitionASCE7Parameters(1, ASCE7Edition.ASCE7_16, 85.0,
        0.0, ASCE7_BldgClass.ASCE7_TypeI, ASCE7_BldgType.ASCE7_Building,
        ASCE7_ExposureCategory.ASCE7_ExpB, TRUE, ASCE7_WallType.ASCE7_WindWard, FALSE,
        varescarpmentData, varbldgData, varUnitsData, varFactorsUserInput, varFactors );
```

<Consider Wind Speed-Up over Hill or Escarpment. If bEscarpment is TRUE or 1,> dEscarpment data will be counted while calculation>

<If any value is passed except 0,1,2 or 3, then treat it as default which in this case is 1[Bldg_PartEnclosed].>

<If any value is passed except 0,1 or 2, then treat it as default which in this case is 1[Bldg_PartEnclosed].>

<If any value is passed except 0,1,2,3,4 or 5 then treat it as default which in this case is 3[Tank_Round].>

<If any value is passed except 0,1,2 or 3, then treat it as default which in this case is 3[Tank_Round].>

<If any value is passed except 0 or 1, then treat it as default which in this case is 0[OPLF_Flat].>

<If any value is passed except 0 or 1, then treat it as default which in this case is 1[TT_Square].>

◆ AddWindExposure()

```
VARIANT OSLoadUI::AddWindExposure ( const VARIANT FAR & varTypeNo,
                                    const VARIANT FAR & varExposureFactor,
                                    const VARIANT FAR & varNodeArray )
```

Adds Wind Exposures factor to Wind Definitions and assign to node(s).

Parameters

varTypeNo	Wind Definition Type number ID.
varExposureFactor	Exposure factor.
varNodeArray	Node number ID(s) VARIANT array. For additional information, please refer to Section 5.31.3 of the Technical Reference manual.

Return values

0 OK.

-1 General error.

C++ Syntax

```
// Add Wind Exposure factor 1.0 to Wind Definitions #1 and assign to node(s)
VARIANT RetVal = OSLoadUI::AddWindExposure(1, 1.0, varNodeNoArray);
```

VBA Syntax

```
' Add Wind Exposure factor 1.0 to Wind Definitions #1 and assign to node(s)
Dim RetVal As VARIANT = OSLoadUI.AddWindExposure(1, 1.0, varNodeNoArray)
```

◆ AddWindIntensity()

VARIANT OSLoadUI::AddWindIntensity (const VARIANT FAR & varTypeNo,
const VARIANT FAR & varIntensity,
const VARIANT FAR & varHeight)

Adds to Wind Definitions Wind Intensity by giving Intensity vs. Height.

Parameters

[in] **varTypeNo** Wind Definition Type number ID.

[in] **varIntensity** Intensity value VARIANT (Double) array (Unit: Force/Length^2).

[in] **varHeight** Height value VARIANT (Double) array (Unit: Length). For additional information, please refer to Section 5.31.3 of the Technical Reference manual.

Return values

0 OK.

-1 General error.

C++ Syntax

```
// Add Wind Intensity to Wind Definitions #1.
VARIANT RetVal = OSLoadUI::AddWindIntensity(1, Intensity, Height);
```

VBA Syntax

```
'1) Add Wind Intensity to Wind Definitions #1. For creating multiple set of height vs
       intensity data.
Dim typ As Integer
Dim intensity(0 To 1) As Double
Dim height(0 To 1) As Double
typ = 1
intensity(0) = 1.24
intensity(1) = 1.41
height(0) = 10.00
height(1) = 15.00
Dim RetVal As Variant
RetVal = objOpenStaad.Load.AddWindIntensity(typ, intensity, height)

'2) Add Wind Intensity to Wind Definitions #1. For creating a single pair of height vs
       intensity data.
Dim typ As Integer
Dim intensity As Double
Dim height As Double
Dim RetVal As Variant
typ = 1
intensity = 1.41
height = 10.00
ino2 = objOpenStaad.Load.AddWindIntensity(typ, intensity, height)
```

◆ ComputeWallWindPressureProfile()

```
long OSLoadUI::ComputeWallWindPressureProfile ( LoadingCode          code,
                                                double               windSpeed,
                                                BldgClass            bldgclass,
                                                BldgType             bldgtype,
                                                ExposureCategory     expCat,
                                                BOOL                 bEscarpment,
                                                const VARIANT FAR &  varUnitData,
                                                const VARIANT FAR &  varescarpmentData,
                                                const VARIANT FAR &  varbldgData,
                                                WallType             wallType )
```

Generates the wall wind pressure profile using ASCE CODE.

Parameters

[in] **code** ASCE CODE:

Value	ASCE CODE
ASCE7Y95 = 0	ACSE 7-1995
ACSE702 = 1	ACSE 7-2002
ACSE705_10 = 2	ACSE 7-2010

[in] **windSpeed** Wind speed.

[in] **bldgclass** Building Classification Category:

Value	Building Classification Category
TypeI = 0	Category I
TypeII = 1	Category II
TypeIII = 2	Category III
TypeIV = 3	Category IV

[in] **bldgtype** Structure Type:

Value	Structure Type
Building = 0	Building Structures
Chimney = 1	Chimney, Tank and similar structures
Solidsign = 2	Solid Signs
Opensign = 3	Open Signs
Laticeframe = 4	Lattice Framework
Trusstower = 5	Trussed Tower

[in] **expCat** Exposure Category.

Value	Exposure Category

ExpA = 0	Exposure A
ExpB = 1	Exposure B
ExpC = 2	Exposure C
ExpD = 3	Exposure D

`[in]` **bEscarpment** Consider Wind Speed-up over Hills (**FALSE**) or Escarpment (**TRUE**).

`[in]` **varescarpmentData** Information describing Hills or Escarpment:

Index	Data
0	Type: 2D Ridge (0), 2D Escarpment (1), 3D Escarpment (2)
1	Height (**H**)
2	Distance upwind of crest (**Lh**)
3	Distance from the crest to the building (**x**)

`[in]` **varbldgData** - Variant Array of Double of size 7 containing information describing the building based on structure type.

Building Data

Index	Item
0	Enclosure Classification: Open Building (0)/ Partially Enclosed (1)/ Enclosed Building (2)
1	Building Height
2	Building length long the direction of Wind (**L**)
3	Building length normal to the direction of Wind (**B**)
4	Building Natural Frequency
5	Building Damping Ratio

OR **Tank** Data

Index	Item
0	Horizontal Cross-section Type:- Square (0)/ Square Diagonal (1)/ Hexagonal or Octagonal (2)/ Round (3)
1	Tank Height (**H**)
2	Least Horizontal Dimension (**W**)
3	Depth of producing elements like Spoilers and Ribs (**D'**)
4	Structure Natural Frequency
5	Structure Damping Ratio

OR |**Solid** Sign Data|

Index	Item
0	Solid Sign Height (**H**)
1	Solid Sign M Dimension (**M**)

2	Solid Sign N Dimension (**N**)
3	Structure Natural Frequency
4	Structure Damping Ratio

OR |**Open** Sign/Lattice Framework Data|

Index	Item
0	Orientation Type: Flat (0)/ Rounded (1)
1	Height (**H**)
2	Width
3	Diameter of typical round member
4	Structure Natural Frequency
5	Structure Damping Ratio
6	Ratio of Solid Area to Gross Area

OR |**Trussed** Tower Data|

Index	Item
0	Horizontal Cross Sectio Type: Triangle (0)/ Square (1)
1	Height (**H**)
2	Width
3	Structure Natural Frequency
4	Structure Damping Ratio
5	Ratio of Solid Area to Gross Area(in percetage)

`[in]` **wallType** Building wall to generate Wind Load on:

Value	Wall Type
WindWard = 0	WindWard
LeeWard = 1	Leeward
SideWall = 2	SideWall

(0 to 2 for WindWard, Leeward and SideWall, respectively).

Return values

number of Height or Intensity data.

-1 General error.

C++ Syntax

```
// Generate wall wind pressure profile using ASCE7-2010, assuming Wind speed 85, Category
       I, Exposure Category B, Building Structure, Consider Wind over 2D Escarpment,
       Windward.
```

```
long RetVal = OSLoadUI::ComputeWallWindPressureProfile(ACSE705_10, 85.0, TypeI, Building,
        EXPB, TRUE, varescarpmentData, varbldgData, WindWard);
```

See also

OSLoadUI::GetWallWindPressureProfile

OSLoadUI::GetDesignWindPressureForLowRise

OSLoadUI::GetDesignComponentWindPressureForLowRise

◆ ComputeWallWindPressureProfileASCE72016()

```
long OSLoadUI::ComputeWallWindPressureProfileASCE72016 ( double              windSpeed,
                                                         double              heightAboveSeaLvl,
                                                         BldgClass           bldgclass,
                                                         BldgType            bldgtype,
                                                         ExposureCategory    expCat,
                                                         BOOL                bEscarpment,
                                                         const VARIANT FAR & varUnitData,
                                                         const VARIANT FAR & varescarpmentData,
                                                         const VARIANT FAR & varbldgData,
                                                         WallType            wallType )
```

Generates the wall wind pressure profile using ASCE7-2016 version only. For any versions before 2016, please use the other API ComputeWallWindPressureProfile.

Parameters

`[in]` **windSpeed** Wind speed. Default value 85 mph.

`[in]` **heightAboveSeaLvl** Ground height above sea level. Used only for ASCE7-2016 Wind. Default value 0.0 ft.

`[in]` **bldgclass** Building Classification Category:

Value	Building Classification Category
TypeI = 0	Category I
TypeII = 1	Category II
TypeIII = 2	Category III
TypeIV = 3	Category IV

`[in]` **bldgtype** Structure Type:

Value	Structure Type
Building = 0	Building Structures
Chimney = 1	Chimney, Tank and similar structures
Solidsign = 2	Solid Signs
Opensign = 3	Open Signs
Laticeframe = 4	Lattice Framework
Trusstower = 5	Trussed Tower

`[in]` **expCat** Exposure Category.

Value	Exposure Category
ExpA = 0	Exposure A
ExpB = 1	Exposure B

ExpC = 2	Exposure C
ExpD = 3	Exposure D

[in] **bEscarpment** Consider Wind Speed-up over Hills (**FALSE**) or Escarpment (**TRUE**).

[in] **varUnitsData** - VARIANT Integer array of size 8 containing Units of data inputs:

Index	Data
0	Unit of Wind Speed {mph(VelocityUnit::mph or 0) or m/sec(VelocityUnit::metersec or 1) or cm/sec(VelocityUnit::cmsec or 2) or mm/sec(VelocityUnit::mmsec or 3) or kmph(VelocityUnit::kmph or 4) or in/sec(VelocityUnit::inchsec or 5) or ft/sec(VelocityUnit::ftsec or 6) or Yd/sec(VelocityUnit::yardsec or 7)}
0	Unit of Ground height above sea level {inch(LengthUnit::In or 0) or ft(LengthUnit::Ft or 1) or foot(LengthUnit::foot or 2) or cm(LengthUnit::cm or 3) or m(LengthUnit::M or 4) or mm(LengthUnit::Mm or 5) or dm(LengthUnit::Dm or 6) or Km(LengthUnit::Km or 7) or yard(LengthUnit::Yd or 8) or mile(LengthUnit::mil or 9)}
1	[Escarpment] Unit of Height (H) {inch(LengthUnit::In or 0) or ft(LengthUnit::Ft or 1) or foot(LengthUnit::foot or 2) or cm(LengthUnit::cm or 3) or m(LengthUnit::M or 4) or mm(LengthUnit::Mm or 5) or dm(LengthUnit::Dm or 6) or Km(LengthUnit::Km or 7) or yard(LengthUnit::Yd or 8) or mile(LengthUnit::mil or 9)}
2	[Escarpment] Unit of Distance upwind of crest (Lh) {inch(LengthUnit::In or 0) or ft(LengthUnit::Ft or 1) or foot(LengthUnit::foot or 2) or cm(LengthUnit::cm or 3) or m(LengthUnit::M or 4) or mm(LengthUnit::Mm or 5) or dm(LengthUnit::Dm or 6) or Km(LengthUnit::Km or 7) or yard(LengthUnit::Yd or 8) or mile(LengthUnit::mil or 9)}
3	[Escarpment] Unit of Distance from the crest to the building (x) {inch(LengthUnit::In or 0) or ft(LengthUnit::Ft or 1) or foot(LengthUnit::foot or 2) or cm(LengthUnit::cm or 3) or m(LengthUnit::M or 4) or mm(LengthUnit::Mm or 5) or dm(LengthUnit::Dm or 6) or Km(LengthUnit::Km or 7) or yard(LengthUnit::Yd or 8) or mile(LengthUnit::mil or 9)}
4	[Building]Unit of Height/ [Tank]Unit of Height/ [Solid Sign]Unit of Height/ [Open Sign/Lattice]Unit of Height/ [Trusses Tower]Unit of Height {inch(LengthUnit::In or 0) or ft(LengthUnit::Ft or 1) or foot(LengthUnit::foot or 2) or cm(LengthUnit::cm or 3) or m(LengthUnit::M or 4) or mm(LengthUnit::Mm or 5) or

	dm(LengthUnit::Dm or 6) or Km(LengthUnit::Km or 7) or yard(LengthUnit::Yd or 8) or mile(LengthUnit::mil or 9)}
5	[Building]Unit of Length/ [Tank]Unit of Width/ [Solid Sign]Unit of M dimension/ [Open Sign/Lattice]Unit of Width/ [Trusses Tower]Unit of Width {inch(LengthUnit::In or 0) or ft(LengthUnit::Ft or 1) or foot(LengthUnit::foot or 2) or cm(LengthUnit::cm or 3) or m(LengthUnit::M or 4) or mm(LengthUnit::Mm or 5) or dm(LengthUnit::Dm or 6) or Km(LengthUnit::Km or 7) or yard(LengthUnit::Yd or 8) or mile(LengthUnit::mil or 9)}
6	[Building]Unit of Width/ [Tank]Unit of Depth/ [Solid Sign]Unit of N dimension/ [Open Sign/Lattice]Unit of Diameter/ [Trusses Tower]Not Applicable {inch(LengthUnit::In or 0) or ft(LengthUnit::Ft or 1) or foot(LengthUnit::foot or 2) or cm(LengthUnit::cm or 3) or m(LengthUnit::M or 4) or mm(LengthUnit::Mm or 5) or dm(LengthUnit::Dm or 6) or Km(LengthUnit::Km or 7) or yard(LengthUnit::Yd or 8) or mile(LengthUnit::mil or 9)}

`[in]` **varescarpmentData** Information describing Hills or Escarpment:

Index	Data
0	Type: 2D Ridge (0), 2D Escarpment (1), 3D Escarpment (2)
1	Height (**H**)
2	Distance upwind of crest (**Lh**)
3	Distance from the crest to the building (**x**)

`[in]` **varbldgData** - Variant Array of Double of size 7 containing information describing the building based on structure type.

Building Data

Index	Item
0	Enclosure Classification: Open Building (0)/ Partially Enclosed (1)/ Enclosed Building (2)
1	Building Height
2	Building length long the direction of Wind (**L**)
3	Building length normal to the direction of Wind (**B**)
4	Building Natural Frequency
5	Building Damping Ratio

OR **Tank** Data

Index	Item
0	Horizontal Cross-section Type:- Square (0)/ Square Diagonal (1)/ Hexagonal or Octagonal (2)/ Round (3)
1	Tank Height (**H**)

2	Least Horizontal Dimension (**W**)
3	Depth of producing elements like Spoilers and Ribs (**D'**)
4	Structure Natural Frequency
5	Structure Damping Ratio

OR |**Solid** Sign Data|

Index	Item
0	Solid Sign Height (**H**)
1	Solid Sign M Dimension (**M**)
2	Solid Sign N Dimension (**N**)
3	Structure Natural Frequency
4	Structure Damping Ratio

OR |**Open** Sign/Lattice Framework Data|

Index	Item
0	Orientation Type: Flat (0)/ Rounded (1)
1	Height (**H**)
2	Width
3	Diameter of typical round member
4	Structure Natural Frequency
5	Structure Damping Ratio
6	Ratio of Solid Area to Gross Area

OR |**Trussed** Tower Data|

Index	Item
0	Horizontal Cross Sectio Type: Triangle (0)/ Square (1)
1	Height (**H**)
2	Width
3	Structure Natural Frequency
4	Structure Damping Ratio
5	Ratio of Solid Area to Gross Area(in percetage)

`[in]` **wallType** Building wall to generate Wind Load on:

Value	Wall Type
WindWard = 0	WindWard
LeeWard = 1	Leeward
SideWall = 2	SideWall

(0 to 2 for WindWard, Leeward and SideWall, respectively).

Return values

number of Height or Intensity data.

-1 General error.

C++ Syntax

```
// Generate wall wind pressure profile using ASCE7-2016, assuming Wind speed 85, Ground
       height above sea level 0.0, Category I, Exposure Category B, Building Structure,
       Consider Wind over 2D Escarpment, Windward.
long RetVal = COSLoad::ComputeWallWindPressureProfileASCE72016(85.0, 0.0, TypeI,
       Building, ExpB, TRUE, varescarpmentData, varbldgData, WindWard);
```

See also

OSLoadUI::GetWallWindPressureProfile

OSLoadUI::GetDesignWindPressureForLowRise

OSLoadUI::GetDesignComponentWindPressureForLowRise

◆ DeleteWindDefinition()

VARIANT OSLoadUI::DeleteWindDefinition (long nTypeNo)

Deletes Wind definition. All defintions will be deleted if this input is set as 0.

Parameters

[in] **nTypeNo** Type of Wind.

Return values

0 OK.

-8039 Invalid load definition.

C++ Syntax

```
// Delete Wind Definition of selected WindType.
long RetVal = COSLoad::DeleteWindDefinition(nTypeNo);
```

.