

# Material: Create Material Information

## Property

## Functions

afx_msg VARIANT	<b>OSPropertyUI::CreateIsotropicMaterialProperties</b> (const VARIANT &strName, const VARIANT &varE, const VARIANT &varPoisson, const VARIANT &varG, const VARIANT &varDensity, const VARIANT &varAlpha, const VARIANT &varCrDamp) Creates isotropic material properties.
afx_msg VARIANT	<b>OSPropertyUI::CreateIsotropicMaterialPropertiesEx</b> (const VARIANT &strName, const VARIANT &varE, const VARIANT &varPoisson, const VARIANT &varG, const VARIANT &varDensity, const VARIANT &varAlpha, const VARIANT &varCrDamp, const VARIANT &varFy, const VARIANT &varFu, const VARIANT &varRy, const VARIANT &varRt, const VARIANT &varFcu) Creates isotropic material property extended.
afx_msg VARIANT	<b>OSPropertyUI::CreateIsotropicMaterialSteel</b> (LPCSTR strName, const VARIANT &dE, const VARIANT &dPoisson, const VARIANT &dG, const VARIANT &dDensity, const VARIANT &dAlpha, const VARIANT &dCrDamp, const VARIANT &dfu, const VARIANT &dfy, const VARIANT &drt, const VARIANT &dry, const VARIANT &bPhysical) Creates isotropic material steel.
afx_msg VARIANT	<b>OSPropertyUI::CreateIsotropicMaterialConcrete</b> (LPCSTR strName, const VARIANT &dE, const VARIANT &dPoisson, const VARIANT &dG, const VARIANT &dDensity, const VARIANT &dAlpha, const VARIANT &dCrDamp, const VARIANT &dfcu, const VARIANT &bPhysical) Creates isotropic material concrete.
afx_msg VARIANT	<b>OSPropertyUI::CreateIsotropicMaterialAluminum</b> (LPCSTR strName, const VARIANT &varE, const VARIANT &varPoisson, const VARIANT &varG, const VARIANT &varDensity, const VARIANT &varAlpha, const VARIANT &varCrDamp, const VARIANT &varPhysical) Creates isotropic material aluminum.
afx_msg VARIANT	<b>OSPropertyUI::CreateIsotropicMaterialTimber</b> (LPCSTR strName, const VARIANT &varE, const VARIANT &varPoisson, const VARIANT &varG, const VARIANT &varDensity, const VARIANT &varAlpha, const VARIANT &varCrDamp, const VARIANT &varPhysical) Creates isotropic material timber.

## Detailed Description

These functions are related to create material information.

## Function Documentation

## ◆ CreateIsotropicMaterialAluminum()

```
VARIANT OSPropertyUI::CreateIsotropicMaterialAluminum ( LPCSTR      strName,
                                                         const VARIANT & varE,
                                                         const VARIANT & varPoisson,
                                                         const VARIANT & varG,
                                                         const VARIANT & varDensity,
                                                         const VARIANT & varAlpha,
                                                         const VARIANT & varCrDamp,
                                                         const VARIANT & varPhysical )
```

Creates isotropic material aluminum.

### Parameters

- [in] **strName** Identification title of material.
- [in] **dE** Modulus of elasticity (**E**).
- [in] **dPoisson** Poisson's ratio (**POI**).
- [in] **dG** Shear modulus (**G**).
- [in] **dDensity** Weight density (**DEN**).
- [in] **dAlpha** Coefficient of thermal expansion (**ALP**).
- [in] **dCrDamp** Damping ratio (**DAMP**).
- [in] **bPhysical** Indetifies if the material is for physical member

### Return values

- 1** Material is updated as a material with that name was already present.
- 0** Material is created.
- 1** General Error

### C++ Syntax

```
// Create material aluminum.
string strMaterialName = "AluminumM124";
double dE = 1.44e+06, dPoisson = 0.33, dG = 5413.5, dDensity = 0.17, dAlpha = 1.3e-05,
       dCrDamp = 0.03;
int bPhysical = 0;
long RetVal = m_OStd.Property.CreateIsotropicMaterialAluminum(strMaterialName, dE,
                                                              dPoisson, dG, dDensity, dAlpha, dCrDamp, bPhysical);
```

### C# Syntax

```
// Create material aluminum for physical member
double dE = 1.44e+06, dPoisson = 0.33, dG = 5413.5, dDensity = 0.17, dAlpha = 1.3e-05,
       dCrDamp = 0.03;
int bPhysical = 1;
long RetVal = OSPropertyUI::CreateIsotropicMaterialAluminum((LPCSTR)"AluminumM124", dE,
                                                              dPoisson, dG, dDensity, dAlpha, dCrDamp, bPhysical);
```

## VB Syntax

```
// Create material aluminum
Dim RetVal As Variant
Dim strMaterialName As String
Dim dE As Double
Dim dPoisson As Double
Dim dG As Double
Dim dDensity As Double
Dim dAlpha As Double
Dim dCrDamp As Double
Dim bPhysical As Integer
strMaterialName = "AluminumM124"
dE = 1.44e+06
dPoisson = 0.33
dG = 5413.5
dDensity = 0.5
dAlpha = 1.3e-05
dCrDamp = 0.5
bPhysical = 0
RetVal = objOpenStaad.Property.CreateIsotropicMaterialAluminum(strMaterialName, dE,
    dPoisson, dG, dDensity, dAlpha, dCrDamp, bPhysical)
```

### See also

[OSPropertyUI::CreateIsotropicMaterialSteel](#)

[OSPropertyUI::CreateIsotropicMaterialConcrete](#)

[OSPropertyUI::CreateIsotropicMaterialTimber](#)

[OSPropertyUI::CreateIsotropicMaterialProperties](#)

## ◆ CreateIsotropicMaterialConcrete()

```
VARIANT OSPropertyUI::CreateIsotropicMaterialConcrete ( LPCSTR      strName,
                                                         const VARIANT & dE,
                                                         const VARIANT & dPoisson,
                                                         const VARIANT & dG,
                                                         const VARIANT & dDensity,
                                                         const VARIANT & dAlpha,
                                                         const VARIANT & dCrDamp,
                                                         const VARIANT & dfcu,
                                                         const VARIANT & bPhysical )
```

Creates isotropic material concrete.

### Parameters

- [in] **strName** Identification title of material.
- [in] **dE** Modulus of elasticity (**E**).
- [in] **dPoisson** Poisson's ratio (**POI**).
- [in] **dG** Shear modulus (**G**).
- [in] **dDensity** Weight density (**DEN**).
- [in] **dAlpha** Coefficient of thermal expansion (**ALP**).
- [in] **dCrDamp** Damping ratio (**DAMP**).
- [in] **dfcu** Compressive strength (**Fcu**).
- [in] **bPhysical** Indetifies if the material is for physical member

### Return values

- 1** Material is updated as a material with that name was already present.
- 0** Material is created.
- 1** General Error

### C++ Syntax

```
// Create material concrete.
string strMaterialName = "CONCRETE1";
double dE = 453600, dPoisson = 0.17, dG = 193846, dDensity = 0.15, dAlpha = 5e-06,
        dCrDamp = 0.03, dFcu= 576;
int bPhysical = 0;.
long RetVal = OSPropertyUI::CreateIsotropicMaterialConcrete(strMaterialName, dE,
        dPoisson, dG, dDensity, dAlpha, dCrDamp, dFcu, bPhysical);
```

### C# Syntax

```
// Create material concrete for physical member
string strMaterialName = "CONCRETE1";
double dE = 453600, dPoisson = 0.17, dG = 193846, dDensity = 0.15, dAlpha = 5e-06,
        dCrDamp = 0.03, dFcu= 576;
```

```
int bPhysical = 1;.
long RetVal = OSPropertyUI::CreateIsotropicMaterialConcrete(strMaterialName, dE,
    dPoisson, dG, dDensity, dAlpha, dCrDamp, dFcu, bPhysical);
```

## VB Syntax

```
// Create material concrete
Dim RetVal As Variant
Dim strMaterialName As String
Dim dE As Double
Dim dPoisson As Double
Dim dG As Double
Dim dDensity As Double
Dim dAlpha As Double
Dim dCrDamp As Double
Dim bPhysical As Integer
strMaterialName = "AluminumM124"
dE = 453600
dPoisson = 0.17
dG = 193846
dDensity = 0.15
dAlpha = 5e-05
dCrDamp = 0.03
dFcu= 576
bPhysical = 0
RetVal = objOpenStaad.Property.CreateIsotropicMaterialConcrete(strMaterialName, dE,
    dPoisson, dG, dDensity, dAlpha, dCrDamp, dFcu, bPhysical)
```

## See also

[OSPropertyUI::CreateIsotropicMaterialSteel](#)

[OSPropertyUI::CreateIsotropicMaterialAluminum](#)

[OSPropertyUI::CreateIsotropicMaterialTimber](#)

[OSPropertyUI::CreateIsotropicMaterialProperties](#)

## ◆ CreateIsotropicMaterialProperties()

```
VARIANT OSPropertyUI::CreateIsotropicMaterialProperties ( const VARIANT & strName,
                                                         const VARIANT & varE,
                                                         const VARIANT & varPoisson,
                                                         const VARIANT & varG,
                                                         const VARIANT & varDensity,
                                                         const VARIANT & varAlpha,
                                                         const VARIANT & varCrDamp )
```

Creates isotropic material properties.

### Parameters

- [in] **strName** Material Name.
- [out] **varE** Modulus of elasticity (**E**) VARIANT array (of size 3).
- [out] **varPoisson** Poisson's ratio (**POI**) VARIANT array (of size 3).
- [out] **varG** Shear modulus (**G**) VARIANT array (of size 3).
- [out] **varDensity** Weight density (**DEN**) VARIANT array (of size 3).
- [out] **varAlpha** Coefficient of thermal expansion (**ALP**) VARIANT array (of size 3).
- [out] **varCrDamp** Damping ratio (**DAMP**) VARIANT array (of size 3).

### Return values

- 1** Material is updated as a material with that name was already present.
- 0** Material is created.
- 1** General Error

### C++ Syntax

```
// Create material timber.
string strMaterialName = "MATERIAL1";
double dE = 262040, dPoisson = 0.2, dG = 0.06, dDensity = 0.04, dAlpha = 2.78e-6, dCrDamp
    = 0.06;
int bPhysical = 0;
VARIANT var = OSPropertyUI::CreateIsotropicMaterialProperties(strName, dE, dPoisson, dG,
    dDensity, dAlpha, dCrDamp);
```

### C++ Syntax

```
// Create material timber for physical member
double dE = 262040, dPoisson = 0.2, dG = 0.06, dDensity = 0.04, dAlpha = 2.78e-6, dCrDamp
    = 0.06;
int bPhysical = 1;
long RetVal = COSProperty::CreateIsotropicMaterialProperties((LPCSTR)"MATERIAL1", &dE,
    &dPoisson, &dG, &dDensity, &dAlpha, &dCrDamp);
```

### VB Syntax

```
// Create material timber
```

```
Dim RetVal As Variant
Dim strMaterialName As String
Dim dE As Double
Dim dPoisson As Double
Dim dG As Double
Dim dDensity As Double
Dim dAlpha As Double
Dim dCrDamp As Double
Dim bPhysical As Integer
strMaterialName = "AluminumM124"
dE = 262040
dPoisson = 0.2
dG = 0.06
dDensity = 0.04
dAlpha = 2.78e-06
dCrDamp = 0.06
RetVal = objOpenStaad.Property.CreateIsotropicMaterialProperties(strMaterialName, dE,
    dPoisson, dG, dDensity, dAlpha, dCrDamp)
```

**See also**[OSPropertyUI::CreateIsotropicMaterialSteel](#)[OSPropertyUI::CreateIsotropicMaterialConcrete](#)[OSPropertyUI::CreateIsotropicMaterialTimber](#)[OSPropertyUI::CreateIsotropicMaterialAluminum](#)[OSPropertyUI::CreateIsotropicMaterialPropertiesEx](#)**◆ CreateIsotropicMaterialPropertiesEx()**



```
VARIANT OSPropertyUI::CreateIsotropicMaterialPropertiesEx ( const VARIANT & strName,
                                                            const VARIANT & varE,
                                                            const VARIANT & varPoisson,
                                                            const VARIANT & varG,
                                                            const VARIANT & varDensity,
                                                            const VARIANT & varAlpha,
                                                            const VARIANT & varCrDamp,
                                                            const VARIANT & varFy,
                                                            const VARIANT & varFu,
                                                            const VARIANT & varRy,
                                                            const VARIANT & varRt,
                                                            const VARIANT & varFcu )
```

Creates isotropic material property extended.

### Parameters

- [in] **strName** Identification title of material of material..
- [in] **dE** Modulus of elasticity (**E**).
- [in] **dPoisson** Poisson's ratio (**POI**).
- [in] **dG** Shear modulus (**G**).
- [in] **dDensity** Weight density (**DEN**).
- [in] **dAlpha** Coefficient of thermal expansion (**ALP**).
- [in] **dCrDamp** Damping ratio (**DAMP**).
- [in] **dFy** Yield stress (**Fy**).
- [in] **dFu** Tensile strength (**Fu**).
- [in] **dRy** Yield strength ratio (**Ry**).
- [in] **dRt** Tensile strength ratio (**Rt**).
- [in] **dFcu** Compressive strength (**Fcu**).

### Return values

- 1** Material is updated as a material with that name was already present.
- 0** Material is created.
- 1** General Error

### C++ Syntax

```
// Create material steel.
string strMaterialName = "STEEL1";
double dE = 4176000, dPoisson = 0.3, dG = 1606150, dDensity = 0.49, dAlpha = 6e-05,
       dCrDamp = 0.03, dFy = 8352, dFu = 5184, dRt = 1.2, dRy = 1.5, dFcu = 1000;
int bPhysical = 0;
```

```
long RetVal = OSPropertyUI::CreateIsotropicMaterialPropertiesEx(strMaterialName, dE,
    dPoisson, dG, dDensity, dAlpha, dCrDamp, dFy, dFu, dRt, dRy, dFcu);
```

@par C# Syntax

@code {.cs}

```
// Create material steel for physical member
double dE = 4176000, dPoisson = 0.3, dG = 1606150, dDensity = 0.49, dAlpha = 6e-05,
    dCrDamp = 0.03, dFy = 8352, dFu = 5184, dRt = 1.2, dRy = 1.5, dFcu = 1000;
int bPhysical = 1;
long RetVal = OSPropertyUI::CreateIsotropicMaterialPropertiesEx(strMaterialName, dE,
    dPoisson, dG, dDensity, dAlpha, dCrDamp, dFy, dFu, dRt, dRy, dFcu);
```

## VB Syntax

```
// Create material steel
Dim RetVal As Variant
Dim strMaterialName As String
Dim dE As Double
Dim dPoisson As Double
Dim dG As Double
Dim dDensity As Double
Dim dAlpha As Double
Dim dCrDamp As Double
Dim dFy As Double
Dim dFu As Double
Dim dRt As Double
Dim dRy As Double
Dim Fcu As Double
strMaterialName = "STEEL1"
dE = 4176000
dPoisson = 0.3
dG = 1606150
dDensity = 0.49
dAlpha = 6e-05
dCrDamp = 0.03
dFy = 8352
dFu = 5184
dRt = 1.2
dRy = 1.5
Fcu = 1000
RetVal = objOpenStaad.Property.CreateIsotropicMaterialPropertiesEx(strMaterialName, dE,
    dPoisson, dG, dDensity, dAlpha, dCrDamp, dFy, dFu, dRt, dRy, dFcu)
```

## See also

[OSPropertyUI::CreateIsotropicMaterialSteel](#)

[OSPropertyUI::CreateIsotropicMaterialConcrete](#)

[OSPropertyUI::CreateIsotropicMaterialTimber](#)

[OSPropertyUI::CreateIsotropicMaterialProperties](#)

[OSPropertyUI::CreateIsotropicMaterialAluminum](#)

## ◆ CreateIsotropicMaterialSteel()

```
VARIANT OSPropertyUI::CreateIsotropicMaterialSteel ( LPCSTR      strName,
                                                    const VARIANT & dE,
                                                    const VARIANT & dPoisson,
                                                    const VARIANT & dG,
                                                    const VARIANT & dDensity,
                                                    const VARIANT & dAlpha,
                                                    const VARIANT & dCrDamp,
                                                    const VARIANT & dfu,
                                                    const VARIANT & dfy,
                                                    const VARIANT & drt,
                                                    const VARIANT & dry,
                                                    const VARIANT & bPhysical )
```

Creates isotropic material steel.

### Parameters

- [in] **strName** Identification title of material.
- [in] **dE** Modulus of elasticity (**E**).
- [in] **dPoisson** Poisson's ratio (**POI**).
- [in] **dG** Shear modulus (**G**).
- [in] **dDensity** Weight density (**DEN**).
- [in] **dAlpha** Coefficient of thermal expansion (**ALP**).
- [in] **dCrDamp** Damping ratio (**DAMP**).
- [in] **dfu** Tensile strength (**Fu**).
- [in] **dfy** Yield stress (**Fy**).
- [in] **drt** Tensile strength ratio (**Rt**).
- [in] **dry** Yield strength ratio (**Ry**).
- [in] **bPhysical** Identifies if the material is for physical member

### Return values

- 1** Material is updated as a material with that name was already present.
- 0** Material is created.
- 1** General Error

### C++ Syntax

```
// Create material steel.
string strMaterialName = "STEEL1";
double dE = 4176000, dPoisson = 0.3, dG = 1606150, dDensity = 0.49, dAlpha = 6e-05,
       dCrDamp = 0.03, dFy = 8352, dFu = 5184, dRt = 1.2, dRy = 1.5;
int bPhysical = 0;
```

```
long RetVal = OSPropertyUI::CreateIsotropicMaterialSteel(strMaterialName, dE, dPoisson,
    dG, dDensity, dAlpha, dCrDamp, dFy, dFu, dRt, dRy, bPhysical);
```

## C# Syntax

```
// Create material steel for physical member
double dE = 4176000, dPoisson = 0.3, dG = 1606150, dDensity = 0.49, dAlpha = 6e-05,
    dCrDamp = 0.03, dFy = 8352, dFu = 5184, dRt = 1.2, dRy = 1.5;
int bPhysical = 1;
long RetVal = OSPropertyUI::CreateIsotropicMaterialSteel(strMaterialName, dE, dPoisson,
    dG, dDensity, dAlpha, dCrDamp, dFy, dFu, dRt, dRy, bPhysical);
```

## VB Syntax

```
// Create material steel
Dim RetVal As Variant
Dim strMaterialName As String
Dim dE As Double
Dim dPoisson As Double
Dim dG As Double
Dim dDensity As Double
Dim dAlpha As Double
Dim dCrDamp As Double
Dim dFy As Double
Dim dFu As Double
Dim dRt As Double
Dim dRy As Double
Dim bPhysical As Integer
strMaterialName = "STEEL1"
dE = 4176000
dPoisson = 0.3
dG = 1606150
dDensity = 0.49
dAlpha = 6e-05
dCrDamp = 0.03
dFy = 8352
dFu = 5184
dRt = 1.2
dRy = 1.5
bPhysical = 0
RetVal = objOpenStaad.Property.CreateIsotropicMaterialSteel(strMaterialName, dE,
    dPoisson, dG, dDensity, dAlpha, dCrDamp, dFy, dFu, dRt, dRy, bPhysical)
```

## See also

[OSPropertyUI::CreateIsotropicMaterialConcrete](#)

[OSPropertyUI::CreateIsotropicMaterialAluminum](#)

[OSPropertyUI::CreateIsotropicMaterialTimber](#)

[OSPropertyUI::CreateIsotropicMaterialProperties](#)

## ◆ CreateIsotropicMaterialTimber()

```
VARIANT OSPropertyUI::CreateIsotropicMaterialTimber ( LPCSTR      strName,
                                                    const VARIANT & varE,
                                                    const VARIANT & varPoisson,
                                                    const VARIANT & varG,
                                                    const VARIANT & varDensity,
                                                    const VARIANT & varAlpha,
                                                    const VARIANT & varCrDamp,
                                                    const VARIANT & varPhysical )
```

Creates isotropic material timber.

### Parameters

- [in] **strName** Identification title of material.
- [in] **dE** Modulus of elasticity (**E**).
- [in] **dPoisson** Poisson's ratio (**POI**).
- [in] **dG** Shear modulus (**G**).
- [in] **dDensity** Weight density (**DEN**).
- [in] **dAlpha** Coefficient of thermal expansion (**ALP**).
- [in] **dCrDamp** Damping ratio (**DAMP**).
- [in] **bPhysical** Indetifies if the material is for physical member

### Return values

- 1** Material is updated as a material with that name was already present.
- 0** Material is created.
- 1** General Error

### C++ Syntax

```
// Create material timber.
string strMaterialName = "TIMBER1";
double dE = 262040, dPoisson = 0.2, dG = 0.06, dDensity = 0.04, dAlpha = 2.78e-6, dCrDamp
    = 0.06;
int bPhysical = 0;
long RetVal = m_OStd.Property.CreateIsotropicMaterialTimber(strMaterialName, dE,
    dPoisson, dG, dDensity, dAlpha, dCrDamp, bPhysical);
```

### C# Syntax

```
// Create material timber for physical member
double dE = 262040, dPoisson = 0.2, dG = 0.06, dDensity = 0.04, dAlpha = 2.78e-6, dCrDamp
    = 0.06;
int bPhysical = 1;
long RetVal = OSPropertyUI::CreateIsotropicMaterialTimber((LPCSTR)"TIMBER1", dE,
    dPoisson, dG, dDensity, dAlpha, dCrDamp, bPhysical);
```

## VB Syntax

```
// Create material timber
Dim RetVal As Variant
Dim strMaterialName As String
Dim dE As Double
Dim dPoisson As Double
Dim dG As Double
Dim dDensity As Double
Dim dAlpha As Double
Dim dCrDamp As Double
Dim bPhysical As Integer
strMaterialName = "AluminumM124"
dE = 262040
dPoisson = 0.2
dG = 0.06
dDensity = 0.04
dAlpha = 2.78e-06
dCrDamp = 0.06
bPhysical = 0
RetVal = objOpenStaad.Property.CreateIsotropicMaterialTimber(strMaterialName, dE,
    dPoisson, dG, dDensity, dAlpha, dCrDamp, bPhysical)
```

### See also

[OSPropertyUI::CreateIsotropicMaterialSteel](#)

[OSPropertyUI::CreateIsotropicMaterialConcrete](#)

[OSPropertyUI::CreateIsotropicMaterialTimber](#)

[OSPropertyUI::CreateIsotropicMaterialProperties](#)

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