















is2GraphObject Namespace

[Missing <summary> documentation for "N:is2GraphObject"]

▲ Classes

	Class	Description
	ArcException	Representa una exception de tipo Arc Error.
	ArcType	<p>Representa un tipo Arco que puede ser definido de 7 formas.</p> <p>[1]: [Start, Any, End] [2]: [Start, Center, End] [3]: [Start, Center, Angle] [4]: [Start, Center, Length] [5]: [Start, End, Angle] [6]: [Start, End, Direction] [7]: [Start, End, Radius]</p> <p>Nota: Por convención el arco siempre queda definido en sentido antihorario.</p>
	CilindricalPointType	
	CircleException	Representa una exception para el tipo Circle Error.
	CircleType	Representa un tipo Circunferencia.
	EllipseType	Representa un tipo Elipse.
	FilletException	Representa una exception para el

		tipo Fillet Error.
	is2GraphObj	Biblioteca gráfica de entidades 2D.
	LineType	Representa un tipo Línea. Esta línea se define por un punto y un ángulo, dando como resultado una línea que pasa por el punto 'P', forma un angulo con el eje de las abcisas determinado por 'Angle' y tiene longitud infinita.
	PlaneType	Representa un tipo Plano.
	PointType	Representa un tipo Punto que puede ser definido tanto en el plano como el espacio.
	PolarPointType	Representa un tipo Punto Polar el cual es definido mediante coordenadas polares.
	PolygonException	Representa una exception de tipo Polygon Error.
	PolygonType	Representa un tipo Polígono regular convexo.
	PolylineElement	Clase fachada que representa el elemento de una polilínea.
	PolylineException	Representa una exception para el tipo Polyline Error.
	PolylineType	Representa un tipo Polilínea.
	RectangleType	Representa un tipo Rectángulo.
	SegmentType	Representa un tipo Segmento. Un

segmento se define por dos puntos.













SphericalPointType



TriangleType





Representa un tipo Triángulo.

▲ Structures

Structure	Description
 ArcAnyPoint	Representa un punto cualquiera (ANY_POINT) de un Arco.
 ArcCenterPoint	Representa el punto del centro (CENTER_POINT) de un Arco.
 ArcDistance	Representa la distancia o longitud de la cuerda de un Arco.
 ArcEndPoint	Representa el punto final (END_POINT) de un Arco.
 ArcGradeAngle	Representa el ángulo en grados de un Arco.
 ArcRadius	Representa el radio de un Arco.
 ArcStartPoint	Representa el punto de inicio (START_POINT) de un Arco.
 ArcVectorDirection	Representa la dirección de un arco. Esta dirección esta dada por un punto en el plano.
 ChamferAngle	
 ChamferDistante	

└ Enumerations

	Enumeration	Description
	ArcType ArcDirection	Define los tipos de Orientación que puede describir un arco.
	CircleCircleRelation	Describe los tipos de relaciones relativas que ocurren entre dos circunferencias.
	CircleSegmentRelation	Describe los tipos de relaciones que ocurren entre una circunferencia y en segmento de recta.
	CircleType Type	Define la posición relativa que ocupa la circunferencia con respecto a un polígono.
	Cuadrante	Define cada una de las cuatro porciones en la que descompone un sistema de 2 planos interceptados en el espacio (Frontal, Horizontal).
	Octante	Define cada una de las ocho porciones en la que descompone un sistema de 3 planos interceptados en el espacio (Frontal, Horizontal, Lateral).
	Plane	Define los 3 planos que se forman por la intersección de los ejes de coordenadas X-Y, Y-Z y X-Z.

	PointLinePosition
	PolylineElementType
	TriangleType TriangleTypeByAngle
	TriangleType TriangleTypeBySide

ArcAnyPoint Structure

Representa un punto cualquiera (ANY_POINT) de un Arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**




C#

[Copy](#)

```
public struct ArcAnyPoint
```





The `ArcAnyPoint` type exposes the following members.

▲ Constructors

	Name	Description
	ArcAnyPoint(PointType)	
	ArcAnyPoint(Double, Double)	
	ArcAnyPoint(Double, Double, Double)	


[Top](#)

▲ Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ Properties

	Name	Description
	val	

[Top](#)




▲ See Also

Reference

[is2GraphObject Namespace](#)

ArcAnyPoint Constructor

▲ Overload List

	Name	Description
	ArcAnyPoint(PointType)	
	ArcAnyPoint(Double, Double)	
	ArcAnyPoint(Double, Double, Double)	

[Top](#)

▲ See Also

Reference

[ArcAnyPoint Structure](#)

[is2GraphObject Namespace](#)

ArcAnyPoint Constructor (PointType)

[Missing <summary> documentation for
"M:is2GraphObject.ArcAnyPoint.#ctor(is2GraphObject.PointType)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcAnyPoint(  
    PointType p  
)
```

Parameters

p

Type: [is2GraphObjectPointType](#)

[Missing <param name="p"/> documentation for
"M:is2GraphObject.ArcAnyPoint.#ctor(is2GraphObject.PointType)"]

▲ See Also

Reference

[ArcAnyPoint Structure](#)

[ArcAnyPoint Overload](#)

[is2GraphObject Namespace](#)

ArcAnyPoint Constructor (Double, Double)

[Missing <summary> documentation for
"M:is2GraphObject.ArcAnyPoint.#ctor(System.Double,System.Double)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcAnyPoint(  
    double x,  
    double y  
)
```

Parameters

x

Type: [SystemDouble](#)

[Missing <param name="x"/> documentation for
"M:is2GraphObject.ArcAnyPoint.#ctor(System.Double,System.Double)"]

y

Type: [SystemDouble](#)

[Missing <param name="y"/> documentation for
"M:is2GraphObject.ArcAnyPoint.#ctor(System.Double,System.Double)"]

▲ See Also

Reference

[ArcAnyPoint Structure](#)

ArcAnyPoint Overload
is2GraphObject Namespace

ArcAnyPoint Constructor (Double, Double, Double)

[Missing <summary> documentation for
"M:is2GraphObject.ArcAnyPoint.#ctor(System.Double,System.Double,System.Double)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcAnyPoint(  
    double x,  
    double y,  
    double z  
)
```

Parameters

x

Type: [System Double](#)

[Missing <param name="x"/> documentation for

"M:is2GraphObject.ArcAnyPoint.#ctor(System.Double,System.Double,System.Double)'

y

Type: [System Double](#)

[Missing <param name="y"/> documentation for

"M:is2GraphObject.ArcAnyPoint.#ctor(System.Double,System.Double,System.Double)'

z

Type: [System Double](#)

[Missing <param name="z"/> documentation for

"M:is2GraphObject.ArcAnyPoint.#ctor(System.Double,System.Double,System.Double)'

▲ See Also

Reference

[ArcAnyPoint Structure](#)





[ArcAnyPoint Overload](#)

[is2GraphObject Namespace](#)

ArcAnyPoint Methods

The [ArcAnyPoint](#) type exposes the following members.

▲ Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ See Also

Reference


[ArcAnyPoint Structure](#)

[is2GraphObject Namespace](#)

ArcAnyPoint Properties

The [ArcAnyPoint](#) type exposes the following members.

▲ Properties

Name	Description
 val	

[Top](#)

▲ See Also

Reference

[ArcAnyPoint Structure](#)

[is2GraphObject Namespace](#)

ArcAnyPointval Property

[Missing <summary> documentation for "P:is2GraphObject.ArcAnyPoint.val"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType val { get; set; }
```

Property Value

Type: [PointType](#)

▲ See Also

Reference

[ArcAnyPoint Structure](#)

[is2GraphObject Namespace](#)

ArcCenterPoint Structure

Representa el punto del centro (CENTER_POINT) de un Arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**




C#

[Copy](#)

```
public struct ArcCenterPoint
```





The `ArcCenterPoint` type exposes the following members.

Constructors

	Name	Description
	ArcCenterPoint(PointType)	
	ArcCenterPoint(Double, Double)	
	ArcCenterPoint(Double, Double, Double)	


[Top](#)

Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ Properties

	Name	Description
	val	

[Top](#)




▲ See Also

Reference

[is2GraphObject Namespace](#)

ArcCenterPoint Constructor

▲ Overload List

	Name	Description
	ArcCenterPoint(PointType)	
	ArcCenterPoint(Double, Double)	
	ArcCenterPoint(Double, Double, Double)	

[Top](#)

▲ See Also

Reference

[ArcCenterPoint Structure](#)

[is2GraphObject Namespace](#)

ArcCenterPoint Constructor (PointType)

[Missing <summary> documentation for "M:is2GraphObject.ArcCenterPoint.#ctor(is2GraphObject.PointType)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcCenterPoint(  
    PointType p  
)
```

Parameters

p

Type: [is2GraphObject.PointType](#)

[Missing <param name="p"/> documentation for "M:is2GraphObject.ArcCenterPoint.#ctor(is2GraphObject.PointType)"]

▲ See Also

Reference

[ArcCenterPoint Structure](#)

[ArcCenterPoint Overload](#)

[is2GraphObject Namespace](#)

ArcCenterPoint Constructor (Double, Double)

[Missing <summary> documentation for
"M:is2GraphObject.ArcCenterPoint.#ctor(System.Double,System.Double)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcCenterPoint(  
    double x,  
    double y  
)
```

Parameters

x

Type: [SystemDouble](#)

[Missing <param name="x"/> documentation for
"M:is2GraphObject.ArcCenterPoint.#ctor(System.Double,System.Double)"]

y

Type: [SystemDouble](#)

[Missing <param name="y"/> documentation for
"M:is2GraphObject.ArcCenterPoint.#ctor(System.Double,System.Double)"]

▲ See Also

Reference

[ArcCenterPoint Structure](#)

ArcCenterPoint Overload
is2GraphObject Namespace

ArcCenterPoint Constructor (Double, Double, Double)

[Missing <summary> documentation for
"M:is2GraphObject.ArcCenterPoint.#ctor(System.Double,System.Double,System.Double)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcCenterPoint(  
    double x,  
    double y,  
    double z  
)
```

Parameters

x

Type: [System.Double](#)

[Missing <param name="x"/> documentation for

"M:is2GraphObject.ArcCenterPoint.#ctor(System.Double,System.Double,System.Double)"]

y

Type: [System.Double](#)

[Missing <param name="y"/> documentation for

"M:is2GraphObject.ArcCenterPoint.#ctor(System.Double,System.Double,System.Double)"]

z

Type: [System.Double](#)

[Missing <param name="z"/> documentation for

"M:is2GraphObject.ArcCenterPoint.#ctor(System.Double,System.Double,System.Double)"]

▲ See Also

Reference

[ArcCenterPoint Structure](#)





[ArcCenterPoint Overload](#)

[is2GraphObject Namespace](#)

ArcCenterPoint Methods

The [ArcCenterPoint](#) type exposes the following members.

▲ Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ See Also

Reference


[ArcCenterPoint](#) Structure

[is2GraphObject](#) Namespace

ArcCenterPoint Properties

The [ArcCenterPoint](#) type exposes the following members.

▲ Properties

Name	Description
 val	

[Top](#)

▲ See Also

Reference

[ArcCenterPoint Structure](#)

[is2GraphObject Namespace](#)

ArcCenterPointval Property

[Missing <summary> documentation for "P:is2GraphObject.ArcCenterPoint.val"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType val { get; set; }
```

Property Value

Type: [PointType](#)

▲ See Also

Reference

[ArcCenterPoint Structure](#)

[is2GraphObject Namespace](#)

ArcDistance Structure

Representa la distancia o longitud de la cuerda de un Arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**


C#

[Copy](#)

```
public struct ArcDistance
```





The `ArcDistance` type exposes the following members.

▲ Constructors

Name	Description
 ArcDistance	

[Top](#)

▲ Methods

Name	Description
 Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
 GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
 GetType	Gets the Type of the current instance. (Inherited from Object .)
 ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ Properties

Name	Description
 val	

[Top](#)

▲ See Also

Reference

[is2GraphObject Namespace](#)

ArcDistance Constructor

[Missing <summary> documentation for
"M:is2GraphObject.ArcDistance.#ctor(System.Double)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcDistance(  
    double d  
)
```

Parameters

d

Type: [SystemDouble](#)

[Missing <param name="d"/> documentation for
"M:is2GraphObject.ArcDistance.#ctor(System.Double)"]

▲ See Also

Reference





[ArcDistance Structure](#)

[is2GraphObject Namespace](#)

ArcDistance Methods

The [ArcDistance](#) type exposes the following members.

▲ Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ See Also

Reference


[ArcDistance Structure](#)

[is2GraphObject Namespace](#)

ArcDistance Properties

The [ArcDistance](#) type exposes the following members.

▲ Properties

Name	Description
 val	

[Top](#)

▲ See Also

Reference

[ArcDistance](#) Structure

[is2GraphObject](#) Namespace

ArcDistanceeval Property

[Missing <summary> documentation for "P:is2GraphObject.ArcDistance.val"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double val { get; set; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[ArcDistance Structure](#)

[is2GraphObject Namespace](#)

ArcEndPoint Structure

Representa el punto final (END_POINT) de un Arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**




C#

[Copy](#)

```
public struct ArcEndPoint
```





The `ArcEndPoint` type exposes the following members.

Constructors

	Name	Description
	ArcEndPoint(PointType)	
	ArcEndPoint(Double, Double)	
	ArcEndPoint(Double, Double, Double)	


[Top](#)

Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ Properties

	Name	Description
	val	

[Top](#)




▲ See Also

Reference

[is2GraphObject Namespace](#)

ArcEndPoint Constructor

▲ Overload List

	Name	Description
	ArcEndPoint(PointType)	
	ArcEndPoint(Double, Double)	
	ArcEndPoint(Double, Double, Double)	

[Top](#)

▲ See Also

Reference

[ArcEndPoint Structure](#)

[is2GraphObject Namespace](#)

ArcEndPoint Constructor (PointType)

[Missing <summary> documentation for
"M:is2GraphObject.ArcEndPoint.#ctor(is2GraphObject.PointType)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcEndPoint(  
    PointType p  
)
```

Parameters

p

Type: [is2GraphObjectPointType](#)

[Missing <param name="p"/> documentation for
"M:is2GraphObject.ArcEndPoint.#ctor(is2GraphObject.PointType)"]

▲ See Also

Reference

[ArcEndPoint Structure](#)

[ArcEndPoint Overload](#)

[is2GraphObject Namespace](#)

ArcEndPoint Constructor (Double, Double)

[Missing <summary> documentation for
"M:is2GraphObject.ArcEndPoint.#ctor(System.Double,System.Double)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcEndPoint(  
    double x,  
    double y  
)
```

Parameters

x

Type: [SystemDouble](#)

[Missing <param name="x"/> documentation for
"M:is2GraphObject.ArcEndPoint.#ctor(System.Double,System.Double)"]

y

Type: [SystemDouble](#)

[Missing <param name="y"/> documentation for
"M:is2GraphObject.ArcEndPoint.#ctor(System.Double,System.Double)"]

▲ See Also

Reference

[ArcEndPoint Structure](#)

ArcEndPoint Overload
is2GraphObject Namespace

ArcEndPoint Constructor (Double, Double, Double)

[Missing <summary> documentation for
"M:is2GraphObject.ArcEndPoint.#ctor(System.Double,System.Double,System.Double)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcEndPoint(  
    double x,  
    double y,  
    double z  
)
```

Parameters

x

Type: [System Double](#)

[Missing <param name="x"/> documentation for

"M:is2GraphObject.ArcEndPoint.#ctor(System.Double,System.Double,System.Double)'

y

Type: [System Double](#)

[Missing <param name="y"/> documentation for

"M:is2GraphObject.ArcEndPoint.#ctor(System.Double,System.Double,System.Double)'

z

Type: [System Double](#)

[Missing <param name="z"/> documentation for

"M:is2GraphObject.ArcEndPoint.#ctor(System.Double,System.Double,System.Double)'

▲ See Also

Reference

[ArcEndPoint Structure](#)





[ArcEndPoint Overload](#)

[is2GraphObject Namespace](#)

ArcEndPoint Methods

The [ArcEndPoint](#) type exposes the following members.

▲ Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ See Also

Reference


[ArcEndPoint Structure](#)

[is2GraphObject Namespace](#)

ArcEndPoint Properties

The [ArcEndPoint](#) type exposes the following members.

▲ Properties

Name	Description
 val	

[Top](#)

▲ See Also

Reference

[ArcEndPoint Structure](#)

[is2GraphObject Namespace](#)

ArcEndPointval Property

[Missing <summary> documentation for "P:is2GraphObject.ArcEndPoint.val"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType val { get; set; }
```

Property Value

Type: [PointType](#)

▲ See Also

Reference

[ArcEndPoint Structure](#)

[is2GraphObject Namespace](#)

ArcException Class

Representa una exception de tipo Arc Error.

▲ Inheritance Hierarchy [System Object System](#)

[Exception](#)

[is2GraphObject ArcException](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**



C#

[Copy](#)

```
public class ArcException : Exception
```





The `ArcException` type exposes the following members.

▲ Constructors





	Name	Description
	ArcException	Constructor por defecto.
	ArcException(String)	Representa una exception de tipo Arc Error.

[Top](#)

▲ Methods




	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash






function.
(Inherited from [Object](#).)

	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

[Top](#)


▲ Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)

	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

[Top](#)

▲ Events

Name		Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

[Top](#)



▲ See Also

Reference

is2GraphObject Namespace

ArcException Constructor

▲ Overload List

	Name	Description
	ArcException	Constructor por defecto.
	ArcException(String)	Representa una exception de tipo Arc Error.

[Top](#)

▲ See Also

Reference

[ArcException Class](#)

[is2GraphObject Namespace](#)

ArcException Constructor

Constructor por defecto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcException()
```

▲ See Also

Reference

[ArcException Class](#)

[ArcException Overload](#)

[is2GraphObject Namespace](#)

ArcException Constructor (String)

Representa una exception de tipo Arc Error.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcException(  
    string msg  
)
```

Parameters

msg

Type: [SystemString](#)

[Missing <param name="msg"/> documentation for
"M:is2GraphObject.ArcException.#ctor(System.String)"]

▲ See Also

Reference

[ArcException Class](#)

[ArcException Overload](#)







[is2GraphObject Namespace](#)

[SystemString](#)

ArcException Methods

The [ArcException](#) type exposes the following members.

▲ Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current

instance.
(Inherited from [Exception.](#))



[MemberwiseClone](#)

Creates a shallow copy of the current [Object](#).
(Inherited from [Object.](#))



[ToString](#)

Creates and returns a string representation of the current exception.
(Inherited from [Exception.](#))

[Top](#)

▲ See Also

Reference








[ArcException Class](#)

[is2GraphObject Namespace](#)

ArcException Properties

The [ArcException](#) type exposes the following members.

▲ Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the

immediate frames on the call stack.
(Inherited from [Exception](#).)



[TargetSite](#)

Gets the method that throws the current exception.
(Inherited from [Exception](#).)

[Top](#)

▲ See Also

Reference


[ArcException Class](#)

[is2GraphObject Namespace](#)

ArcException Events

The [ArcException](#) type exposes the following members.

▲ Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

[Top](#)

▲ See Also

Reference

[ArcException Class](#)

[is2GraphObject Namespace](#)

ArcGradeAngle Structure

Representa el ángulo en grados de un Arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:


1.0.0.0 (1.0.0.0) ▲ **Syntax**

```
C#  
public struct ArcGradeAngle
```

[Copy](#)





The `ArcGradeAngle` type exposes the following members.

▲ Constructors

Name	Description
 ArcGradeAngle	

[Top](#)

▲ Methods

Name	Description
 Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
 GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
 GetType	Gets the Type of the current instance. (Inherited from Object .)
 ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ Properties

Name	Description
 ToRadian	



val

[Top](#)

▲ [See Also](#)

Reference

[is2GraphObject Namespace](#)

ArcGradeAngle Constructor

[Missing <summary> documentation for
"M:is2GraphObject.ArcGradeAngle.#ctor(System.Double)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcGradeAngle(  
    double a  
)
```

Parameters

a

Type: [System.Double](#)

[Missing <param name="a"/> documentation for
"M:is2GraphObject.ArcGradeAngle.#ctor(System.Double)"]

▲ See Also

Reference





[ArcGradeAngle Structure](#)

[is2GraphObject Namespace](#)

ArcGradeAngle Methods

The [ArcGradeAngle](#) type exposes the following members.

▲ Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ See Also

Reference



[ArcGradeAngle](#) Structure

[is2GraphObject](#) Namespace

ArcGradeAngle Properties

The [ArcGradeAngle](#) type exposes the following members.

▲ Properties

	Name	Description
	ToRadian	
	val	

[Top](#)

▲ See Also

Reference

[ArcGradeAngle Structure](#)

[is2GraphObject Namespace](#)

ArcGradeAngleToRadian Property

[Missing <summary> documentation for "P:is2GraphObject.ArcGradeAngle.ToRadian"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double ToRadian { get; }
```

Return Value

Type: [Double](#)

▲ See Also

Reference

[ArcGradeAngle Structure](#)

[is2GraphObject Namespace](#)

ArcGradeAngleval Property

[Missing <summary> documentation for "P:is2GraphObject.ArcGradeAngle.val"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double val { get; set; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[ArcGradeAngle](#) Structure

[is2GraphObject](#) Namespace

ArcRadius Structure

Representa el radio de un Arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**


C#

[Copy](#)

```
public struct ArcRadius
```





The **ArcRadius** type exposes the following members.

▲ Constructors

Name	Description
 ArcRadius	

[Top](#)

▲ Methods

Name	Description
 Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
 GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
 GetType	Gets the Type of the current instance. (Inherited from Object .)
 ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ Properties

Name	Description
 val	

[Top](#)

▲ [See Also](#)

Reference

[is2GraphObject Namespace](#)

ArcRadius Constructor

[Missing <summary> documentation for
"M:is2GraphObject.ArcRadius.#ctor(System.Double)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcRadius(  
    double r  
)
```

Parameters

r

Type: [SystemDouble](#)

[Missing <param name="r"/> documentation for
"M:is2GraphObject.ArcRadius.#ctor(System.Double)"]

▲ See Also

Reference





[ArcRadius Structure](#)

[is2GraphObject Namespace](#)

ArcRadius Methods

The [ArcRadius](#) type exposes the following members.

Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

See Also

Reference


[ArcRadius Structure](#)

[is2GraphObject Namespace](#)

ArcRadius Properties

The [ArcRadius](#) type exposes the following members.

▲ Properties

Name	Description
 val	

[Top](#)

▲ See Also

Reference

[ArcRadius Structure](#)

[is2GraphObject Namespace](#)

ArcRadiusval Property

[Missing <summary> documentation for "P:is2GraphObject.ArcRadius.val"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double val { get; set; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[ArcRadius Structure](#)

[is2GraphObject Namespace](#)

ArcStartPoint Structure

Representa el punto de inicio (START_POINT) de un Arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**




C#

[Copy](#)

```
public struct ArcStartPoint
```





The `ArcStartPoint` type exposes the following members.

Constructors

	Name	Description
	ArcStartPoint(PointType)	
	ArcStartPoint(Double, Double)	
	ArcStartPoint(Double, Double, Double)	


[Top](#)

Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ Properties

	Name	Description
	val	

[Top](#)




▲ See Also

Reference

[is2GraphObject Namespace](#)

ArcStartPoint Constructor

▲ Overload List

	Name	Description
	ArcStartPoint(PointType)	
	ArcStartPoint(Double, Double)	
	ArcStartPoint(Double, Double, Double)	

[Top](#)

▲ See Also

Reference

[ArcStartPoint Structure](#)

[is2GraphObject Namespace](#)

ArcStartPoint Constructor (PointType)

[Missing <summary> documentation for
"M:is2GraphObject.ArcStartPoint.#ctor(is2GraphObject.PointType)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcStartPoint(  
    PointType p  
)
```

Parameters

p

Type: [is2GraphObject.PointType](#)

[Missing <param name="p"/> documentation for
"M:is2GraphObject.ArcStartPoint.#ctor(is2GraphObject.PointType)"]

▲ See Also

Reference

[ArcStartPoint Structure](#)

[ArcStartPoint Overload](#)

[is2GraphObject Namespace](#)

ArcStartPoint Constructor (Double, Double)

[Missing <summary> documentation for
"M:is2GraphObject.ArcStartPoint.#ctor(System.Double,System.Double)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcStartPoint(  
    double x,  
    double y  
)
```

Parameters

x

Type: [SystemDouble](#)

[Missing <param name="x"/> documentation for
"M:is2GraphObject.ArcStartPoint.#ctor(System.Double,System.Double)"]

y

Type: [SystemDouble](#)

[Missing <param name="y"/> documentation for
"M:is2GraphObject.ArcStartPoint.#ctor(System.Double,System.Double)"]

▲ See Also

Reference

[ArcStartPoint Structure](#)

ArcStartPoint Overload
is2GraphObject Namespace

ArcStartPoint Constructor (Double, Double, Double)

[Missing <summary> documentation for
"M:is2GraphObject.ArcStartPoint.#ctor(System.Double,System.Double,System.Double)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcStartPoint(  
    double x,  
    double y,  
    double z  
)
```

Parameters

x

Type: [System Double](#)

[Missing <param name="x"/> documentation for

"M:is2GraphObject.ArcStartPoint.#ctor(System.Double,System.Double,System.Double)

y

Type: [System Double](#)

[Missing <param name="y"/> documentation for

"M:is2GraphObject.ArcStartPoint.#ctor(System.Double,System.Double,System.Double)

z

Type: [System Double](#)

[Missing <param name="z"/> documentation for

"M:is2GraphObject.ArcStartPoint.#ctor(System.Double,System.Double,System.Double)

▲ See Also

Reference

[ArcStartPoint Structure](#)





[ArcStartPoint Overload](#)

[is2GraphObject Namespace](#)

ArcStartPoint Methods

The [ArcStartPoint](#) type exposes the following members.

Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

See Also

Reference


[ArcStartPoint Structure](#)

[is2GraphObject Namespace](#)

ArcStartPoint Properties

The [ArcStartPoint](#) type exposes the following members.

▲ Properties

Name	Description
 val	

[Top](#)

▲ See Also

Reference

[ArcStartPoint Structure](#)

[is2GraphObject Namespace](#)

ArcStartPointval Property

[Missing <summary> documentation for "P:is2GraphObject.ArcStartPoint.val"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType val { get; set; }
```

Property Value

Type: [PointType](#)

▲ See Also

Reference

[ArcStartPoint Structure](#)

[is2GraphObject Namespace](#)

ArcType Class

Representa un tipo Arco que puede ser definido de 7 formas.

[1]: [Start, Any, End]

[2]: [Start, Center, End]

[3]: [Start, Center, Angle]

[4]: [Start, Center, Length]

[5]: [Start, End, Angle]

[6]: [Start, End, Direction]

[7]: [Start, End, Radius]

Nota: Por convención el arco siempre queda definido en sentido antihorario.

▲ Inheritance Hierarchy [System.Object](#)

[is2GraphObject](#) [ArcType](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**









C#



[Copy](#)

```
public class ArcType
```

The `ArcType` type exposes the following members.






Constructors







	Name	Description
	<code>ArcType</code>	Constructor por defecto.
	<code>ArcType(ArcStartPoint, ArcAnyPoint, ArcEndPoint)</code>	[1]: Crea un Arco usando 3 puntos [Start, Any, End].
	<code>ArcType(ArcType, ArcEndPoint, Boolean)</code>	[9]: Crea un Arco continuo dado un arco y un punto [Arc_continue, End].
	<code>ArcType(SegmentType, ArcEndPoint, Boolean)</code>	[8]: Crea un Arco continuo dado una recta y un punto [Line_continue, End].
	<code>ArcType(ArcStartPoint, ArcCenterPoint, ArcDistance, Boolean)</code>	[4]: Crea un Arco usando 2 puntos y una longitud [Start, Center, Length].
	<code>ArcType(ArcStartPoint, ArcCenterPoint, ArcEndPoint, Boolean)</code>	[2]: Crea un Arco usando 3 puntos [Start, Center, End].
	<code>ArcType(ArcStartPoint, ArcCenterPoint, ArcGradeAngle, Boolean)</code>	[3]: Crea un Arco usando 2 puntos y un ángulo [Start, Center, Angle].
	<code>ArcType(ArcStartPoint,</code>	Variante 5: Crea un Arco

	ArcEndPoint, ArcGradeAngle, Boolean)	usando 2 puntos y un ángulo [Start, End, angle].
	ArcType(ArcStartPoint, ArcEndPoint, ArcRadius, Boolean)	[7]: Crea un Arco usando 2 puntos y radio [Start, End, Radius]
	ArcType(ArcStartPoint, ArcEndPoint, ArcVectorDirection, Boolean)	[6]: Crea un Arco usando 2 puntos y una dirección [Start, End, Direction].

[Top](#)


Methods






	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetArc3Points	Devuelve por referencia los puntos [Start], [Mid] y [End] del Arco.
 	GetArcDirection_2P	Devuelve la orientacion del arco en {HORARIO / ANTIHORARIO} según sus puntos Start y End.

	GetArcDirection_3P	Devuelve la orientación del arco en {HORARIO / ANTIHORARIO} analizando la disposición de 3 de sus puntos (start, any y end).
	GetArcDirectionByAngle	Devuelve la orientación del arco en {HORARIO / ANTIHORARIO} según el ángulo de inicio, ángulo final y ángulo de barrido.
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

[Top](#)

▲ Properties

	Name	Description
	Angle	Representa el ángulo del arco.
	Center	Representa el punto centro del arco.
	EndAngle	Representa el ángulo final del arco.

	EndPoint	Representa el punto final del arco.
	Longitude	Propiedad de solo lectura. Devuelve la longitud del arco.
	MidPoint	Representa el punto medio del arco.
	Radius	Representa el radio del arco.
	StartAngle	Representa el ángulo de inicio del arco.
	StartPoint	Representa el punto de inicio del arco.

[Top](#)









▲ [See Also](#)

Reference

[is2GraphObject Namespace](#)

ArcType Constructor

▲ Overload List

	Name	Description
	ArcType	Constructor por defecto.
	ArcType(ArcStartPoint, ArcAnyPoint, ArcEndPoint)	[1]: Crea un Arco usando 3 puntos [Start, Any, End].
	ArcType(ArcType, ArcEndPoint, Boolean)	[9]: Crea un Arco continuo dado un arco y un punto [Arc_continue, End].
	ArcType(SegmentType, ArcEndPoint, Boolean)	[8]: Crea un Arco continuo dado una recta y un punto [Line_continue, End].
	ArcType(ArcStartPoint, ArcCenterPoint, ArcDistance, Boolean)	[4]: Crea un Arco usando 2 puntos y una longitud [Start, Center, Length].
	ArcType(ArcStartPoint, ArcCenterPoint, ArcEndPoint, Boolean)	[2]: Crea un Arco usando 3 puntos [Start, Center, End].
	ArcType(ArcStartPoint, ArcCenterPoint, ArcGradeAngle, Boolean)	[3]: Crea un Arco usando 2 puntos y un ángulo [Start, Center, Angle].
	ArcType(ArcStartPoint, ArcEndPoint, ArcGradeAngle, Boolean)	Variante 5: Crea un Arco usando 2 puntos y un ángulo [Start, End, angle].



[ArcType\(ArcStartPoint, ArcEndPoint, ArcRadius, Boolean\)](#)

[7]: Crea un Arco usando 2 puntos y radio [Start, End, Radius]



[ArcType\(ArcStartPoint, ArcEndPoint, ArcVectorDirection, Boolean\)](#)

[6]: Crea un Arco usando 2 puntos y una dirección [Start, End, Direction].

[Top](#)

▲ See Also

Reference

[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcType Constructor

Constructor por defecto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcType()
```

▲ See Also

Reference

[ArcType Class](#)

[ArcType Overload](#)

[is2GraphObject Namespace](#)

ArcType Constructor (ArcStartPoint, ArcAnyPoint, ArcEndPoint)

[1]: Crea un Arco usando 3 puntos [Start, Any, End].

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public ArcType(  
    ArcStartPoint start,  
    ArcAnyPoint any,  
    ArcEndPoint end  
)
```

Parameters

start

Type: [is2GraphObject ArcStartPoint](#)
Punto de inicio de arco (Startpoint).

any

Type: [is2GraphObject ArcAnyPoint](#)
Un punto cualquiera del arco (Anypoint).

end

Type: [is2GraphObject ArcEndPoint](#)
Punto final del arco (Endpoint).

▲ See Also

Reference

[ArcType Class](#)

[ArcType Overload](#)

[is2GraphObject Namespace](#)

ArcType Constructor (ArcType, ArcEndPoint, Boolean)

[9]: Crea un Arco continuo dado un arco y un punto [Arc_continue, End].

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public ArcType(  
    ArcType A,  
    ArcEndPoint end,  
    bool inverse = false  
)
```

Parameters

A

Type: [is2GraphObject ArcType](#)

Arco de referencia el cual es tangente al arco a crear.

end

Type: [is2GraphObject ArcEndPoint](#)

Punto final del arco a crear (Endpoint).

inverse (**Optional**)

Type: [System Boolean](#)

Indica si se invierte el sentido de barrido por defecto del arco.

See Also

Reference

[ArcType Class](#)

[ArcType Overload](#)

[is2GraphObject Namespace](#)

ArcType Constructor (SegmentType, ArcEndPoint, Boolean)

[8]: Crea un Arco continuo dado una recta y un punto
[Line_continue, End].

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public ArcType(  
    SegmentType S,  
    ArcEndPoint end,  
    bool inverse = false  
)
```

Parameters

S

Type: [is2GraphObject SegmentType](#)

Segmento de referencia el cual es tangente al arco.

end

Type: [is2GraphObject ArcEndPoint](#)

Punto final del arco (Endpoint).

inverse (Optional)

Type: [System Boolean](#)

[Missing <param name="inverse"/> documentation for

"M:is2GraphObject.ArcType.#ctor(is2GraphObject.SegmentType,is2GraphObject.ArcE

See Also

Reference

[ArcType Class](#)

[ArcType Overload](#)

[is2GraphObject Namespace](#)

ArcType Constructor

(ArcStartPoint, ArcCenterPoint, ArcDistance, Boolean)

[4]: Crea un Arco usando 2 puntos y una longitud [Start, Center, Length].

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▴ **Syntax**

C#

[Copy](#)

```
public ArcType(  
    ArcStartPoint start,  
    ArcCenterPoint center,  
    ArcDistance length,  
    bool inverse = false  
)
```

Parameters

start

Type: [is2GraphObject ArcStartPoint](#)
Punto de inicio del arco (Startpoint).

center

Type: [is2GraphObject ArcCenterPoint](#)
Punto centro del arco (Centerpoint).

length

Type: [is2GraphObject ArcDistance](#)
Distancia de la cuerda (Lenght).

inverse (Optional)

Type: [System Boolean](#)
Indica si se invierte el sentido de barrido por defecto del arco.

▲ Exceptions

Exception	Condition
ArcException	Se lanza cuando se intenta crear un arco cuyo valor de longitud de la cuerda es incorrecto.

▲ See Also

Reference

[ArcType Class](#)

[ArcType Overload](#)

[is2GraphObject Namespace](#)

ArcType Constructor (ArcStartPoint, ArcCenterPoint, ArcEndPoint, Boolean)

[2]: Crea un Arco usando 3 puntos [Start, Center, End].

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public ArcType(  
    ArcStartPoint start,  
    ArcCenterPoint center,  
    ArcEndPoint end,  
    bool inverse = false  
)
```

Parameters

start

Type: [is2GraphObject ArcStartPoint](#)
Punto de inicio del arco (Startpoint).

center

Type: [is2GraphObject ArcCenterPoint](#)
Punto centro del arco (Centerpoint).

end

Type: [is2GraphObject ArcEndPoint](#)
Punto final del arco (Endpoint).

inverse (Optional)

Type: [System Boolean](#)
Indica si se invierte el sentido de barrido por defecto del arco.

See Also

Reference

[ArcType Class](#)

[ArcType Overload](#)

[is2GraphObject Namespace](#)

ArcType Constructor

(ArcStartPoint, ArcCenterPoint, ArcGradeAngle, Boolean)

[3]: Crea un Arco usando 2 puntos y un ángulo [Start, Center, Angle].

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▴ **Syntax**

C#

[Copy](#)

```
public ArcType(  
    ArcStartPoint start,  
    ArcCenterPoint center,  
    ArcGradeAngle angle,  
    bool inverse = false  
)
```

Parameters

start

Type: [is2GraphObject ArcStartPoint](#)
Punto de inicio del arco (Startpoint).

center

Type: [is2GraphObject ArcCenterPoint](#)
Punto de centro del arco (Centerpoint).

angle

Type: [is2GraphObject ArcGradeAngle](#)
Ángulo del arco (Angle).

inverse (Optional)

Type: [System Boolean](#)
Indica si se invierte el sentido de barrido por defecto del arco.

▲ See Also

Reference

[ArcType Class](#)

[ArcType Overload](#)

[is2GraphObject Namespace](#)

ArcType Constructor

(ArcStartPoint, ArcEndPoint, ArcGradeAngle, Boolean)

Variante 5: Crea un Arco usando 2 puntos y un ángulo [Start, End, angle].

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▀ **Syntax**

C#

[Copy](#)

```
public ArcType(  
    ArcStartPoint start,  
    ArcEndPoint end,  
    ArcGradeAngle angle,  
    bool inverse = false  
)
```

Parameters

start

Type: [is2GraphObject ArcStartPoint](#)
Punto de inicio del arco (Startpoint).

end

Type: [is2GraphObject ArcEndPoint](#)
Punto final del arco (Endpoint).

angle

Type: [is2GraphObject ArcGradeAngle](#)
Ángulo del arco (Angle).

inverse (Optional)

Type: [System Boolean](#)
Indica si se invierte el sentido de barrido por defecto del arco.

▲ See Also

Reference

[ArcType Class](#)

[ArcType Overload](#)

[is2GraphObject Namespace](#)

ArcType Constructor (ArcStartPoint, ArcEndPoint, ArcRadius, Boolean)

[7]: Crea un Arco usando 2 puntos y radio [Start, End, Radius]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public ArcType(  
    ArcStartPoint start,  
    ArcEndPoint end,  
    ArcRadius radius,  
    bool inverse = false  
)
```

Parameters

start

Type: [is2GraphObject ArcStartPoint](#)
Punto de inicio del arco (Startpoint).

end

Type: [is2GraphObject ArcEndPoint](#)
Punto final del arco (Endpoint).

radius

Type: [is2GraphObject ArcRadius](#)
Radio del arco (Radius).

inverse (Optional)

Type: [System Boolean](#)
Indica si se invierte el sentido de barrido por defecto del arco.

Exceptions

Exception	Condition
ArcException	Se lanza cuando se intenta crear un arco con un valor de radio incorrecto.

See Also

Reference

[ArcType Class](#)

[ArcType Overload](#)

[is2GraphObject Namespace](#)

ArcType Constructor

(ArcStartPoint, ArcEndPoint, ArcVectorDirection, Boolean)

[6]: Crea un Arco usando 2 puntos y una dirección [Start, End, Direction].

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▴ **Syntax**

C#

[Copy](#)

```
public ArcType(  
    ArcStartPoint start,  
    ArcEndPoint end,  
    ArcVectorDirection direction,  
    bool inverse = false  
)
```

Parameters

start

Type: [is2GraphObject ArcStartPoint](#)
Punto de inicio del arco (Startpoint).

end

Type: [is2GraphObject ArcEndPoint](#)
Punto final del arco (Endpoint).

direction

Type: [is2GraphObject ArcVectorDirection](#)
Punto que define un vector dirección que es tangente al arco en el punto de inicio (ArcDirection).

inverse (**Optional**)

Type: [System Boolean](#)
Indica si se invierte el sentido de barrido por defecto del arco.

▲ See Also

Reference

[ArcType Class](#)

[ArcType Overload](#)

[is2GraphObject Namespace](#)

ArcType Methods

The [ArcType](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetArc3Points	Devuelve por referencia los puntos [Start], [Mid] y [End] del Arco.
 	GetArcDirection_2P	Duevuelve la orientacion del arco en {HORARIO / ANTIHORARIO} según sus puntos Start y End.
 	GetArcDirection_3P	Duevuelve la orientacion del arco en {HORARIO / ANTIHORARIO} analizado la disposicion de 3 de sus puntos (start, any y end).
 	GetArcDirectionByAngle	Duevuelve la orientacion del

arco en {HORARIO / ANTIHORARIO} según el ángulo de inicio, ángulo final y ángulo de barribo.



[GetHashCode](#)

Serves as the default hash function.
(Inherited from [Object](#).)



[GetType](#)

Gets the [Type](#) of the current instance.
(Inherited from [Object](#).)



[MemberwiseClone](#)

Creates a shallow copy of the current [Object](#).
(Inherited from [Object](#).)



[ToString](#)

Returns a string that represents the current object.
(Inherited from [Object](#).)

[Top](#)

▲ See Also

Reference

[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcType GetArc3Points Method

Devuelve por referencia los puntos [Start], [Mid] y [End] del Arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public void GetArc3Points(  
    out PointType Pi,  
    out PointType Pm,  
    out PointType Pf  
)
```

Parameters

Pi

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto de inicio del arco.

Pm

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia punto medio del arco.

Pf

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto final del arco.

▲ See Also

Reference

[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcType GetArcDirection_2P Method

Devuelve la orientación del arco en {HORARIO / ANTIHORARIO} según sus puntos Start y End.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static ArcType ArcDirection GetArcDirection  
    PointType start,  
    PointType end  
)
```

Parameters

start

Type: [is2GraphObject PointType](#)

Punto de inicio del arco.

end

Type: [is2GraphObject PointType](#)

Punto final del arco.

Return Value

Type: [ArcType ArcDirection](#)

[Missing <returns> documentation for

"M:is2GraphObject.ArcType.GetArcDirection_2P(is2GraphObject.PointType,is2GraphObj

See Also

Reference

[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcType GetArcDirection_3P Method

Devuelve la orientacion del arco en {HORARIO / ANTIHORARIO} analizado la disposicion de 3 de sus puntos (start, any y end).

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static ArcType ArcDirection GetArcDirection(  
    PointType start,  
    PointType any,  
    PointType end  
)
```

Parameters

start

Type: [is2GraphObject PointType](#)
Punto de inicio del arco.

any

Type: [is2GraphObject PointType](#)
Un punto cualquiera del arco.

end

Type: [is2GraphObject PointType](#)
Punto final del arco.

Return Value

Type: [ArcType ArcDirection](#)

[Missing <returns> documentation for

"M:is2GraphObject.ArcType.GetArcDirection_3P(is2GraphObject.PointType,is2GraphObj

See Also

Reference

[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcType GetArcDirectionByAngle Method

Devuelve la orientación del arco en {HORARIO / ANTIHORARIO} según el ángulo de inicio, ángulo final y ángulo de barrido.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static ArcType ArcDirection GetArcDirection(  
    double startAngle,  
    double endAngle,  
    double sweepAngle  
)
```

Parameters

startAngle

Type: [System Double](#)

Ángulo de inicio en grados.

endAngle

Type: [System Double](#)

Ángulo de final en grados.

sweepAngle

Type: [System Double](#)

Ángulo de barrido en grados.

Return Value

Type: [ArcType ArcDirection](#)

[Missing <returns> documentation for

"M:is2GraphObject.ArcType.GetArcDirectionByAngle(System.Double,System.Double,Sy:

See Also

Reference





[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcType Properties

The [ArcType](#) type exposes the following members.

▲ Properties

	Name	Description
	Angle	Representa el angulo del arco.
	Center	Representa el punto centro del arco.
	EndAngle	Representa el ángulo final del arco.
	EndPoint	Representa el punto final del arco.
	Longitude	Propiedad de solo lectura. Devuelve la longitud del arco.
	MidPoint	Representa el punto medio del arco.
	Radius	Representa el radio del arco.
	StartAngle	Representa el ángulo de inicio del arco.
	StartPoint	Representa el punto de inicio del arco.

[Top](#)

▲ See Also

Reference

[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcTypeAngle Property

Representa el angulo del arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Angle { get; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcTypeCenter Property

Representa el punto centro del arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType Center { get; set; }
```

Property Value

Type: [PointType](#)

▲ See Also

Reference

[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcTypeEndAngle Property

Representa el ángulo final del arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double EndAngle { get; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcTypeEndPoint Property

Representa el punto final del arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType EndPoint { get; }
```

Property Value

Type: [PointType](#)

▲ See Also

Reference

[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcTypeLongitude Property

Propiedad de solo lectura. Devuelve la longitud del arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Longitude { get; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcTypeMidPoint Property

Representa el punto medio del arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType MidPoint { get; }
```

Property Value

Type: [PointType](#)

▲ See Also

Reference

[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcTypeRadius Property

Representa el radio del arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Radius { get; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcTypeStartAngle Property

Representa el ángulo de inicio del arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double StartAngle { get; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcTypeStartPoint Property

Representa el punto de inicio del arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType StartPoint { get; }
```

Property Value

Type: [PointType](#)

▲ See Also

Reference

[ArcType Class](#)

[is2GraphObject Namespace](#)

ArcTypeArcDirection Enumeration

Define los tipos de Orientación que puede describir un arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public enum ArcDirection
```

▲ Members

Member name	Value	Description
AntiHorario	0	Representa el sentido Anti-horario.
Horario	1	Representa el sentido Horario.

▲ See Also

Reference

[is2GraphObject Namespace](#)

ArcVectorDirection Structure

Representa la dirección de un arco. Esta dirección esta dada por un punto en el plano.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**




C#

[Copy](#)

```
public struct ArcVectorDirection
```





The `ArcVectorDirection` type exposes the following members.

Constructors

	Name	Description
	ArcVectorDirection(PointType)	
	ArcVectorDirection(Double, Double)	
	ArcVectorDirection(Double, Double, Double)	


[Top](#)

Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ Properties

	Name	Description
	val	

[Top](#)




▲ See Also

Reference

[is2GraphObject Namespace](#)

ArcVectorDirection Constructor

▲ Overload List

	Name	Description
	ArcVectorDirection(PointType)	
	ArcVectorDirection(Double, Double)	
	ArcVectorDirection(Double, Double, Double)	

[Top](#)

▲ See Also

Reference

[ArcVectorDirection Structure](#)
[is2GraphObject Namespace](#)

ArcVectorDirection Constructor (PointType)

[Missing <summary> documentation for
"M:is2GraphObject.ArcVectorDirection.#ctor(is2GraphObject.PointType)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcVectorDirection(  
    PointType p  
)
```

Parameters

p

Type: [is2GraphObject.PointType](#)

[Missing <param name="p"/> documentation for
"M:is2GraphObject.ArcVectorDirection.#ctor(is2GraphObject.PointType)"]

▲ See Also

Reference

[ArcVectorDirection Structure](#)

[ArcVectorDirection Overload](#)

[is2GraphObject Namespace](#)

ArcVectorDirection Constructor (Double, Double)

[Missing <summary> documentation for "M:is2GraphObject.ArcVectorDirection.#ctor(System.Double,System.Double)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcVectorDirection(  
    double x,  
    double y  
)
```

Parameters

x

Type: [SystemDouble](#)

[Missing <param name="x"/> documentation for "M:is2GraphObject.ArcVectorDirection.#ctor(System.Double,System.Double)"]

y

Type: [SystemDouble](#)

[Missing <param name="y"/> documentation for "M:is2GraphObject.ArcVectorDirection.#ctor(System.Double,System.Double)"]

▲ See Also

Reference

[ArcVectorDirection Structure](#)

ArcVectorDirection Overload
is2GraphObject Namespace

ArcVectorDirection Constructor (Double, Double, Double)

[Missing <summary> documentation for
"M:is2GraphObject.ArcVectorDirection.#ctor(System.Double,System.Double,System.Doub

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcVectorDirection(  
    double x,  
    double y,  
    double z  
)
```

Parameters

x

Type: [System Double](#)

[Missing <param name="x"/> documentation for

"M:is2GraphObject.ArcVectorDirection.#ctor(System.Double,System.Double,System.Dc

y

Type: [System Double](#)

[Missing <param name="y"/> documentation for

"M:is2GraphObject.ArcVectorDirection.#ctor(System.Double,System.Double,System.Dc

z

Type: [System Double](#)

[Missing <param name="z"/> documentation for

"M:is2GraphObject.ArcVectorDirection.#ctor(System.Double,System.Double,System.Dc

▲ See Also

Reference

[ArcVectorDirection Structure](#)





[ArcVectorDirection Overload](#)

[is2GraphObject Namespace](#)

ArcVectorDirection Methods

The [ArcVectorDirection](#) type exposes the following members.

▲ Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ See Also


Reference

[ArcVectorDirection Structure](#)
[is2GraphObject Namespace](#)

ArcVectorDirection Properties

The [ArcVectorDirection](#) type exposes the following members.

▲ Properties

Name	Description
 val	

[Top](#)

▲ See Also

Reference

[ArcVectorDirection Structure](#)
[is2GraphObject Namespace](#)

ArcVectorDirectionval Property

[Missing <summary> documentation for "P:is2GraphObject.ArcVectorDirection.val"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType val { get; set; }
```

Property Value

Type: [PointType](#)

▲ See Also

Reference

[ArcVectorDirection Structure](#)

[is2GraphObject Namespace](#)

ChamferAngle Structure

[Missing <summary> documentation for "T:is2GraphObject.ChamferAngle"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax


C#

[Copy](#)

```
public struct ChamferAngle
```





The `ChamferAngle` type exposes the following members.

▲ Constructors

Name	Description
 ChamferAngle	

[Top](#)

▲ Methods

Name	Description
 Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
 GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
 GetType	Gets the Type of the current instance. (Inherited from Object .)
 ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ Properties

Name	Description
 val	

[Top](#)

▲ [See Also](#)

Reference

[is2GraphObject Namespace](#)

ChamferAngle Constructor

[Missing <summary> documentation for
"M:is2GraphObject.ChamferAngle.#ctor(System.Double)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ChamferAngle(  
    double angle  
)
```

Parameters

angle

Type: [SystemDouble](#)

[Missing <param name="angle"/> documentation for
"M:is2GraphObject.ChamferAngle.#ctor(System.Double)"]

▲ See Also

Reference





[ChamferAngle Structure](#)

[is2GraphObject Namespace](#)

ChamferAngle Methods

The [ChamferAngle](#) type exposes the following members.

Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

See Also

Reference


[ChamferAngle Structure](#)

[is2GraphObject Namespace](#)

ChamferAngle Properties

The [ChamferAngle](#) type exposes the following members.

▲ Properties

Name	Description
 val	

[Top](#)

▲ See Also

Reference

[ChamferAngle Structure](#)

[is2GraphObject Namespace](#)

ChamferAngleval Property

[Missing <summary> documentation for "P:is2GraphObject.ChamferAngle.val"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double val { get; set; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[ChamferAngle Structure](#)

[is2GraphObject Namespace](#)

ChamferDistante Structure

[Missing <summary> documentation for "T:is2GraphObject.ChamferDistante"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax


C#

[Copy](#)

```
public struct ChamferDistante
```





The `ChamferDistante` type exposes the following members.

▲ Constructors

Name	Description
 ChamferDistante	

[Top](#)

▲ Methods

Name	Description
 Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
 GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
 GetType	Gets the Type of the current instance. (Inherited from Object .)
 ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

▲ Properties

Name	Description
 val	

[Top](#)

▲ [See Also](#)

Reference

[is2GraphObject Namespace](#)

ChamferDistante Constructor

[Missing <summary> documentation for
"M:is2GraphObject.ChamferDistante.#ctor(System.Double)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ChamferDistante(  
    double dist  
)
```

Parameters

dist

Type: [SystemDouble](#)

[Missing <param name="dist"/> documentation for
"M:is2GraphObject.ChamferDistante.#ctor(System.Double)"]

▲ See Also

Reference





[ChamferDistante Structure](#)

[is2GraphObject Namespace](#)

ChamferDistante Methods

The [ChamferDistante](#) type exposes the following members.

Methods

	Name	Description
	Equals	Indicates whether this instance and a specified object are equal. (Inherited from ValueType .)
	GetHashCode	Returns the hash code for this instance. (Inherited from ValueType .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	ToString	Returns the fully qualified type name of this instance. (Inherited from ValueType .)

[Top](#)

See Also

Reference


[ChamferDistante Structure](#)

[is2GraphObject Namespace](#)

ChamferDistante Properties

The [ChamferDistante](#) type exposes the following members.

▲ Properties

Name	Description
 val	

[Top](#)

▲ See Also

Reference

[ChamferDistante Structure](#)

[is2GraphObject Namespace](#)

ChamferDistanteval Property

[Missing <summary> documentation for "P:is2GraphObject.ChamferDistante.val"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double val { get; set; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[ChamferDistante Structure](#)

[is2GraphObject Namespace](#)

CilindricalPointType Class

[Missing <summary> documentation for "T:is2GraphObject.CilindricalPointType"]

▲ Inheritance Hierarchy [System Object](#) [is2GraphObject](#)
[CilindricalPointType](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public class CilindricalPointType
```






The `CilindricalPointType` type exposes the following members.

▲ Constructors

Name	Description
 CilindricalPointType	

[Top](#)

▲ Methods

Name	Description
 Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
 Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
 GetHashCode	Serves as the default hash function. (Inherited from Object .)
 GetType	Gets the Type of the current instance. (Inherited from Object .)
 MemberwiseClone	Creates a shallow copy of the current Object .

(Inherited from [Object](#).)



[ToString](#)

Returns a string that represents the current object.
(Inherited from [Object](#).)

[Top](#)

▲ [See Also](#)

Reference

[is2GraphObject Namespace](#)

CilindricalPointType Constructor

[Missing <summary> documentation for "M:is2GraphObject.CilindricalPointType.#ctor"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public CilindricalPointType()
```

▲ See Also

Reference







[CilindricalPointType Class](#)

[is2GraphObject Namespace](#)

CilindricalPointType Methods

The [CilindricalPointType](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

[Top](#)

▲ See Also

Reference

[CylindricalPointType Class](#)

[is2GraphObject Namespace](#)

CircleCircleRelation Enumeration

Describe los tipos de relaciones relativas que ocurren entre dos circunferencias.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public enum CircleCircleRelation
```

▲ Members

Member name	Value	Description
Equal	0	Indica que dos circunferencias son iguales.
Concentric	1	Indica que dos circunferencias son concéntricas.
Interior	2	Indica que una de las circunferencias es interior a la otra.
Exterior	3	Indica que dos circunferencias son exteriores.
Tangent_In	4	Indica que una circunferencia es tangente interior a otra circunferencia.
Tangent_Out	5	Indica que una circunferencia es tangente exterior a otra circunferencia.
Secant	6	Indica que dos circunferencias son secantes.

▲ See Also

Reference

[is2GraphObject Namespace](#)

CircleException Class

Representa una exception para el tipo Circle Error.

▲ Inheritance Hierarchy [System Object System](#)

[Exception](#)

[is2GraphObject CircleException](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**



C#

[Copy](#)

```
public class CircleException : Exception
```





The `CircleException` type exposes the following members.

▲ Constructors





	Name	Description
	CircleException	Constructor por defecto.
	CircleException(String)	Constructor que toma como parámetro un tipo String.

[Top](#)

▲ Methods




	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash






function.
(Inherited from [Object](#).)

	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

[Top](#)


▲ Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)

	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

[Top](#)

▲ Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

[Top](#)



▲ See Also

Reference

is2GraphObject Namespace

CircleException Constructor

▲ Overload List

	Name	Description
	CircleException	Constructor por defecto.
	CircleException(String)	Constructor que toma como parámetro un tipo String.

[Top](#)

▲ See Also

Reference

[CircleException Class](#)

[is2GraphObject Namespace](#)

CircleException Constructor

Constructor por defecto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public CircleException()
```

▲ See Also

Reference

[CircleException Class](#)

[CircleException Overload](#)

[is2GraphObject Namespace](#)

CircleException Constructor (String)

Constructor que toma como parámetro un tipo String.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public CircleException(  
    string msg  
)
```

Parameters

msg

Type: [SystemString](#)

Representa una cadena de caracteres que indica la naturaleza de la exception.

▲ See Also

Reference

[CircleException Class](#)

[CircleException Overload](#)







[is2GraphObject Namespace](#)

[SystemString](#)

CircleException Methods

The [CircleException](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current

instance.
(Inherited from [Exception.](#))



[MemberwiseClone](#)

Creates a shallow copy of the current [Object](#).
(Inherited from [Object.](#))



[ToString](#)

Creates and returns a string representation of the current exception.
(Inherited from [Exception.](#))

[Top](#)

▲ See Also

Reference








[CircleException Class](#)

[is2GraphObject Namespace](#)

CircleException Properties

The [CircleException](#) type exposes the following members.

▲ Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the

immediate frames on the call stack.
(Inherited from [Exception](#).)



[TargetSite](#)

Gets the method that throws the current exception.
(Inherited from [Exception](#).)

[Top](#)

▲ See Also

Reference


[CircleException Class](#)

[is2GraphObject Namespace](#)

CircleException Events

The [CircleException](#) type exposes the following members.

▲ Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

[Top](#)

▲ See Also

Reference

[CircleException Class](#)

[is2GraphObject Namespace](#)

CircleSegmentRelation Enumeration

Describe los tipos de relaciones que ocurren entre una circunferencia y en segmento de recta.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public enum CircleSegmentRelation
```

▲ Members

Member name	Value	Description
Exterior	0	Indica que el segmento es exterior a la circunferencia.
Tangent	1	Indica que el segmento es tangente a la circunferencia.
Secant	2	Indica que el segmento es Secante a la circunferencia.
SimpleAcross	3	Indica que el segmento intercepta a la circunferencia en un solo punto pero sin ser tangente.

▲ See Also

Reference

[is2GraphObject Namespace](#)

CircleType Class

Representa un tipo Circunferencia.

▲ Inheritance Hierarchy [System Object](#) [is2GraphObject](#)
[CircleType](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax






C#

[Copy](#)

```
public class CircleType
```

The `CircleType` type exposes the following members.

Constructors

	Name	Description
	<code>CircleType</code>	Constructor por defecto.
	<code>CircleType(PointType, PointType)</code>	Constructor que toma como parámetros 2 puntos que pertenecen a la mayor cuerda (diámetro) de una circunferencia.
	<code>CircleType(PointType, Double)</code>	Constructor que toma como parámetros el punto centro y el radio de la circunferencia.
	<code>CircleType(SegmentType, SegmentType, SegmentType)</code>	Constructor que toma como parámetros 3 segmentos y crea una circunferencia que es tangente al mismo tiempo a los 3 segmentos.
	<code>CircleType(PointType, PointType, PointType, CircleType Type)</code>	Constructor que toma como parámetros 3 puntos, y crea una circunferencia que pasa por estos 3 puntos si el valor del parámetro "t" es - Circunscripta-. Por el contrario el valor de "t" es - Inscripta- los puntos dados sirven como vértices para el

cálculo del incentro del triángulo imaginario que estos forman, con lo que se obtiene una circunferencia inscrita.



[CircleType\(SegmentType, SegmentType, Double, Boolean, Boolean\)](#)

Constructor que toma como parámetros 2 segmentos y un valor de radio, y XXX

Top


Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object.

(Inherited from [Object](#).)





[Top](#)

▲ Fields

	Name	Description
	Center	Represente el punto centro de la circunferencia.

[Top](#)

▲ Properties

	Name	Description
	Area	Propiedad de solo lectura. Devuelve el área de la círculo que define la circunferencia.
	Diameter	Propiedad de solo lectura. Devuelve el diámetro de la circunferencia.
	Perimeter	Propiedad de solo lectura. Devuelve el perímetro de la circunferencia.
	Radius	Representa el radio de la circunferencia. Nota: El valor del radio se considera siempre positivo, por lo que establecer un valor de radio negativo no tiene ninguna influencia.

[Top](#)





▲ See Also

Reference

[is2GraphObject Namespace](#)

CircleType Constructor

▲ Overload List

	Name	Description
	CircleType	Constructor por defecto.
	CircleType(PointType, PointType)	Constructor que toma como parámetros 2 puntos que pertenecen a la mayor cuerda (diámetro) de una circunferencia.
	CircleType(PointType, Double)	Constructor que toma como parámetros el punto centro y el radio de la circunferencia.
	CircleType(SegmentType, SegmentType, SegmentType)	Constructor que toma como parámetros 3 segmentos y crea una circunferencia que es tangente al mismo tiempo a los 3 segmentos.
	CircleType(PointType, PointType, PointType, CircleType Type)	Constructor que toma como parámetros 3 puntos, y crea una circunferencia que pasa por estos 3 puntos si el valor del parámetro "t" es -Circunscripta-. Por el contrario el valor de "t" es -Inscripta- los puntos dados sirven como vértices para el cálculo del incentro del triángulo imaginario que

estos forman, con lo que se obtiene una circunferencia inscrita.



[CircleType\(SegmentType, SegmentType, Double, Boolean, Boolean\)](#)

Constructor que toma como parámetros 2 segmentos y un valor de radio, y XXX

[Top](#)

▲ See Also

Reference

[CircleType Class](#)

[is2GraphObject Namespace](#)

CircleType Constructor

Constructor por defecto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public CircleType()
```

▲ See Also

Reference

[CircleType Class](#)

[CircleType Overload](#)

[is2GraphObject Namespace](#)

CircleType Constructor (PointType, PointType)

Constructor que toma como parámetros 2 puntos que pertenecen a la mayor cuerda (diámetro) de una circunferencia.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public CircleType(  
    PointType P1,  
    PointType P2  
)
```

Parameters

P1

Type: [is2GraphObject PointType](#)
Primer punto.

P2

Type: [is2GraphObject PointType](#)
Segundo punto.

▲ See Also

Reference

[CircleType Class](#)

[CircleType Overload](#)

[is2GraphObject Namespace](#)

CircleType Constructor (PointType, Double)

Constructor que toma como parámetros el punto centro y el radio de la circunferencia.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public CircleType(  
    PointType P,  
    double radius  
)
```

Parameters

P

Type: [is2GraphObject PointType](#)
Punto centro de la circunferencia.

radius

Type: [System Double](#)
Radio de la circunferencia.

▲ See Also

Reference

[CircleType Class](#)

[CircleType Overload](#)

[is2GraphObject Namespace](#)

CircleType Constructor (SegmentType, SegmentType, SegmentType)

Constructor que toma como parámetros 3 segmentos y crea una circunferencia que es tangente al mismo tiempo a los 3 segmentos.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public CircleType(  
    SegmentType S1,  
    SegmentType S2,  
    SegmentType S3  
)
```

Parameters

S1

Type: [is2GraphObject SegmentType](#)
Primer segmento.

S2

Type: [is2GraphObject SegmentType](#)
Segundo segmento.

S3

Type: [is2GraphObject SegmentType](#)
Tercer segmento.

▲ See Also

Reference

[CircleType Class](#)

[CircleType Overload](#)

[is2GraphObject Namespace](#)

CircleType Constructor (PointType, PointType, PointType, CircleType Type)

Constructor que toma como parámetros 3 puntos, y crea una circunferencia que pasa por estos 3 puntos si el valor del parámetro "t" es -Circunscripta-. Por el contrario el valor de "t" es -Inscripta- los puntos dados sirven como vértices para el cálculo del incentro del triángulo imaginario que estos forman, con lo que se obtiene una circunferencia inscripta.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

Copy

```
public CircleType(  
    PointType P1,  
    PointType P2,  
    PointType P3,  
    CircleType Type t = CircleType Type.Circu  
)
```

Parameters

P1

Type: [is2GraphObject PointType](#)

Primer punto.

P2

Type: [is2GraphObject PointType](#)

Segundo punto.

P3

Type: [is2GraphObject PointType](#)

Tercer punto.

t (Optional)

Type: [is2GraphObject CircleType Type](#)

Indica de que forma respecto a P1, P2, P3 se crea la circunferencia. La cuál puede ser "inscripta" al triangulo imaginario que forman estos 3 puntos o "circunscripta" a este.

See Also

Reference

[CircleType Class](#)

[CircleType Overload](#)

[is2GraphObject Namespace](#)

CircleType Constructor (SegmentType, SegmentType, Double, Boolean, Boolean)

Constructor que toma como parámetros 2 segmentos y un valor de radio, y XXX

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public CircleType(  
    SegmentType S1,  
    SegmentType S2,  
    double radius,  
    bool s1_right_up = true,  
    bool s2_right_up = true  
)
```

Parameters

S1

Type: [is2GraphObject SegmentType](#)

Primer segmento.

S2

Type: [is2GraphObject SegmentType](#)

Segundo segmento.

radius

Type: [System Double](#)

Equidistancia a ambos segmentos.

Nota: El valor del radio se considera siempre positivo, por lo que establecer un valor de radio negativo no tiene ninguna influencia.

s1_right_up (Optional)

Type: [System Boolean](#)

Determina hacia que lado del segmento "S1" se calcula el circle.

Nota: Si el valor es **true**, el circle se calcula hacia la derecha-o-arriba del segmento. Por el contrario si el valor es **false**, el circle se calcula hacia la izquierda-o-abajo del segmento. En ambos casos su aplicación es sobre el segmento "S1".

s2_right_up (Optional)

Type: [System Boolean](#)

Determina hacia que lado del segmento "S2" se calcula el circle.

Nota: Si el valor es **true**, el circle se calcula hacia la derecha-o-arriba del segmento. Por el contrario si el valor es **false**, el circle

se calcula hacia la izquierda-o-abajo del segmento. En ambos casos su aplicación es sobre el segmento "S2".

▲ See Also

Reference

[CircleType Class](#)


[CircleType Overload](#)

[is2GraphObject Namespace](#)

CircleType Fields

The [CircleType](#) type exposes the following members.

▲ Fields

	Name	Description
	Center	Represente el punto centro de la circunferencia.

[Top](#)

▲ See Also

Reference

[CircleType Class](#)

[is2GraphObject Namespace](#)

CircleTypeCenter Field

Represente el punto centro de la circunferencia.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType Center
```

Field Value

Type: [PointType](#)

▲ See Also

Reference







[CircleType Class](#)

[is2GraphObject Namespace](#)

CircleType Methods

The [CircleType](#) type exposes the following members.

▲ Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

[Top](#)

▲ See Also

Reference





[CircleType Class](#)

[is2GraphObject Namespace](#)

CircleType Properties

The [CircleType](#) type exposes the following members.

▲ Properties

	Name	Description
	Area	Propiedad de solo lectura. Devuelve el área de la círculo que define la circunferencia.
	Diameter	Propiedad de solo lectura. Devuelve el diámetro de la circunferencia.
	Perimeter	Propiedad de solo lectura. Devuelve el perímetro de la circunferencia.
	Radius	Representa el radio de la circunferencia. Nota: El valor del radio se considera siempre positivo, por lo que establecer un valor de radio negativo no tiene ninguna influencia.

[Top](#)

▲ See Also

Reference

[CircleType Class](#)

[is2GraphObject Namespace](#)

CircleTypeArea Property

Propiedad de solo lectura. Devuelve el área de la círculo que define la circunferencia.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Area { get; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[CircleType Class](#)

[is2GraphObject Namespace](#)

CircleTypeDiameter Property

Propiedad de solo lectura. Devuelve el diámetro de la circunferencia.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Diameter { get; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[CircleType Class](#)

[is2GraphObject Namespace](#)

CircleTypePerimeter Property

Propiedad de solo lectura. Devuelve el perímetro de la circunferencia.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Perimeter { get; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[CircleType Class](#)

[is2GraphObject Namespace](#)

CircleType Radius Property

Representa el radio de la circunferencia.

Nota: El valor del radio se considera siempre positivo, por lo que establecer un valor de radio negativo no tiene ninguna influencia.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Radius { get; set; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[CircleType Class](#)

[is2GraphObject Namespace](#)

CircleTypeType Enumeration

Define la posicion relativa que ocupa la circunferencia con respecto aun polígono.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public enum Type
```

▲ Members

Member name	Value	Description
Circunscripta	1	Indica que la circunferencia es circunscripta a un polígono.
Inscripta	2	Indica que la circunferencia es inscripta a un polígono.

▲ See Also

Reference

[is2GraphObject Namespace](#)

Cuadrante Enumeration

Define cada una de las cuatro porciones en la que descompone un sistema de 2 planos interceptados en el espacio (Frontal, Horizontal).

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public enum Cuadrante
```

▲ Members

Member name	Value	Description
I	1	Representa el 1er cuadrante [+X +Y +Z] para un sistema de 2 planos.
II	2	Representa el 2do cuadrante [+X -Y +Z] para un sistema de 2 planos.
III	3	Representa el 3er cuadrante [+X -Y -Z] para un sistema de 2 planos.
IV	4	Representa el 4to cuadrante [+X +Y -Z] para un sistema de 2 planos.

▲ See Also

Reference

[is2GraphObject Namespace](#)

EllipseType Class

Representa un tipo Elipse.

▲ Inheritance Hierarchy [System.Object](#) [is2GraphObject](#)
[EllipseType](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax



C#

[Copy](#)

```
public class ElipseType
```





The `ElipseType` type exposes the following members.






Constructors

	Name	Description
	ElipseType	
	ElipseType(PointType, Double, Double)	
	ElipseType(PointType, Double, Double, Double)	

[Top](#)



Methods

	Name	Description
	Area	
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)

	GetType	Gets the Type of the current instance. (Inherited from Object .)
	isPointInside	
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Perimeter	
	ToString	Returns a string that represents the current object. (Inherited from Object .)

[Top](#)

▲ Fields

	Name	Description
	Center	Representa el centro de la elipse.
	SemiEjeA	Representa el valor del semi-eje horizontal.
	SemiEjeB	Representa el valor del semi-eje vertical.

[Top](#)

▲ See Also

Reference

[is2GraphObject Namespace](#)

EllipseType Constructor

▲ Overload List

	Name	Description
	EllipseType	
	EllipseType(PointType, Double, Double)	
	EllipseType(PointType, Double, Double, Double)	

[Top](#)

▲ See Also

Reference

[EllipseType Class](#)

[is2GraphObject Namespace](#)

EllipseType Constructor

[Missing <summary> documentation for "M:is2GraphObject.ElipseType.#ctor"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ElipseType()
```

▲ See Also

Reference

[ElipseType Class](#)

[ElipseType Overload](#)

[is2GraphObject Namespace](#)

EllipseType Constructor (PointType, Double, Double)

[Missing <summary> documentation for

"M:is2GraphObject.ElipseType.#ctor(is2GraphObject.PointType,System.Double,System.I

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public EllipseType(  
    PointType P,  
    double semiX,  
    double semiY  
)
```

Parameters

P

Type: [is2GraphObject PointType](#)

[Missing <param name="P"/> documentation for

"M:is2GraphObject.EllipseType.#ctor(is2GraphObject.PointType,System.Double,System.Double)"]

semiX

Type: [System Double](#)

[Missing <param name="semiX"/> documentation for

"M:is2GraphObject.EllipseType.#ctor(is2GraphObject.PointType,System.Double,System.Double)"]

semiY

Type: [System Double](#)

[Missing <param name="semiY"/> documentation for

"M:is2GraphObject.EllipseType.#ctor(is2GraphObject.PointType,System.Double,System.Double)"]

See Also

Reference

[EllipseType Class](#)

[EllipseType Overload](#)

[is2GraphObject Namespace](#)

EllipseType Constructor (PointType, Double, Double, Double)

[Missing <summary> documentation for
"M:is2GraphObject.ElipseType.#ctor(is2GraphObject.PointType,System.Double,System.I

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public EllipseType(  
    PointType P,  
    double semiX,  
    double semiY,  
    double angle  
)
```

Parameters

P

Type: [is2GraphObject PointType](#)

[Missing <param name="P"/> documentation for

"M:is2GraphObject.EllipseType.#ctor(is2GraphObject.PointType,System.Double,System

semiX

Type: [System Double](#)

[Missing <param name="semiX"/> documentation for

"M:is2GraphObject.EllipseType.#ctor(is2GraphObject.PointType,System.Double,System

semiY

Type: [System Double](#)

[Missing <param name="semiY"/> documentation for

"M:is2GraphObject.EllipseType.#ctor(is2GraphObject.PointType,System.Double,System

angle

Type: [System Double](#)

[Missing <param name="angle"/> documentation for

"M:is2GraphObject.EllipseType.#ctor(is2GraphObject.PointType,System.Double,System

▲ See Also

Reference

[EllipseType Class](#)




[EllipseType Overload](#)

[is2GraphObject Namespace](#)

EllipseType Fields

The [EllipseType](#) type exposes the following members.

▲ Fields

	Name	Description
	Center	Representa el centro de la elipse.
	SemiEjeA	Representa el valor del semi-eje horizontal.
	SemiEjeB	Representa el valor del semi-eje vertical.

[Top](#)

▲ See Also

Reference

[EllipseType Class](#)

[is2GraphObject Namespace](#)

EllipseTypeCenter Field

Representa el centro de la elipse.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType Center
```

Field Value

Type: [PointType](#)

▲ See Also

Reference

[EllipseType Class](#)

[is2GraphObject Namespace](#)

EllipseTypeSemiEjeA Field

Representa el valor del semi-eje horizontal.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double SemiEjeA
```

Field Value

Type: [Double](#)

▲ See Also

Reference

[EllipseType Class](#)

[is2GraphObject Namespace](#)

EllipseTypeSemiEjeB Field

Representa el valor del semi-eje vertical.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double SemiEjeB
```

Field Value

Type: [Double](#)

▲ See Also

Reference









[EllipseType Class](#)

[is2GraphObject Namespace](#)

EllipseType Methods

The [EllipseType](#) type exposes the following members.

Methods

	Name	Description
	Area	
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	isPointInside	
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Perimeter	



[ToString](#)

Returns a string that represents the current object.
(Inherited from [Object](#).)

[Top](#)

▲ See Also

Reference

[ElipseType Class](#)

[is2GraphObject Namespace](#)

EllipseTypeArea Method

[Missing <summary> documentation for "M:is2GraphObject.ElipseType.Area"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Area()
```

Return Value

Type: [Double](#)

[Missing <returns> documentation for "M:is2GraphObject.ElipseType.Area"]

▲ See Also

Reference

[EllipseType Class](#)

[is2GraphObject Namespace](#)

EllipseTypeisPointInside Method

[Missing <summary> documentation for
"M:is2GraphObject.ElipseType.isPointInside(is2GraphObject.PointType)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public bool isPointInside(  
    PointType P  
)
```

Parameters

P

Type: [is2GraphObjectPointType](#)

[Missing <param name="P"/> documentation for
"M:is2GraphObject.ElipseType.isPointInside(is2GraphObject.PointType)"]

Return Value

Type: [Boolean](#)

[Missing <returns> documentation for
"M:is2GraphObject.ElipseType.isPointInside(is2GraphObject.PointType)"]

▲ See Also

Reference

[ElipseType Class](#)

[is2GraphObject Namespace](#)

EllipseTypePerimeter Method

[Missing <summary> documentation for "M:is2GraphObject.ElipseType.Perimeter"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Perimeter()
```

Return Value

Type: [Double](#)

[Missing <returns> documentation for "M:is2GraphObject.ElipseType.Perimeter"]

▲ See Also

Reference

[EllipseType Class](#)

[is2GraphObject Namespace](#)

FilletException Class

Representa una exception para el tipo Fillet Error.

▲ Inheritance Hierarchy [System Object System](#)

[Exception](#)

[is2GraphObject](#) [FilletException](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**



C#

[Copy](#)

```
public class FilletException : Exception
```





The `FilletException` type exposes the following members.

▲ Constructors





	Name	Description
	FilletException	Constructor por defecto.
	FilletException(String)	Constructor que toma como parámetro un tipo String.

[Top](#)

▲ Methods




	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash






function.
(Inherited from [Object](#).)

	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

[Top](#)


▲ Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)

	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

[Top](#)

▲ Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

[Top](#)



▲ See Also

Reference

is2GraphObject Namespace

FilletException Constructor

▲ Overload List

	Name	Description
	FilletException	Constructor por defecto.
	FilletException(String)	Constructor que toma como parámetro un tipo String.

[Top](#)

▲ See Also

Reference

[FilletException Class](#)

[is2GraphObject Namespace](#)

FilletException Constructor

Constructor por defecto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public FilletException()
```

▲ See Also

Reference

[FilletException Class](#)

[FilletException Overload](#)

[is2GraphObject Namespace](#)

FilletException Constructor (String)

Constructor que toma como parámetro un tipo String.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#[Copy](#)

```
public FilletException(  
    string msg  
)
```

Parameters

msg

Type: [SystemString](#)

Representa una cadena de caracteres que indica la naturaleza de la exception.

▲ See Also

Reference

[FilletException Class](#)

[FilletException Overload](#)







[is2GraphObject Namespace](#)

[SystemString](#)

FilletException Methods

The [FilletException](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current

instance.
(Inherited from [Exception.](#))



[MemberwiseClone](#)

Creates a shallow copy of the current [Object](#).
(Inherited from [Object.](#))



[ToString](#)

Creates and returns a string representation of the current exception.
(Inherited from [Exception.](#))

[Top](#)

▲ See Also

Reference








[FilletException Class](#)

[is2GraphObject Namespace](#)

FilletException Properties

The [FilletException](#) type exposes the following members.

▲ Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the

immediate frames on the call stack.
(Inherited from [Exception](#).)



[TargetSite](#)

Gets the method that throws the current exception.
(Inherited from [Exception](#).)

[Top](#)

▲ See Also

Reference


[FilletException Class](#)

[is2GraphObject Namespace](#)

FilletException Events

The [FilletException](#) type exposes the following members.

▲ Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

[Top](#)

▲ See Also

Reference

[FilletException Class](#)

[is2GraphObject Namespace](#)

is2GraphObj Class

Biblioteca gráfica de entidades 2D.

▲ Inheritance Hierarchy [System.Object](#) [is2GraphObject](#)
[is2GraphObj](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax


C#

[Copy](#)

```
public class is2GraphObj
```





The `is2GraphObj` type exposes the following members.

▲ Constructors

Name	Description
 is2GraphObj	

[Top](#)

▲ Methods

Name	Description
  AngleInQuadrant	Determina el cuadrante en el que se ubica el valor de ángulo dado. Nota: el valor de ángulo esta expresado en grados.
  ArcArcFillet(ArcType, ArcType, Double, Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos arcos de circunferencias "A1" y "A2" dados, con radio determinado por "r_fillet".

ArcArcFillet(ArcType, ArcType, Double,
PointType , PointType , PointType ,
Boolean, Boolean, Boolean)

Calcula el fillet
(EMPALME)
entre dos arcos
de
circunferencias
"A1" y "A2"
dados, con
radio
determinado
por "r_fillet".



ArcArcIntersect

Calcula el o los
puntos de
intercepcion
entre dos arcos
de
circunferencias.



ArcCircleIntersect

Calcula el o los
puntos de
intercepcion
entre una
circunferencia y
un arco de
circunferencia.








ArcLineIntersect

Calcula el o los
puntos de
intercepcion
entre una línea
y un arco de
circunferencia.















ArcSegmentFillet(ArcType, SegmentType,
Double, Boolean, Boolean, Boolean)












Calcula el fillet
(EMPALME)
entre un arco
de
circunferencia
"A" y un

		segmento de recta "S" dados, con radio determinado por "r_fillet".
 	<code>ArcSegmentFillet(ArcType, SegmentType, Double, PointType , PointType , PointType , Boolean, Boolean, Boolean)</code>	Calcula el fillet (EMPALME) entre un arco de circunferencia "A" y un segmento de recta "S" dados, con radio determinado por "r_fillet".
 	<code>ArcSegmentIntersect</code>	Calcula el o los puntos de intercepcion entre un arco de circunferencia y un segmento de recta.
 	<code>CanonicEllipseCoefficient</code>	Determina los coeficientes de la ecuación canónica de la elipse ($Ax^2 + By^2 + E = 0$) para la elipse 'Elip' dada.
 	<code>CartesianToPolar</code>	Convierte













		coordenadas Cartesianas a coordenadas Polares.
	CheckPointLineRelativePosition	Determina si la posición relativa que tiene un punto "P" respecto a recta "L" cumple con la condición indicada por "condition".
	CilindricalPoint	Ubica un punto mediante coordenadas cilíndricas.
	CircleCircleFillet(CircleType, CircleType, Double, Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos circunferencias "C1" y "C2" dados, con radio determinado por "r_fillet".
	CircleCircleFillet(CircleType, CircleType, Double, PointType, PointType, PointType, Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos circunferencias "C1" y "C2" dados, con radio determinado

		por "r_fillet".
 	CircleCircleIntersect	Calcula el o los puntos de intercepción entre dos circunferencias.
 	CircleCircleRelationShip	Determina la relacion que hay entre 2 circunferencias.
 	CircleCoefficient	Determina los coeficientes de la ecuación general de la circunferencia $(x^2 + y^2 + Ax + By + C = 0)$ para la circunferencia 'Cir' dada.
 	CircleEllipseIntersect	
 	CircleLineIntersect	Calcula el o los puntos de intercepción entre una línea y un circunferencia.
 	CircleSegmentFillet(CircleType, SegmentType, Double, Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre una circunferencia "C" y un segmento de

		recta "S" dados, con radio determinado por "r_fillet".
	CircleSegmentFillet(CircleType, SegmentType, Double, PointType , PointType , PointType , Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre una circunferencia "C" y un segmento de recta "S" dados, con radio determinado por "r_fillet".
	CircleSegmentIntersect	Calcula el o los puntos de intercepcion entre una circunferencia y un segmento de recta.
	CircleSegmentRelationShip	Determina la relación que hay entre una circunfencias y un segmento.
	CircleTangentToLine	Determina la circunferencia que tiene su centro en "P" y para la cual la línea "L" dada es tangente.

 	ComplementaryAngle	Determina dado un angulo, el valor del angulo complementario
 	EllipseCoefficient	Determina los coeficientes de la ecuación general de la ellipse ($Ax^2 + By^2 + Cx + Dy + E = 0$) para la ellipse 'Elip' dada.
 	EllipseLineIntersect	
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
 	FarInXAxis	Determina cuál de los puntos "P1" y "P2" dados esta más alejado de "pbase" sobre el eje de las abcisas (Eje X).
 	FarInYAxis	Determina cuál de los puntos "P1" y "P2" dados esta más alejado de

		"pbase" sobre el eje de las ordenadas (Eje Y).
 	FarToOrigen	Determina cuál de los puntos "P1" y "P2" dados está más lejano del origen de coordenadas [0; 0].
 	FarToPoint(PointType, List PointType)	Determina de una lista de puntos definidas por "list" el punto más alejado del punto base "pbase".
 	FarToPoint(PointType, PointType, PointType)	Determina cuál de los dos puntos dados "P1" y "P2" está más alejado al punto "pbase".
 	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage

		collection. (Inherited from Object.)
	GetHashCode	Serves as the default hash function. (Inherited from Object.)
	GetType	Gets the Type of the current instance. (Inherited from Object.)
 	GradToRad	Convierte de Grados a Radianes.
 	isColinearSegment	Determina si dos segmetos "S1" y "S2" dados son colineales.
 	isEqualArc	Comprueba si dos arcos dados "A1" y "A2" son iguales.
 	isEqualCero	Determina si el parámetro 'value' se considera un cero real.
 	isEqualCircle	Comprueba si

dos
circunferencias
dadas "C1" y
"C2" son
iguales.



isEqualPoint

Comprueba si
dos puntos
dados "P1" y
"P2" son
iguales.



isEqualSegment

Comprueba si
dos Segmentos
dado "S1" y
"S2" son
iguales.



isEqualValues

Determina si
los dos valor
pasados por
parámetro son
iguales.



isNegative

Determina si el
valor pasado
por parámetro
es negativo.



isPositive

Determina si el
valor pasado
por parámetro
es positivo.



LineAccordingLineAt

Determina
una línea que
pasa por el
punto "P" y que
forma con la

		línea "L" dada, un ángulo determinado por el parámetro "angle".
 	<code>LineCoefficient(LineType, Double , Double , Double)</code>	Determina los coeficientes de la ecuacion general de la recta ($Ax + By + C = 0$) para la recta 'L' dada.
 	<code>LineCoefficient(SegmentType, Double , Double , Double)</code>	Determina los coeficientes de la ecuacion general de la recta ($Ax + By + C = 0$) para el segmento 'S' dado.
 	<code>LineLineAngle</code>	Determina el ángulo según su tipo (agudo u obtuso) que se forma entre dos líneas "L1" y "L2" dadas.
 	<code>LineLineIntersect</code>	Calcula el punto de intercepción entre dos líneas.
 	<code>LineTangentToCircle</code>	Determina las

rectas
tangentes a la
circunferencia
"C" dada y que
pasan por el
punto "P".



MakeMPQx

Resuelve un
sistema de
ecuaciones de
2 con 2 por el
método de
sustitución,
despejando 'X'
en la ecuación
lineal para
luego sustituirla
en la ecuación
cuadrática. De
lo que se
obtiene un
polinomio de la
forma: $mx^2 +$
 $px + q$.

Nota: Los
parámetros A,
B, C, A1, B1,
C1 se
interpretan de
la siguiente
forma:
- A, B, C:
coeficiente de
una ecuación
lineal de la
forma $Ax + By$
 $+ C = 0$ - A1,
B1, C1:













coeficientes de una ecuación cuadrática $X^2 + y^2 + A_1x + B_1y + C_1 = 0$



MakeMPQy













Resuelve un sistema de ecuaciones de 2 con 2 por el método de sustitución, despejando 'Y' en la ecuación lineal para luego sustituirla en la ecuación cuadrática. De lo que se obtiene un polinomio de la forma: $mx^2 + px + q$. **Nota:** Los parámetros A, B, C, A1, B1, C1 se interpretan de la siguiente forma:
- A, B, C: coeficiente de una ecuación lineal de la forma $Ax + By + C = 0$
- A1, B1, C1: coeficientes de una ecuación cuadrática X^2

$$+ y^2 + A_1x + B_1y + C_1 = 0$$















 	MaxXSegment	Determina el mayor valor de la coordenada X que contiene el segmento 'S' dado.
 	MaxYSegment	Determina el mayor valor de la coordenada Y que contiene el segmento 'S' dado.
 	MayorEstricto	Determina si el primer parámetro es estrictamente mayor que el segundo.
 	MayorOrEqual	Comprueba si el primer parámetro es mayor o igual que el segundo.
 	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
 	MenorEstricto	Determina si el primer




		parámetro es estrictamente menor que el segundo.
 	MenorOrEqual	Comprueba si el parámetro 'a' es menor o igual que 'b'.
 	MidPointBetweenPoint	Determina el punto medio o punto equidistante entre 2 puntos "P1" y "P2" dados.
 	MinXSegment	Determina el menor valor de la coordenada X que contiene el segmento 'S' dado.
 	MinYSegment	Determina el menor valor de la coordenada Y que contiene el segmento 'S' dado.
 	NearInXAxis	Determina cuál de los puntos "P1" y "P2" dados esta más cercano a "pbase" sobre el eje de las













		abcisas (Eje X).
 	NearInYAxis	Determina cuál de los puntos "P1" y "P2" dados esta más cercano a "pbase" sobre el eje de las ordenadas (Eje Y).
 	NearToOrigen	Determina cuál de los puntos "P1" y "P2" dados está más cercano al origen de coordenadas [0; 0].
 	NearToPoint(PointType, List PointType)	Determina que punto de una lista de puntos dada esta más cercano al punto "pbase".
 	NearToPoint(PointType, PointType, PointType)	Determina cuál de los puntos "P1" y "P2" dados está más cercano al punto "pbase".
 	NormalizeAngle	Normaliza un valor de ángulo.
 	OppositeAngle	Determina dado

		un ángulo, el valor del ángulo opuesto en el sistema de ejes de coordenadas.
 	ParallelLineAt	Determina una línea que es paralela a la línea "L" dada y que pasa por el punto "P".
 	PerpendicularLineAt	Determina una línea que es perpendicular a la línea "L" dada y que pasa por el punto "P".
 	PointInArc	Determina si el punto "P" dado pertenece al arco "A".
 	PointInCircle	Determina si el punto "P" dado pertenece al circunferencia "C".
 	PointInLine	Determina si el punto "P" dado pertenece la línea "L".
 	PointInOctant	Determina

		sobre que octante se ubica el punto "P" dado.
 	PointInQuadrant	Determina el cuadrante en el que se ubica el punto dado.
 	PointInSegment	Determina si el punto "P" dado pertenece al segmento "S".
 	PointLineDistance	Determina la distancia un punto "P" a una línea "L" dados.
 	PointPointAngle	Determina el valor del ángulo que se forma entre segmento que describen los puntos "P1" y "P2" dados y el eje de las abcisas (Eje X).
 	PointPointDistance	Calcula la distancia entre los puntos "P1" y "P2" dados.
 	PointPointSlope	Calcula la pendiente de la línea que pasa por los puntos

		"P1" y "P2" dados.
 	PolarPoint	Ubica un punto 2D mediante coordenadas polares partir de un punto base, un ángulo y una distancia.
 	PolarToCartesian	Convierte coordenadas Polares a coordenadas Cartesianas.
 	PolygonLineIntersect	
 	RadToGrad	Convierte de Radianes a Grados.
 	RectangleLineIntersect	
 	RootMPQ	Calcula las raíces de polinomios de 2do orden que tienen la forma: $mx^2 + px + q$ por el método del Discriminante.
 	RotateArc	Rota en el plano XY el arco 'A' sobre un punto

		'pbase' dado según un ángulo especificado.
 	RotateCircle	Rota en el plano XY el círculo 'C' sobre un punto 'pbase' dado según un ángulo especificado.
 	RotatePoint	Rota en el plano XY el punto 'P' sobre un punto 'pbase' dado según un ángulo especificado.
 	RotateSegment	Rota en el plano XY el segmento 'S' sobre un punto 'pbase' dado según un ángulo especificado.
 	SegmentsApparentIntersect	Determina la intercepción aparente en el plano entre dos segmentos. Nota: La intersección

		aparente indica
 	SegmentSegmentAngle	Determina el ángulo según su tipo (agudo u obtuso) que se forma entre dos segmentos "S1" y "S2" dadas.
 	SegmentSegmentChamfer(SegmentType, SegmentType, ChamferDistante, ChamferAngle, PointType , PointType)	
 	SegmentSegmentChamfer(SegmentType, SegmentType, ChamferDistante, ChamferDistante, PointType , PointType)	
 	SegmentSegmentFillet(SegmentType, SegmentType, Double, Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos segmentos de rectas "S1" y "S2" dados, con radio determinado por "r_fillet".
 	SegmentSegmentFillet(SegmentType, SegmentType, Double, PointType , PointType , PointType , Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos segmentos de rectas "S1" y "S2" dados, con radio determinado por "r_fillet".
 		

SegmentSegmentFilletSave



SegmentsRealIntersect

Determina la intercepción real en el plano entre dos segmentos.

Nota: La intersección real indica que ambos segmentos tienen realmente un punto en común.



SetPresicion

Establece la precision (posiciones decimales) que tiene en cuenta is2Graph para comprobar el valor del Cero Real.



SphericalPoint

Ubica un punto 3D mediante coordenadas esféricas.
















SwapArc

Intercambia los dos arcos pasados por parámetro.



SwapCircle

Intercambia las dos

		circunferencias pasados por parámetro.
 	SwapLine	Intercambia las dos líneas pasados por parámetro.
 	SwapPoint	Intercambia los dos puntos pasados por parámetro.
 	SwapSegment	Intercambia los dos segmentos pasados por parámetro.
 	SwapValue	Intercambia los dos valores pasados por parámetro.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
 	TranslateArc	Traslada el arco 'A' las distancias definidas por las componentes dx, dy, dz.
 	TranslateCircle	Traslada el

circulo 'C' las
distancias
definas por las
componentes
dx, dy, dz.



TranslatePoint

Traslada el
punto 'P' las
distancias
definas por las
componentes
dx, dy, dz.



TranslateSegment

Traslada el
segmento 'S'
las distancias
definas por las
componentes
dx, dy, dz.

[Top](#)

▲ Properties

	Name	Description
	OrigenXYZ	Propiedad de solo lectura. Devuelve el Origen de Sistema de Coordenadas.

[Top](#)

▲ **Remarks** is2GraphObj es la versión Orientada a Objetos de su antecesora is2Graph, re-escrita completamente para la tecnologia .Net de Microsoft. Representa la evolución de la libreria gráfica escrita por DrC. Ricardo Ávila Rondón llamada is2Graph (implementada inicialmente en c++ la cuál no tenia soporte para la orientacion a objetos). Is2GraphObj corrige un grupo de bugs que existían en su antecesora, asi mismo agrega nuevas entidades geometricas, muchas más características y más funcionalidades.

▲ See Also

Reference

[is2GraphObject Namespace](#)

is2GraphObj Constructor

[Missing <summary> documentation for "M:is2GraphObject.is2GraphObj.#ctor"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public is2GraphObj()
```

▲ See Also

Reference




[is2GraphObj Class](#)



[is2GraphObject Namespace](#)

is2GraphObj Methods

The [is2GraphObj](#) type exposes the following members.

Methods

	Name	Description
	AngleInQuadrant	Determina el cuadrante en el que se ubica el valor de ángulo dado. Nota: el valor de ángulo esta expresado en grados.
	ArcArcFillet(ArcType, ArcType, Double, Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos arcos de circunferencias "A1" y "A2" dados, con radio determinado por "r_fillet".
	ArcArcFillet(ArcType, ArcType, Double, PointType , PointType , PointType , Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos arcos de circunferencias "A1" y "A2" dados, con radio

		determinado por "r_fillet".
	ArcArcIntersect	Calcula el o los puntos de intercepcion entre dos arcos de circunferencias.
	ArcCircleIntersect	Calcula el o los puntos de intercepcion entre una circunferencia y un arco de circunferencia.
	ArcLineIntersect	Calcula el o los puntos de intercepcion entre una línea y un arco de circunferencia.
	ArcSegmentFillet(ArcType, SegmentType, Double, Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre un arco de circunferencia "A" y un segmento de recta "S" dados, con radio determinado por "r_fillet".
	ArcSegmentFillet(ArcType, SegmentType,	Calcula el fillet

Double, PointType , PointType ,
PointType , Boolean, Boolean, Boolean)

(EMPALME)
entre un arco
de
circunferencia
"A" y un
segmento de
recta "S"
dados, con
radio
determinado
por "r_fillet".



ArcSegmentIntersect

Calcula el o los
puntos de
intercepcion
entre un arco
de
circunferencia y
un segmento
de recta.



CanonicEllipseCoefficient

Determina los
coeficientes de
la ecuación
canónica de la
elipse ($Ax^2 + By^2 + E = 0$)
para la elipse
'Elip' dada.








CartesianToPolar











Convierte
coordenadas
Cartesianas a
coordenadas
Polares.







CheckPointLineRelativePosition

Determina si la
posición
relativa que

		tiene un punto "P" respecto a recta "L" cumple con la condición indicada por "condition".
	CilindricalPoint	Ubica un punto mediante coordenadas cilíndricas.
	CircleCircleFillet(CircleType, CircleType, Double, Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos circunferencias "C1" y "C2" dados, con radio determinado por "r_fillet".
	CircleCircleFillet(CircleType, CircleType, Double, PointType , PointType , PointType , Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos circunferencias "C1" y "C2" dados, con radio determinado por "r_fillet".
	CircleCircleIntersect	Calcula el o los puntos de intercepción entre dos circunferencias.
		

	CircleCircleRelationShip	Determina la relacion que hay entre 2 circunfencias.
 	CircleCoefficient	Determina los coeficientes de la ecuación general de la circunferencia $(x^2 + y^2 + Ax + By + C = 0)$ para la circunferencia 'Cir' dada.
 	CircleEllipseIntersect	
 	CircleLineIntersect	Calcula el o los puntos de intercepción entre una línea y un circunferencia.
 	CircleSegmentFillet(CircleType, SegmentType, Double, Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre una circunferencia "C" y un segmento de recta "S" dados, con radio determinado por "r_fillet".
 	CircleSegmentFillet(CircleType, SegmentType, Double, PointType ,	Calcula el fillet (EMPALME)

	PointType , PointType , Boolean, Boolean, Boolean)	entre una circunferencia "C" y un segmento de recta "S" dados, con radio determinado por "r_fillet".
	CircleSegmentIntersect	Calcula el o los puntos de intercepcion entre una circunferencia y un segmento de recta.
	CircleSegmentRelationShip	Determina la relación que hay entre una circunfencias y un segmento.
	CircleTangentToLine	Determina la circunferencia que tiene su centro en "P" y para la cual la línea "L" dada es tangente.
	ComplementaryAngle	Determina dado un angulo, el valor del angulo complementario
	ElipseCoefficient	Determina los coeficientes de

la ecuación general de la elipse ($Ax^2 + By^2 + Cx + Dy + E = 0$) para la elipse 'Elip' dada.



EllipseLineIntersect



Equals

Determines whether the specified object is equal to the current object. (Inherited from [Object](#).)



FarInXAxis

Determina cuál de los puntos "P1" y "P2" dados esta más alejado de "pbase" sobre el eje de las abcisas (Eje X).









FarInYAxis














Determina cuál de los puntos "P1" y "P2" dados esta más alejado de "pbase" sobre el eje de las ordenadas (Eje Y).

























FarToOrigen

Determina cuál de los puntos

		"P1" y "P2" dados está más lejano del origen de coordenadas [0; 0].
 	FarToPoint(PointType, List PointType)	Determina de una lista de puntos definidas por "list" el punto más alejado del punto base "pbase".
 	FarToPoint(PointType, PointType, PointType)	Determina cuál de los dos puntos dados "P1" y "P2" está más alejado al punto "pbase".
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function.

		(Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
 	GradToRad	Convierte de Grados a Radianes.
 	isColinearSegment	Determina si dos segmetos "S1" y "S2" dados son colineales.
 	isEqualArc	Comprueba si dos arcos dados "A1" y "A2" son iguales.
 	isEqualCero	Determina si el parámetro 'value' se considera un cero real.
 	isEqualCircle	Comprueba si dos circunferencias dadas "C1" y "C2" son iguales.
 	isEqualPoint	Comprueba si

		dos puntos dados "P1" y "P2" son iguales.
 	isEqualSegment	Comprueba si dos Segmentos dado "S1" y "S2" son iguales.
 	isEqualValues	Determina si los dos valor pasados por parámetro son iguales.
 	isNegative	Determina si el valor pasado por parámetro es negativo.
 	isPositive	Determina si el valor pasado por parámetro es positivo.
 	LineAccordingLineAt	Determina una línea que pasa por el punto "P" y que forma con la línea "L" dada, un ángulo determinado por el parámetro "angle".
 		

	LineCoefficient(LineType, Double , Double , Double)	Determina los coeficientes de la ecuacion general de la recta ($Ax + By + C = 0$) para la recta 'L' dada.
 	LineCoefficient(SegmentType, Double , Double , Double)	Determina los coeficientes de la ecuacion general de la recta ($Ax + By + C = 0$) para el segmento 'S' dado.
 	LineLineAngle	Determina el ángulo según su tipo (agudo u obtuso) que se forma entre dos líneas "L1" y "L2" dadas.
 	LineLineIntersect	Calcula el punto de intercepción entre dos líneas.
 	LineTangentToCircle	Determina las rectas tangentes a la circunferencia "C" dada y que pasan por el punto "P".
 		












Resuelve un sistema de ecuaciones de 2 con 2 por el método de sustitución, despejando 'X' en la ecuación lineal para luego sustituirla en la ecuación cuadrática. De lo que se obtiene un polinomio de la forma: $mx^2 + px + q$.

Nota: Los parámetros A, B, C, A1, B1, C1 se interpretan de la siguiente forma:
- A, B, C: coeficiente de una ecuación lineal de la forma $Ax + By + C = 0$
- A1, B1, C1: coeficientes de una ecuación cuadrática $X^2 + y^2 + A1x + B1y + C1 = 0$

sistema de ecuaciones de 2 con 2 por el método de sustitución, despejando 'Y' en la ecuación lineal para luego sustituirla en la ecuación cuadrática. De lo que se obtiene un polinomio de la forma: $mx^2 + px + q$. **Nota:** Los parámetros A, B, C, A1, B1, C1 se interpretan de la siguiente forma:

- A, B, C: coeficiente de una ecuación lineal de la forma $Ax + By + C = 0$
- A1, B1, C1: coeficientes de una ecuación cuadrática $X^2 + y^2 + A1x + B1y + C1 = 0$



		el segmento 'S' dado.
 	MaxYSegment	Determina el mayor valor de la coordenada Y que contiene el segmento 'S' dado.
 	MayorEstricto	Determina si el primer parámetro es estrictamente mayor que el segundo.
 	MayorOrEqual	Comprueba si el primer parámetro es mayor o igual que el segundo.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
 	MenorEstricto	Determina si el primer parámetro es estrictamente menor que el segundo.
 	MenorOrEqual	Comprueba si el parámetro 'a'

es menor o igual que 'b'.



MidPointBetweenPoint

Determina el punto medio o punto equidistante entre 2 puntos "P1" y "P2" dados.



MinXSegment

Determina el menor valor de la coordenada X que contiene el segmento 'S' dado.



MinYSegment

Determina el menor valor de la coordenada Y que contiene el segmento 'S' dado.















NearInXAxis




Determina cuál de los puntos "P1" y "P2" dados esta más cercano a "pbase" sobre el eje de las abcisas (Eje X).




























NearInYAxis









Determina cuál de los puntos "P1" y "P2" dados esta más cercano a







		"pbase" sobre el eje de las ordenadas (Eje Y).
 	NearToOrigen	Determina cuál de los puntos "P1" y "P2" dados está más cercano al origen de coordenadas [0; 0].
 	NearToPoint(PointType, List PointType)	Determina que punto de una lista de puntos dada esta más cercano al punto "pbase".
 	NearToPoint(PointType, PointType, PointType)	Determina cuál de los puntos "P1" y "P2" dados está más cercano al punto "pbase".
 	NormalizeAngle	Normaliza un valor de ángulo.
 	OppositeAngle	Determina dado un ángulo, el valor del ángulo opuesto en el sistema de ejes de coordenadas.
 		

	ParallelLineAt	Determina una línea que es paralela a la línea "L" dada y que pasa por el punto "P".
 	PerpendicularLineAt	Determina una línea que es perpendicular a la línea "L" dada y que pasa por el punto "P".
 	PointInArc	Determina si el punto "P" dado pertenece al arco "A".
 	PointInCircle	Determina si el punto "P" dado pertenece al circunferencia "C".
 	PointInLine	Determina si el punto "P" dado pertenece la línea "L".
 	PointInOctant	Determina sobre que octante se ubica el punto "P" dado.
 	PointInQuadrant	Determina el cuadrante en el

		que se ubica el punto dado.
 	PointInSegment	Determina si el punto "P" dado pertenece al segmento "S".
 	PointLineDistance	Determina la distancia un punto "P" a una línea "L" dados.
 	PointPointAngle	Determina el valor del ángulo que se forma entre segmento que describen los puntos "P1" y "P2" dados y el eje de las abscisas (Eje X).
 	PointPointDistance	Calcula la distancia entre los puntos "P1" y "P2" dados.
 	PointPointSlope	Calcula la pendiente de la línea que pasa por los puntos "P1" y "P2" dados.
 	PolarPoint	Ubica un punto 2D mediante coordenadas polares partir

		de un punto base, un ángulo y una distancia.
 	PolarToCartesian	Convierte coordenadas Polares a coordenadas Cartesianas.
 	PolygonLineIntersect	
 	RadToGrad	Convierte de Radianes a Grados.
 	RectangleLineIntersect	
 	RootMPQ	Calcula las raíces de polinomios de 2do orden que tienen la forma: $mx^2 + px + q$ por el método del Discriminante.
 	RotateArc	Rota en el plano XY el arco 'A' sobre un punto 'pbase' dado según un ángulo especificado.
 	RotateCircle	Rota en el plano XY el

		<p>circulo 'C' sobre un punto 'pbase' dado según un ángulo especificado.</p>
 	RotatePoint	<p>Rota en el plano XY el punto 'P' sobre un punto 'pbase' dado según un ángulo especificado.</p>
 	RotateSegment	<p>Rota en el plano XY el segmento 'S' sobre un punto 'pbase' dado según un ángulo especificado.</p>
 	SegmentsApparentIntersect	<p>Determina la intercepción aparente en el plano entre dos segmentos. Nota: La intersección aparente indica</p>
 	SegmentSegmentAngle	<p>Determina el ángulo según su tipo (agudo u obtuso) que se forma entre</p>

		dos segmentos "S1" y "S2" dadas.
	SegmentSegmentChamfer(SegmentType, SegmentType, ChamferDistante, ChamferAngle, PointType , PointType)	
	SegmentSegmentChamfer(SegmentType, SegmentType, ChamferDistante, ChamferDistante, PointType , PointType)	
	SegmentSegmentFillet(SegmentType, SegmentType, Double, Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos segmentos de rectas "S1" y "S2" dados, con radio determinado por "r_fillet".
	SegmentSegmentFillet(SegmentType, SegmentType, Double, PointType , PointType , PointType , Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos segmentos de rectas "S1" y "S2" dados, con radio determinado por "r_fillet".
	SegmentSegmentFilletSave	
	SegmentsRealIntersect	Determina la intercepción real en el plano entre dos segmentos.

Nota: La intersección real indica que ambos segmentos tienen realmente un punto en común.



SetPresicion

Establece la precision (posiciones decimales) que tiene en cuenta is2Graph para comprobar el valor del Cero Real.



SphericalPoint

Ubica un punto 3D mediante coordenadas esféricas.



SwapArc

Intercambia los dos arcos pasados por parámetro.










SwapCircle

Intercambia las dos circunferencias pasados por parámetro.



SwapLine

Intercambia las dos líneas pasados por

		parámetro.
	SwapPoint	Intercambia los dos puntos pasados por parámetro.
	SwapSegment	Intercambia los dos segmentos pasados por parámetro.
	SwapValue	Intercambia los dos valores pasados por parámetro.
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TranslateArc	Traslada el arco 'A' las distancias definidas por las componentes dx, dy, dz.
	TranslateCircle	Traslada el círculo 'C' las distancias definidas por las componentes dx, dy, dz.
	TranslatePoint	Traslada el

punto 'P' las
distancias
definas por las
componentes
dx, dy, dz.



TranslateSegment

Traslada el
segmento 'S'
las distancias
definas por las
componentes
dx, dy, dz.

[Top](#)

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj AngleInQuadrant Method

Determina el cuadrante en el que se ubica el valor de ángulo dado.
Nota: el valor de ángulo esta expresado en grados.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static Cuadrante AngleInQuadrant(  
    double angle  
)
```

Parameters

angle

Type: [System Double](#)

Valor de ángulo en grados.

Return Value

Type: [Cuadrante](#)

Devuelve un tipo enum que representa el cuadrante en que se ubica el ángulo dado.

▲ See Also



Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj ArcArcFillet Method

▲ Overload List

	Name	Description
	ArcArcFillet(ArcType, ArcType, Double, Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos arcos de circunferencias "A1" y "A2" dados, con radio determinado por "r_fillet".
	ArcArcFillet(ArcType, ArcType, Double, PointType , PointType , PointType , Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos arcos de circunferencias "A1" y "A2" dados, con radio determinado por "r_fillet".

[Top](#)

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj ArcArcFillet Method (ArcType, ArcType, Double, Boolean, Boolean, Boolean)

Calcula el fillet (EMPALME) entre dos arcos de circunferencias "A1" y "A2" dados, con radio determinado por "r_fillet".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public static ArcType ArcArcFillet(  
    ArcType A1,  
    ArcType A2,  
    double r_fillet,  
    bool right_up = true,  
    bool a1_outside = true,  
    bool a2_outside = true  
)
```

Parameters

A1

Type: [is2GraphObject ArcType](#)

Representa el 1er arco de circunferencia.

A2

Type: [is2GraphObject ArcType](#)

Representa el 2do arco de circunferencia.

r_fillet

Type: [System Double](#)

Representa el radio del fillet.

right_up (**Optional**)

Type: [System Boolean](#)

Indica hacia que lado se calcula el fillet, tomando como referencia el segmento imaginario "Sx" que se forma entre los centros de ambos arcos.

Nota: Si el valor es **true**, el fillet se calcula hacia la derecha-o-arriba del segmento "Sx". Por el contrario si el valor es **false**, el fillet se calcula hacia la izquierda-o-abajo del segmento.

a1_outside (**Optional**)

Type: [System Boolean](#)

Indica si el fillet se calcula hacia adentro o hacia afuera de la circunferencia imaginaria que forma el arco "A1".

Nota: Si el valor es **true**, el fillet se calcula hacia el exterior de la circunferencia. Por el contrario si el valor es **false**, el fillet se calcula hacia el interior de la circunferencia. En ambos casos su

aplicación es sobre la circunferencia "C1".

a2_outside (Optional)

Type: [System Boolean](#)

Indica si el fillet se calcula hacia adentro o hacia afuera de la circunferencia imaginaria que forma el arco "A2".

Nota: Si el valor es **true**, el fillet se calcula hacia el exterior de la circunferencia. Por el contrario si el valor es **false**, el fillet se calcula hacia el interior de la circunferencia. En ambos casos su aplicación es sobre la circunferencia "C2".

Return Value

Type: [ArcType](#)

Devuelve un tipo Arco que representa el fillet calculado los dos arcos de circunferencias dados.

▲ See Also

Reference

[is2GraphObj Class](#)

[ArcArcFillet Overload](#)

[is2GraphObject Namespace](#)

is2GraphObj ArcArcFillet Method (ArcType, ArcType, Double, PointType , PointType , PointType , Boolean, Boolean, Boolean)

Calcula el fillet (EMPALME) entre dos arcos de circunferencias "A1" y "A2" dados, con radio determinado por "r_fillet".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static void ArcArcFillet(  
    ArcType A1,  
    ArcType A2,  
    double r_fillet,  
    out PointType P1,  
    out PointType P2,  
    out PointType Pc,  
    bool right_up = true,  
    bool a1_outside = true,  
    bool a2_outside = true  
)
```

Parameters

A1

Type: [is2GraphObject ArcType](#)

Representa el 1er arco de circunferencia.

A2

Type: [is2GraphObject ArcType](#)

Representa el 2do arco de circunferencia.

r_fillet

Type: [System Double](#)

Representa el radio del fillet.

P1

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto del arc-fillet que pertenece a la circunferencia.

P2

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto del arc-fillet que pertenece al segmento de recta.

Pc

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto centro del arc-fillet.

right_up (Optional)

Type: [System Boolean](#)

Indica hacia que lado se calcula el fillet, tomando como referencia el segmento imaginario "Sx" que se forma entre los centros de ambos arcos.

Nota: Si el valor es **true**, el fillet se calcula hacia la derecha-o-arriba del segmento "Sx". Por el contrario si el valor es **false**, el fillet se calcula hacia la izquierda-o-abajo del segmento.

a1_outside (Optional)

Type: [System Boolean](#)

Indica si el fillet se calcula hacia adentro o hacia afuera de la circunferencia imaginaria que forma el arco "A1".

Nota: Si el valor es **true**, el fillet se calcula hacia el exterior de la circunferencia. Por el contrario si el valor es **false**, el fillet se calcula hacia el interior de la circunferencia. En ambos casos su aplicación es sobre la circunferencia "C1".

a2_outside (Optional)

Type: [System Boolean](#)

Indica si el fillet se calcula hacia adentro o hacia afuera de la circunferencia imaginaria que forma el arco "A2".

Nota: Si el valor es **true**, el fillet se calcula hacia el exterior de la circunferencia. Por el contrario si el valor es **false**, el fillet se calcula hacia el interior de la circunferencia. En ambos casos su aplicación es sobre la circunferencia "C2".

Return Value

Type:

Devuelve un tipo void. Por referencia mediante los parámetros P1, P2 y Pc se obtienen los 3 puntos del arc-fillet.

▲ **Remarks** Nota: "P1" pertenece al arco "A1", "P2" pertenece al arco "A2" y "Pc" es el centro del arco.

▲ **See Also**

Reference

[is2GraphObj Class](#)

[ArcArcFillet Overload](#)

is2GraphObject Namespace
is2GraphObject ArcType
is2GraphObject PointType

is2GraphObj ArcArcIntersect Method

Calcula el o los puntos de intercepcion entre dos arcos de circunferencias.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static byte ArcArcIntersect(  
    ArcType A1,  
    ArcType A2,  
    out PointType P1,  
    out PointType P2  
)
```

Parameters

A1

Type: [is2GraphObject ArcType](#)
Primer arco.

A2

Type: [is2GraphObject ArcType](#)
Segundo arco.

P1

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia uno de los puntos de intersección, si existe.

P2

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia el otro punto de intersección, si existe.

Return Value

Type: [Byte](#)

Devuelve un valor entero corto que indica lo siguiente:

- 0 - Los arcos no intersectan entre si.
- 1 - Los arcos se intersectan en un único punto.
- 2 - Los arcos se intersectan en dos puntos.

- ▲ **Remarks** - Si los arcos de circunferencia no se interceptan, entonces "P1" y "P2" retornarán NULL.
- Si los arcos de circunferencia se interceptan en un solo punto, entonces "P1" y "P2" son iguales.

- Si los arcos de circunferencia se interceptan en dos puntos, entonces "P1" y "P2" son distintos.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject ArcType](#)

[is2GraphObject PointType](#)

is2GraphObj ArcCircleIntersect Method

Calcula el o los puntos de intercepcion entre una circunferencia y un arco de circunferencia.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static byte ArcCircleIntersect(  
    ArcType A,  
    CircleType C,  
    out PointType P1,  
    out PointType P2  
)
```

Parameters

A

Type: [is2GraphObject ArcType](#)
Representa el arco de circunferencia.

C

Type: [is2GraphObject CircleType](#)
Representa la circunferencia.

P1

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia uno de los puntos de intersección, si existe.

P2

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia el otro punto de intersección, si existe.

Return Value

Type: [Byte](#)

Devuelve un valor entero corto que indica lo siguiente:

- 0 - El arco y la circunferencia no se intersectan en ningún punto.
- 1 - El arco y la circunferencia se intersectan en un único punto.
- 2 - El arco y la circunferencia se intersectan en dos puntos.

- ▲ **Remarks** - Si el arco y la circunferencia no se interceptan, entonces "P1" y "P2" retornarán NULL.
- Si el arco y la circunferencia se interceptan en un solo punto, entonces "P1" y "P2" son iguales.

- Si el arco y la circunferencia se interceptan en dos puntos, entonces "P1" y "P2" son distintos.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject ArcType](#)

[is2GraphObject CircleType](#)

[is2GraphObject PointType](#)

is2GraphObj ArcLineIntersect Method

Calcula el o los puntos de intercepcion entre una línea y un arco de circunferencia.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static byte ArcLineIntersect(  
    ArcType A,  
    LineType L,  
    out PointType P1,  
    out PointType P2  
)
```

Parameters

A

Type: [is2GraphObject ArcType](#)
Representa el arco de circunferencia.

L

Type: [is2GraphObject LineType](#)
Representa la línea.

P1

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia uno de los puntos de intersección, si existe.

P2

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia el otro punto de intersección, si existe.

Return Value

Type: [Byte](#)

Devuelve un valor entero corto que indica lo siguiente:

- 0 - La línea no intersecta al arco.
- 1 - La línea intersecta al arco en un único punto.
- 2 - La línea intersecta al arco en dos puntos.

- ▲ **Remarks** - Si la línea no intersecta al arco, entonces "P1" y "P2" retornarán NULL.
- Si la línea intersecta al arco en un solo punto, entonces "P1" y "P2" son iguales.

- Si la línea intersecta al arco en dos puntos, entonces "P1" y "P2" son distintos.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)



[is2GraphObject ArcType](#)

[is2GraphObject LineType](#)

[is2GraphObject PointType](#)

is2GraphObj ArcSegmentFillet Method

▲ Overload List

	Name	Description
	ArcSegmentFillet(ArcType, SegmentType, Double, Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre un arco de circunferencia "A" y un segmento de recta "S" dados, con radio determinado por "r_fillet".
	ArcSegmentFillet(ArcType, SegmentType, Double, PointType, PointType, PointType, Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre un arco de circunferencia "A" y un segmento de recta "S" dados, con radio determinado por "r_fillet".

[Top](#)

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj ArcSegmentFillet Method (ArcType, SegmentType, Double, Boolean, Boolean, Boolean)

Calcula el fillet (EMPALME) entre un arco de circunferencia "A" y un segmento de recta "S" dados, con radio determinado por "r_fillet".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static ArcType ArcSegmentFillet(  
    ArcType A,  
    SegmentType S,  
    double r_fillet,  
    bool right = true,  
    bool outside = true,  
    bool up = true  
)
```

Parameters

A

Type: [is2GraphObject ArcType](#)

Representa el arco de circunferencia.

S

Type: [is2GraphObject SegmentType](#)

Representa el segmento de recta.

r_fillet

Type: [System Double](#)

Representa el radio del fillet.

right (**Optional**)

Type: [System Boolean](#)

Indica hacia que lado del segmento "S" dado se calcula el fillet (empalme). Los lados quedan determinados por la recta que parte del centro del arco "A" y es perpendicular al segmento "S". Si el valor es **true** el fillet se calcula hacia la derecha del segmento "S", por el contrario si el valor es **false** el fillet se calcula hacia la izquierda de dicho segmento.

outside (**Optional**)

Type: [System Boolean](#)

Indica si el fillet (empalme) se calcula exterior o interior a la circunferencia imaginaria a la pertenece el arco "A" dado. Si el valor es **true** el fillet se calcula exterior a la circunferencia, por el contrario si el valor es **false** el fillet se calcula hacia el interior de la circunferencia imaginaria.

up (Optional)

Type: [System Boolean](#)

Indica si el fillet se calcula sobre el segmento "S" dado o por debajo de él. Si el valor es **true** el fillet se calcula sobre el segmento "S" dado, por el contrario si el valor es **false** el fillet se calcula por debajo de dicho segmento.

Return Value

Type: [ArcType](#)

Devuelve un tipo Arco que representa el fillet calculado para el arco de circunferencia y el segmento de recta dados.

▲ See Also

Reference

[is2GraphObj Class](#)

[ArcSegmentFillet Overload](#)

[is2GraphObject Namespace](#)

is2GraphObj ArcSegmentFillet Method (ArcType, SegmentType, Double, PointType , PointType , PointType , Boolean, Boolean, Boolean)

Calcula el fillet (EMPALME) entre un arco de circunferencia "A" y un segmento de recta "S" dados, con radio determinado por "r_fillet".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static void ArcSegmentFillet(  
    ArcType A,  
    SegmentType S,  
    double r_fillet,  
    out PointType P1,  
    out PointType P2,  
    out PointType Pc,  
    bool right = true,  
    bool outside = true,  
    bool up = true  
)
```

Parameters

A

Type: [is2GraphObject ArcType](#)

Representa el arco de circunferencia.

S

Type: [is2GraphObject SegmentType](#)

Representa el segmento de recta.

r_fillet

Type: [System Double](#)

Representa el radio del fillet.

P1

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto del arc-fillet que pertenece a la circunferencia.

P2

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto del arc-fillet que pertenece al segmento de recta.

Pc

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto centro del arc-fillet.

right (Optional)

Type: [System Boolean](#)

Indica hacia que lado del segmento "S" dado se calcula el fillet (empalme). Los lados quedan determinados por la recta que parte del centro del arco "A" y es perpendicular al segmento "S". Si el valor es **true** el fillet se calcula hacia la derecha del segmento "S", por el contrario si el valor es **false** el fillet se calcula hacia la izquierda de dicho segmento.

outside (Optional)

Type: [System Boolean](#)

Indica si el fillet (empalme) se calcula exterior o interior a la circunferencia imaginaria a la pertenece el arco "A" dado. Si el valor es **true** el fillet se calcula exterior a la circunferencia, por el contrario si el valor es **false** el fillet se calcula hacia el interior de la circunferencia imaginaria.

up (Optional)

Type: [System Boolean](#)

Indica si el fillet se calcula sobre el segmento "S" dado o por debajo de él. Si el valor es **true** el fillet se calcula sobre el segmento "S" dado, por el contrario si el valor es **false** el fillet se calcula por debajo de dicho segmento.

Return Value

Type:

Devuelve un tipo void. Por referencia mediante los parámetros P1, P2 y Pc se obtienen los 3 puntos del arc-fillet.

▲ **Remarks** Nota: "P1" pertenece al arco, "P2" pertenece al segmento de recta y "Pc" es el centro del arco.

▲ See Also

Reference

[is2GraphObj Class](#)

[ArcSegmentFillet Overload](#)

[is2GraphObject Namespace](#)

[is2GraphObject ArcType](#)

[is2GraphObject SegmentType](#)

is2GraphObject PointType

is2GraphObj

ArcSegmentIntersect Method

Calcula el o los puntos de intercepcion entre un arco de circunferencia y un segmento de recta.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static int ArcSegmentIntersect(  
    ArcType A,  
    SegmentType S,  
    out PointType P1,  
    out PointType P2  
)
```

Parameters

A

Type: [is2GraphObject ArcType](#)
Representa el arco de circunferencia.

S

Type: [is2GraphObject SegmentType](#)
Representa el segmento del recta.

P1

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia uno de los puntos de intersección, si existe.

P2

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia el otro punto de intersección, si existe.

Return Value

Type: [Int32](#)

Devuelve un valor entero corto que indica lo siguiente:

- 0 - El segmento no intersecta al arco.
- 1 - El segmento intersecta al arco en un único punto.
- 2 - El segmento intersecta al arco en dos puntos.

- ▲ **Remarks** - Si el segmento no intercepta al arco de circunferencia, entonces "P1" y "P2" retornarán NULL.
- Si el segmento intercepta al arco de circunferencia en un punto, entonces "P1" y "P2" son iguales.

- Si el segmento intercepta al arco de circunferencia en dos puntos, entonces "P1" y "P2" son distintos.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject ArcType](#)

[is2GraphObject SegmentType](#)

[is2GraphObject PointType](#)

is2GraphObj

CanonicEllipseCoefficient Method

Determina los coeficientes de la ecuación canónica de la elipse ($Ax^2 + By^2 + E = 0$) para la elipse 'Elip' dada.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public static void CanonicEllipseCoefficient(  
    EllipseType Elip,  
    out double A,  
    out double B,  
    out double C  
)
```

Parameters

Elip

Type: [is2GraphObject EllipseType](#)

[Missing <param name="Elip"/> documentation for

"M:is2GraphObject.is2GraphObj.CanonicEllipseCoefficient(is2GraphObject.EllipseType,

A

Type: [System Double](#)

[Missing <param name="A"/> documentation for

"M:is2GraphObject.is2GraphObj.CanonicEllipseCoefficient(is2GraphObject.EllipseType,

B

Type: [System Double](#)

[Missing <param name="B"/> documentation for

"M:is2GraphObject.is2GraphObj.CanonicEllipseCoefficient(is2GraphObject.EllipseType,

C

Type: [System Double](#)

[Missing <param name="C"/> documentation for

"M:is2GraphObject.is2GraphObj.CanonicEllipseCoefficient(is2GraphObject.EllipseType,

Return Value

Type:

Devuelve un tipo void. Por referencia mediante los parámetros A, B, E se obtienen los coeficientes de la ecuación.

▲ See Also

Reference

[is2GraphObj Class](#)

is2GraphObj CartesianToPolar Method

Convierte coordenadas Cartesianas a coordenadas Polares.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

Copy

```
public static void CartesianToPolar(  
    double dX,  
    double dY,  
    out double angle,  
    out double dist,  
    PointType pbase = null  
)
```

Parameters

dX

Type: [System Double](#)

Coordena X en el sistema cartesiano. Tambien se interpreta como desplazamiento a partir del punto "pbase" sobre el eje de las abscisas (Eje X).

dY

Type: [System Double](#)

Coordena Y en el sistema cartesiano. Tambien se interpreta como desplazamiento a partir del punto "pbase" sobre el eje de las ordenadas (Eje Y).

angle

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia la componente angular del sistema de coordenadas polares.

dist

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia la componente distacia del sistema de coordenadas polares.

pbase (Optional)

Type: [is2GraphObject PointType](#)

Define el punto base que se usa como origen para calcular las componentes angular y distancia. Si "pbase" es igual a null se asume como punto base el Origen del Sistema de Coordenadas (0,0,0).

Return Value

Type:

Devuelve un tipo void. Por referencia se devuelve las componentes angular y distancia del Sistema de Coordenadas Polares.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj

CheckPointLineRelativePosition

Method

Determina si la posición relativa que tiene un punto "P" respecto a recta "L" cumple con la condición indicada por "condition".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool CheckPointLineRelativePositi  
    PointType P,  
    LineType L,  
    PointLinePosition condition  
)
```

Parameters

P

Type: [is2GraphObject PointType](#)
Representa el punto.

L

Type: [is2GraphObject LineType](#)
Representa la linea.

condition

Type: [is2GraphObject PointLinePosition](#)

[Missing <param name="condition"/> documentation for

"M:is2GraphObject.is2GraphObj.CheckPointLineRelativePosition(is2GraphObject.Poi

Return Value

Type: [Boolean](#)

Devuelve **true** si se cumple la condición dada, en caso contrario

devuelve **false** ▲ [See Also](#)

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObjCilindricalPoint Method

Ubica un punto mediante coordenadas cilíndricas.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public static PointType CilindricalPoint(  
    PointType pbase  
)
```

Parameters

pbase

Type: [is2GraphObjectPointType](#)

[Missing <param name="pbase"/> documentation for
"M:is2GraphObject.is2GraphObj.CilindricalPoint(is2GraphObject.PointType)"]

Return Value

Type: [PointType](#)

[Missing <returns> documentation for
"M:is2GraphObject.is2GraphObj.CilindricalPoint(is2GraphObject.PointType)"]

▲ See Also

Reference



[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObjectPointType](#)

is2GraphObj CircleCircleFillet Method

▲ Overload List

	Name	Description
	CircleCircleFillet(CircleType, CircleType, Double, Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos circunferencias "C1" y "C2" dados, con radio determinado por "r_fillet".
	CircleCircleFillet(CircleType, CircleType, Double, PointType , PointType , PointType , Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos circunferencias "C1" y "C2" dados, con radio determinado por "r_fillet".

[Top](#)

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj CircleCircleFillet Method (CircleType, CircleType, Double, Boolean, Boolean, Boolean)

Calcula el fillet (EMPALME) entre dos circunferencias "C1" y "C2" dados, con radio determinado por "r_fillet".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static ArcType CircleCircleFillet(  
    CircleType C1,  
    CircleType C2,  
    double r_fillet,  
    bool right_up = true,  
    bool c1_outside = true,  
    bool c2_outside = true  
)
```

Parameters

C1

Type: [is2GraphObject CircleType](#)
Representa la 1ra circunferencia.

C2

Type: [is2GraphObject CircleType](#)
Representa la 2da circunferencia.

r_fillet

Type: [System Double](#)
Representa el radio del fillet.

right_up (**Optional**)

Type: [System Boolean](#)
Indica hacia que lado se calcula el fillet, tomando como referencia el segmento imaginario "Sx" que se forma entre los centros de ambas circunferencias.
Nota: Si el valor es **true**, el fillet se calcula hacia la derecha-o-arriba del segmento "Sx". Por el contrario si el valor es **false**, el fillet se calcula hacia la izquierda-o-abajo del segmento.

c1_outside (**Optional**)

Type: [System Boolean](#)
Indica si el fillet se calcula hacia adentro o hacia afuera de la circunferencia "C1".
Nota: Si el valor es **true**, el fillet se calcula hacia el exterior de la circunferencia. Por el contrario si el valor es **false**, el fillet se calcula hacia el interior de la circunferencia. En ambos casos su

aplicación es sobre la circunferencia "C1".

c2_outside (Optional)

Type: [System Boolean](#)

Indica si el fillet se calcula hacia adentro o hacia afuera de la circunferencia "C2".

Nota: Si el valor es **true**, el fillet se calcula hacia el exterior de la circunferencia. Por el contrario si el valor es **false**, el fillet se calcula hacia el interior de la circunferencia. En ambos casos su aplicación es sobre la circunferencia "C2".

Return Value

Type: [ArcType](#)

Devuelve un tipo Arco que representa el fillet calculado las dos circunferencias dadas.

▲ See Also

Reference

[is2GraphObj Class](#)

[CircleCircleFillet Overload](#)

[is2GraphObject Namespace](#)

[is2GraphObject ArcType](#)

[is2GraphObject CircleType](#)

is2GraphObj CircleCircleFillet Method (CircleType, CircleType, Double, PointType , PointType , PointType , Boolean, Boolean, Boolean)

Calcula el fillet (EMPALME) entre dos circunferencias "C1" y "C2" dados, con radio determinado por "r_fillet".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static void CircleCircleFillet(  
    CircleType C1,  
    CircleType C2,  
    double r_fillet,  
    out PointType P1,  
    out PointType P2,  
    out PointType Pc,  
    bool right_up = true,  
    bool c1_outside = true,  
    bool c2_outside = true  
)
```

Parameters

C1

Type: [is2GraphObject CircleType](#)

Representa la 1ra circunferencia.

C2

Type: [is2GraphObject CircleType](#)

Representa la 2da circunferencia.

r_fillet

Type: [System Double](#)

Representa el radio del fillet.

P1

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto del arc-fillet que pertenece a la circunferencia.

P2

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto del arc-fillet que pertenece al segmento de recta.

Pc

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto centro del arc-fillet.

right_up (Optional)

Type: [System Boolean](#)

Indica hacia que lado se calcula el fillet, tomando como referencia el segmento imaginario "Sx" que se forma entre los centros de ambas circunferencias.

Nota: Si el valor es **true**, el fillet se calcula hacia la derecha-o-arriba del segmento "Sx". Por el contrario si el valor es **false**, el fillet se calcula hacia la izquierda-o-abajo del segmento.

c1_outside (Optional)

Type: [System Boolean](#)

Indica si el fillet se calcula hacia adentro o hacia afuera de la circunferencia "C1".

Nota: Si el valor es **true**, el fillet se calcula hacia el exterior de la circunferencia. Por el contrario si el valor es **false**, el fillet se calcula hacia el interior de la circunferencia. En ambos casos su aplicación es sobre la circunferencia "C1".

c2_outside (Optional)

Type: [System Boolean](#)

Indica si el fillet se calcula hacia adentro o hacia afuera de la circunferencia "C2".

Nota: Si el valor es **true**, el fillet se calcula hacia el exterior de la circunferencia. Por el contrario si el valor es **false**, el fillet se calcula hacia el interior de la circunferencia. En ambos casos su aplicación es sobre la circunferencia "C2".

Return Value

Type:

Devuelve un tipo void. Por referencia mediante los parámetros P1, P2 y Pc se obtienen los 3 puntos del arc-fillet.

▲ **Remarks** Nota: "P1" pertenece a la circunferencia "C1", "P2" pertenece a la circunferencia "C2" y "Pc" es el centro del arco.

▲ **See Also**

Reference

[is2GraphObj Class](#)

[CircleCircleFillet Overload](#)

is2GraphObject Namespace

is2GraphObj CircleCircleIntersect Method

Calcula el o los puntos de intercepción entre dos circunferencias.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static byte CircleCircleIntersect(  
    CircleType C1,  
    CircleType C2,  
    out PointType P1,  
    out PointType P2  
)
```

Parameters

C1

Type: [is2GraphObject CircleType](#)

Primera circunferencia.

C2

Type: [is2GraphObject CircleType](#)

Segunda circunferencia.

P1

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia uno de los puntos de intersección, si existe.

P2

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el otro punto de intersección, si existe.

Return Value

Type: [Byte](#)

Devuelve un valor entero corto que indica lo siguiente:

0 - Las circunferencias son exteriores. No existen puntos de intersección.

1 - Las circunferencias son tangentes. Existe solo un punto de intersección.

2 - Las circunferencia son secantes. Existen dos puntos de intersección.

▲ **Remarks** - Si las circunferencias son exteriores, entonces "P1"

y "P2" retornarán NULL.

- Si las circunferencias son tangente, entonces solo existe un punto de intersección por lo que "P1" y "P2" serán iguales.
- Si las circunferencias son secantes, entonces "P1" y "P2" serán distintos.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject CircleType](#)

[is2GraphObject PointType](#)

is2GraphObj

CircleCircleRelationShip Method

Determina la relacion que hay entre 2 circunfencias.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static CircleCircleRelation CircleCircleRe  
    CircleType C1,  
    CircleType C2  
)
```

Parameters

C1

Type: [is2GraphObject CircleType](#)
Representa la 1ra circunferencia.

C2

Type: [is2GraphObject CircleType](#)
Representa la 2da circunferencia.

Return Value

Type: [CircleCircleRelation](#)

Devuelve un tipo enum que indica la relación entre dos circunferencias.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject CircleType](#)

[is2GraphObject CircleCircleRelation](#)

is2GraphObj CircleCoefficient Method

Determina los coeficientes de la ecuación general de la circunferencia ($x^2 + y^2 + Ax + By + C = 0$) para la circunferencia 'Cir' dada.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public static void CircleCoefficient(  
    CircleType Cir,  
    out double A,  
    out double B,  
    out double C  
)
```

Parameters

Cir

Type: [is2GraphObject CircleType](#)

Representa la circunferencia.

A

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor del coeficiente A.

B

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor del coeficiente B.

C

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor del coeficiente C.

Return Value

Type:

Devuelve un tipo void. Por referencia mediante los parámetros A, B, C se obtienen los coeficientes de la ecuación.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj CircleEllipseIntersect Method

[Missing <summary> documentation for

"M:is2GraphObject.is2GraphObj.CircleEllipseIntersect(is2GraphObject.CircleType,is2Gra

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public static byte CircleEllipseIntersect(  
    CircleType C1,  
    EllipseType E2,  
    out PointType P1,  
    out PointType P2,  
    out PointType P3,  
    out PointType P4  
)
```

Parameters

C1

Type: [is2GraphObject CircleType](#)

[Missing <param name="C1"/> documentation for

"M:is2GraphObject.is2GraphObj.CircleEllipseIntersect(is2GraphObject.CircleType,is2C

E2

Type: [is2GraphObject EllipseType](#)

[Missing <param name="E2"/> documentation for

"M:is2GraphObject.is2GraphObj.CircleEllipseIntersect(is2GraphObject.CircleType,is2C

P1

Type: [is2GraphObject PointType](#)

[Missing <param name="P1"/> documentation for

"M:is2GraphObject.is2GraphObj.CircleEllipseIntersect(is2GraphObject.CircleType,is2C

P2

Type: [is2GraphObject PointType](#)

[Missing <param name="P2"/> documentation for

"M:is2GraphObject.is2GraphObj.CircleEllipseIntersect(is2GraphObject.CircleType,is2C

P3

Type: [is2GraphObject PointType](#)

[Missing <param name="P3"/> documentation for

"M:is2GraphObject.is2GraphObj.CircleEllipseIntersect(is2GraphObject.CircleType,is2C

P4

Type: [is2GraphObject PointType](#)

[Missing <param name="P4"/> documentation for

"M:is2GraphObject.is2GraphObj.CircleEllipseIntersect(is2GraphObject.CircleType,is2C

Return Value

Type: [Byte](#)

[Missing <returns> documentation for

"M:is2GraphObject.is2GraphObj.CircleEllipseIntersect(is2GraphObject.CircleType,is2Gra

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj CircleLineIntersect Method

Calcula el o los puntos de intercepción entre una línea y un circunferencia.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static byte CircleLineIntersect(  
    CircleType C,  
    LineType L,  
    out PointType P1,  
    out PointType P2  
)
```

Parameters

C

Type: [is2GraphObject CircleType](#)
Representa la circunferencia.

L

Type: [is2GraphObject LineType](#)
Representa la línea.

P1

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia uno de los puntos de intersección, si existe.

P2

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia el otro punto de intersección, si existe.

Return Value

Type: [Byte](#)

Devuelve un valor entero corto que indica lo siguiente:

0 - La línea es exterior a la circunferencia. No existen puntos de intersección.

1 - La línea es tangente a la circunferencia. Existe solo un punto de intersección.

2 - La línea es secante a la circunferencia. Existen dos puntos de intersección.

▲ **Remarks** - Si la línea es exterior a la circunferencia "P1" y "P2"

retornarán NULL.

- Si la línea es tangente a la circunferencia "P1" y "P2" serán iguales, pues solo existe un punto de intersección.
- Si la línea es secante a la circunferencia "P1" y "P2" serán distintos pues la línea intersecta por dos puntos a la circunferencia.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)



[is2GraphObject CircleType](#)

[is2GraphObject LineType](#)

[is2GraphObject PointType](#)

is2GraphObj CircleSegmentFillet Method

▲ Overload List

	Name	Description
	CircleSegmentFillet(CircleType, SegmentType, Double, Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre una circunferencia "C" y un segmento de recta "S" dados, con radio determinado por "r_fillet".
	CircleSegmentFillet(CircleType, SegmentType, Double, PointType , PointType , PointType , Boolean, Boolean, Boolean)	Calcula el fillet (EMPALME) entre una circunferencia "C" y un segmento de recta "S" dados, con radio determinado por "r_fillet".

[Top](#)

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj CircleSegmentFillet Method (CircleType, SegmentType, Double, Boolean, Boolean, Boolean)

Calcula el fillet (EMPALME) entre una circunferencia "C" y un segmento de recta "S" dados, con radio determinado por "r_fillet".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static ArcType CircleSegmentFillet(  
    CircleType C,  
    SegmentType S,  
    double r_fillet,  
    bool right = true,  
    bool outside = true,  
    bool up = true  
)
```

Parameters

C

Type: [is2GraphObject CircleType](#)
Representa la circunferencia.

S

Type: [is2GraphObject SegmentType](#)
Representa el segmento de recta.

r_fillet

Type: [System Double](#)
Representa el radio del fillet.

right (**Optional**)

Type: [System Boolean](#)
Indica hacia que lado del segmento "S" dado se calcula el fillet (empalme). Los lados quedan determinados por la recta que parte del centro de la circunferencia "C" y es perpendicular al segmento "S". Si el valor es **true** el fillet se calcula hacia la derecha del segmento "S", por el contrario si el valor es **false** el fillet se calcula hacia la izquierda de dicho segmento.

outside (**Optional**)

Type: [System Boolean](#)
Indica si el fillet (empalme) se calcula exterior o interior a la circunferencia "C" dada. Si el valor es **true** el fillet se calcula exterior a la circunferencia, por el contrario si el valor es **false** el fillet se calcula hacia el interior de la circunferencia.

up (**Optional**)

Type: [System Boolean](#)

Indica si el fillet se calcula sobre el segmento "S" dado o por debajo de él. Si el valor es **true** el fillet se calcula sobre el segmento "S" dado, por el contrario si el valor es **false** el fillet se calcula por debajo de dicho segmento.

Return Value

Type: [ArcType](#)

Devuelve un tipo Arco que representa el fillet calculado entre la circunferencia y el segmento de recta dados.

▲ Exceptions

Exception	Condition
FilletException	Se lanza cuando se intenta crear un fillet cuyo valor de radio no lo permite por ser o muy pequeño o muy grande.

▲ See Also

Reference

[is2GraphObj Class](#)

[CircleSegmentFillet Overload](#)

[is2GraphObject Namespace](#)

is2GraphObj CircleSegmentFillet Method (CircleType, SegmentType, Double, PointType , PointType , PointType , Boolean, Boolean, Boolean)

Calcula el fillet (EMPALME) entre una circunferencia "C" y un segmento de recta "S" dados, con radio determinado por "r_fillet".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static void CircleSegmentFillet(  
    CircleType C,  
    SegmentType S,  
    double r_fillet,  
    out PointType P1,  
    out PointType P2,  
    out PointType Pc,  
    bool right = true,  
    bool outside = true,  
    bool up = true  
)
```

Parameters

C

Type: [is2GraphObject CircleType](#)

Representa la circunferencia.

S

Type: [is2GraphObject SegmentType](#)

Representa el segmento de recta.

r_fillet

Type: [System Double](#)

Representa el radio del fillet.

P1

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto del arc-fillet que pertenece a la circunferencia.

P2

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto del arc-fillet que pertenece al segmento de recta.

Pc

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto centro del arc-fillet.

right (Optional)

Type: **System Boolean**

Indica hacia que lado del segmento "S" dado se calcula el fillet (empalme). Los lados quedan determinados por la recta que parte del centro de la circunferencia "C" y es perpendicular al segmento "S". Si el valor es **true** el fillet se calcula hacia la derecha del segmento "S", por el contrario si el valor es **false** el fillet se calcula hacia la izquierda de dicho segmento.

outside (Optional)

Type: **System Boolean**

Indica si el fillet (empalme) se calcula exterior o interior a la circunferencia "C" dada. Si el valor es **true** el fillet se calcula exterior a la circunferencia, por el contrario si el valor es **false** el fillet se calcula hacia el interior de la circunferencia.

up (Optional)

Type: **System Boolean**

Indica si el fillet se calcula sobre el segmento "S" dado o por debajo de él. Si el valor es **true** el fillet se calcula sobre el segmento "S" dado, por el contrario si el valor es **false** el fillet se calcula por debajo de dicho segmento.

Return Value

Type:

Devuelve un tipo void. Por referencia mediante los parámetros P1, P2 y Pc se obtienen los 3 puntos del arc-fillet.

▲ Exceptions

Exception	Condition
FilletException	Se lanza cuando se intenta crear un fillet cuyo valor de radio no lo permite por ser o muy pequeño o muy grande.

▲ **Remarks** Nota: "P1" pertenece a la circunferencia, "P2" pertenece al segmento y "Pc" es el centro del arco.

▲ See Also

Reference

[is2GraphObj Class](#)

[CircleSegmentFillet Overload](#)

[is2GraphObject Namespace](#)

[is2GraphObject CircleType](#)

[is2GraphObject SegmentType](#)

[is2GraphObject PointType](#)

is2GraphObj

CircleSegmentIntersect Method

Calcula el o los puntos de intercepcion entre una circunferencia y un segmento de recta.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static byte CircleSegmentIntersect(  
    CircleType C,  
    SegmentType S,  
    out PointType P1,  
    out PointType P2  
)
```

Parameters

C

Type: [is2GraphObject CircleType](#)
Representa la circunferencia.

S

Type: [is2GraphObject SegmentType](#)
Representa el segmento de recta.

P1

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia uno de los puntos de intersección, si existe.

P2

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia el otro punto de intersección, si existe.

Return Value

Type: [Byte](#)

Devuelve un valor entero corto que indica lo siguiente:

- 0 - El segmento no intersecta a la circunferencia.
- 1 - El segmento intersecta a la circunferencia en un único punto.
- 2 - El segmento intersecta a la circunferencia en dos puntos.

- ▲ **Remarks** - Si el segmento no intercepta a la circunferencia, entonces "P1" y "P2" retornarán NULL.
- Si el segmento intercepta a la circunferencia en un punto, entonces "P1" y "P2" son iguales.

- Si el segmento intercepta a la circunferencia en dos puntos, entonces "P1" y "P2" son distintos.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject CircleType](#)

[is2GraphObject SegmentType](#)

[is2GraphObject PointType](#)

is2GraphObj

CircleSegmentRelationShip

Method

Determina la relación que hay entre una circunferencias y un segmento.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static CircleSegmentRelation CircleSegment  
    CircleType C,  
    SegmentType S  
)
```

Parameters

C

Type: [is2GraphObject CircleType](#)
Representa la circunferencia.

S

Type: [is2GraphObject SegmentType](#)
Representa el segmento.

Return Value

Type: [CircleSegmentRelation](#)

Devuelve un tipo enum que indica la relacion entre la circunferencia y el segmento.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject CircleType](#)

[is2GraphObject SegmentType](#)

[is2GraphObject CircleSegmentRelation](#)

is2GraphObj CircleTangentToLine Method

Determina la circunferencia que tiene su centro en "P" y para la cual la línea "L" dada es tangente.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static CircleType CircleTangentToLine(  
    LineType L,  
    PointType P  
)
```

Parameters

L

Type: [is2GraphObject LineType](#)

Línea que es tangente a la circunferencia calculada.

P

Type: [is2GraphObject PointType](#)

Punto centro de la circunferencia.

Return Value

Type: [CircleType](#)

Devuelve una circunferencia que tiene su centro en el punto "P" y que es tangente a la línea dada.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject LineType](#)

[is2GraphObject PointType](#)

is2GraphObj

ComplementaryAngle Method

Determina dado un angulo, el valor del angulo complementario

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static double ComplementaryAngle(  
    double angle  
)
```

Parameters

angle

Type: [System Double](#)

Valor de ángulo en grados.

Return Value

Type: [Double](#)

Devuelve el valor de ángulo complementario al ángulo dado.

▲ **Remarks** Se define como ángulo complementario de un ??

▲ **See Also**

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj EllipseCoefficient Method

Determina los coeficientes de la ecuación general de la elipse ($Ax^2 + By^2 + Cx + Dy + E = 0$) para la elipse 'Elip' dada.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public static void EllipseCoefficient(  
    EllipseType Elip,  
    out double A,  
    out double B,  
    out double C,  
    out double D,  
    out double E  
)
```

Parameters

Elip

Type: [is2GraphObject EllipseType](#)

[Missing <param name="Elip"/> documentation for

"M:is2GraphObject.is2GraphObj.EllipseCoefficient(is2GraphObject.EllipseType,System.I

A

Type: [System Double](#)

[Missing <param name="A"/> documentation for

"M:is2GraphObject.is2GraphObj.EllipseCoefficient(is2GraphObject.EllipseType,System.I

B

Type: [System Double](#)

[Missing <param name="B"/> documentation for

"M:is2GraphObject.is2GraphObj.EllipseCoefficient(is2GraphObject.EllipseType,System.I

C

Type: [System Double](#)

[Missing <param name="C"/> documentation for

"M:is2GraphObject.is2GraphObj.EllipseCoefficient(is2GraphObject.EllipseType,System.I

D

Type: [System Double](#)

[Missing <param name="D"/> documentation for

"M:is2GraphObject.is2GraphObj.EllipseCoefficient(is2GraphObject.EllipseType,System.I

E

Type: [System Double](#)

[Missing <param name="E"/> documentation for

"M:is2GraphObject.is2GraphObj.EllipseCoefficient(is2GraphObject.EllipseType,System.I

Return Value

Type:

Devuelve un tipo void. Por referencia mediante los parámetros A, B, C, D, E se obtienen los coeficientes de la ecuación.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj EllipseLineIntersect Method

[Missing <summary> documentation for

"M:is2GraphObject.is2GraphObj.EllipseLineIntersect(is2GraphObject.EllipseType,is2Grap

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public static byte EllipseLineIntersect(  
    EllipseType E,  
    LineType L,  
    out PointType P1,  
    out PointType P2  
)
```

Parameters

E

Type: [is2GraphObject EllipseType](#)

[Missing <param name="E"/> documentation for

"M:is2GraphObject.is2GraphObj.EllipseLineIntersect(is2GraphObject.EllipseType,is2G

L

Type: [is2GraphObject LineType](#)

[Missing <param name="L"/> documentation for

"M:is2GraphObject.is2GraphObj.EllipseLineIntersect(is2GraphObject.EllipseType,is2G

P1

Type: [is2GraphObject PointType](#)

[Missing <param name="P1"/> documentation for

"M:is2GraphObject.is2GraphObj.EllipseLineIntersect(is2GraphObject.EllipseType,is2G

P2

Type: [is2GraphObject PointType](#)

[Missing <param name="P2"/> documentation for

"M:is2GraphObject.is2GraphObj.EllipseLineIntersect(is2GraphObject.EllipseType,is2G

Return Value

Type: [Byte](#)

[Missing <returns> documentation for

"M:is2GraphObject.is2GraphObj.EllipseLineIntersect(is2GraphObject.EllipseType,is2Grap

See Also

Reference

[is2GraphObj Class](#)

is2GraphObject Namespace

is2GraphObj FarInXAxis Method

Determina cuál de los puntos "P1" y "P2" dados esta más alejado de "pbase" sobre el eje de las abcisas (Eje X).

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static PointType FarInXAxis(  
    PointType pbase,  
    PointType P1,  
    PointType P2  
)
```

Parameters

pbase

Type: [is2GraphObject PointType](#)

Punto base o de referencia respecto al cuál se calculará la lejanía sobre el eje X.

P1

Type: [is2GraphObject PointType](#)

Primer punto.

P2

Type: [is2GraphObject PointType](#)

Segundo punto.

Return Value

Type: [PointType](#)

Devuelve el punto más alejado sobre el eje de las X al punto base dado.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj FarInYAxis Method

Determina cuál de los puntos "P1" y "P2" dados esta más alejado de "pbase" sobre el eje de las ordenadas (Eje Y).

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static PointType FarInYAxis(  
    PointType pbase,  
    PointType P1,  
    PointType P2  
)
```

Parameters

pbase

Type: [is2GraphObject PointType](#)

Punto base o de referencia respecto al cuál se calculará la lejanía sobre el eje Y.

P1

Type: [is2GraphObject PointType](#)

Primer punto.

P2

Type: [is2GraphObject PointType](#)

Segundo punto.

Return Value

Type: [PointType](#)

Devuelve el punto más alejado sobre el eje de las Y al punto base dado.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj FarToOrigen Method

Determina cuál de los puntos "P1" y "P2" dados está más lejano del origen de coordenadas [0; 0].

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static PointType FarToOrigen(  
    PointType P1,  
    PointType P2  
)
```

Parameters

P1

Type: [is2GraphObject PointType](#)
Primer punto.

P2

Type: [is2GraphObject PointType](#)
Segundo punto.

Return Value

Type: [PointType](#)

Devuelve el punto más alejado del origen de coordenadas.

See Also

Reference



[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObjFarToPoint Method

▲ Overload List

	Name	Description
	FarToPoint(PointType, ListPointType)	Determina de una lista de puntos definidas por "list" el punto más alejado del punto base "pbase".
	FarToPoint(PointType, PointType, PointType)	Determina cuál de los dos puntos dados "P1" y "P2" está más alejado al punto "pbase".

[Top](#)

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj FarToPoint Method (PointType, List PointType)

Determina de una lista de puntos definidas por "list" el punto más alejado del punto base "pbase".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static PointType FarToPoint(  
    PointType pbase,  
    List<PointType> list  
)
```

Parameters

pbase

Type: [is2GraphObject PointType](#)

Punto base o de referencia respecto al cuál se calculará la lejanía.

list

Type: [System.Collections.Generic List PointType](#)

Lista de puntos a comprobar.

Return Value

Type: [PointType](#)

Devuelve el punto más lejano al punto "pbase".

▲ See Also

Reference

[is2GraphObj Class](#)

[FarToPoint Overload](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj FarToPoint Method (PointType, PointType, PointType)

Determina cuál de los dos puntos dados "P1" y "P2" está más alejado al punto "pbase".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static PointType FarToPoint(  
    PointType pbase,  
    PointType P1,  
    PointType P2  
)
```

Parameters

pbase

Type: [is2GraphObject PointType](#)

Punto base o de referencia respecto al cuál se calculará la lejanía.

P1

Type: [is2GraphObject PointType](#)

Primer punto.

P2

Type: [is2GraphObject PointType](#)

Segundo punto.

Return Value

Type: [PointType](#)

Devuelve el punto más alejado del punto base dado.

▲ See Also

Reference

[is2GraphObj Class](#)

[FarToPoint Overload](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObjGradToRad Method

Convierte de Grados a Radianes.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public static double GradToRad(  
    double angle  
)
```

Parameters

angle

Type: [SystemDouble](#)

Valor de ángulo en grados.

Return Value

Type: [Double](#)

Devuelve un valor de ángulo en radianes.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj isColinearSegment Method

Determina si dos segmetos "S1" y "S2" dados son colineales.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool isColinearSegment(  
    SegmentType S1,  
    SegmentType S2  
)
```

Parameters

S1

Type: [is2GraphObject SegmentType](#)
Primer segmento.

S2

Type: [is2GraphObject SegmentType](#)
Segundo segmento.

Return Value

Type: [Boolean](#)

Devuelve **true** si ambos segmentos son colineales. En caso contrario devuelve **false**.

▲ **Remarks** Se define la condición de colinealidad de dos segmentos si estos pertenecen a una misma recta.

▲ **See Also**

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject SegmentType](#)

is2GraphObj isEqualArc Method

Comprueba si dos arcos dados "A1" y "A2" son iguales.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool isEqualArc(  
    ArcType A1,  
    ArcType A2  
)
```

Parameters

A1

Type: [is2GraphObject ArcType](#)
Primer arcos.

A2

Type: [is2GraphObject ArcType](#)
Segundo arco.

Return Value

Type: [Boolean](#)

Devuelve **true** si ambos arcos son iguales. En caso contrario devuelve **false**.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject ArcType](#)

is2GraphObj isEqualCero Method

Determina si el parámetro 'value' se considera un cero real.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool isEqualCero(  
    double value  
)
```

Parameters

value

Type: [System Double](#)

Valor decimal a comprobar.

Return Value

Type: [Boolean](#)

Devuelve **true** si el valor pasado por parámetro presenta un cero real, en caso contrario devuelve **false**.

▲ **Remarks** Se considera que un valor es cero real si este valor es menor que la constante definida CERO_REAL. Por defecto se considera la precision de la constante CERO_REAL igual a: (+-)1E-6 => 0.000001.

▲ **See Also**

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj isEqualCircle Method

Comprueba si dos circunferencias dadas "C1" y "C2" son iguales.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool isEqualCircle(  
    CircleType C1,  
    CircleType C2  
)
```

Parameters

C1

Type: [is2GraphObject CircleType](#)
Primera circunferencia.

C2

Type: [is2GraphObject CircleType](#)
Segunda circunferencia.

Return Value

Type: [Boolean](#)

Devuelve **true** si ambos circulos son iguales, en caso contrario devuelve **false**.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject CircleType](#)

is2GraphObj isEqualPoint Method

Comprueba si dos puntos dados "P1" y "P2" son iguales.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool isEqualPoint(  
    PointType P1,  
    PointType P2  
)
```

Parameters

P1

Type: [is2GraphObject PointType](#)
Primer punto.

P2

Type: [is2GraphObject PointType](#)
Segundo punto.

Return Value

Type: [Boolean](#)

Devuelve **true** si ambas puntos son iguales, en caso contrario devuelve **false**.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj isEqualSegment Method

Comprueba si dos Segmentos dado "S1" y "S2" son iguales.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool isEqualSegment(  
    SegmentType S1,  
    SegmentType S2  
)
```

Parameters

S1

Type: [is2GraphObject SegmentType](#)
Primer segmento.

S2

Type: [is2GraphObject SegmentType](#)
Segundo segmento.

Return Value

Type: [Boolean](#)

Devuelve **true** si ambos segmentos son iguales, en caso contrario devuelve **false**.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject SegmentType](#)

is2GraphObj isEqualValues Method

Determina si los dos valor pasados por parámetro son iguales.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool isEqualValues(  
    double a,  
    double b  
)
```

Parameters

a

Type: [System Double](#)

Primer valor decimal.

b

Type: [System Double](#)

Segundo valor decimal.

Return Value

Type: [Boolean](#)

Devuelve **true** si ambas valores son iguales, en caso contrario devuelve **false**.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj isNegative Method

Determina si el valor pasado por parámetro es negativo.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool isNegative(  
    double value  
)
```

Parameters

value

Type: [System Double](#)

Valor decimal a comprobar.

Return Value

Type: [Boolean](#)

Devuelve **true** si el valor pasado por parámetro es negativo, en caso contrario devuelve **false**.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj isPositive Method

Determina si el valor pasado por parámetro es positivo.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool isPositive(  
    double value  
)
```

Parameters

value

Type: [System Double](#)

Valor decimal a comprobar.

Return Value

Type: [Boolean](#)

Develve **true** si el valor pasado por parámetro es positivo, en caso contrario devuelve **false**.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj

LineAccordingLineAt Method

Determmina una línea que pasa por el punto "P" y que forma con la línea "L" dada, un ángulo determinado por el parámetro "angle".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static LineType LineAccordingLineAt(  
    LineType L,  
    PointType P,  
    double angle  
)
```

Parameters

L

Type: [is2GraphObject LineType](#)

Línea a partir de la cual se determina la otra línea que forma con esta un ángulo determinado.

P

Type: [is2GraphObject PointType](#)

Punto por el que pasa la línea oblicua a calcular.

angle

Type: [System Double](#)

Valor de ángulo en grados.

Return Value

Type: [LineType](#)

Devuelve una línea que pasa por el punto "P" y que es oblicua a la línea "L" formando un ángulo determinado.

See Also

Reference

[is2GraphObj Class](#)



[is2GraphObject Namespace](#)

[is2GraphObject LineType](#)

[is2GraphObject PointType](#)

is2GraphObj LineCoefficient Method

▲ Overload List

	Name	Description
	LineCoefficient(LineType, Double , Double , Double)	Determina los coeficientes de la ecuacion general de la recta ($Ax + By + C = 0$) para la recta 'L' dada.
	LineCoefficient(SegmentType, Double , Double , Double)	Determina los coeficientes de la ecuacion general de la recta ($Ax + By + C = 0$) para el segmento 'S' dado.

[Top](#)

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj LineCoefficient Method (LineType, Double , Double , Double)

Determina los coeficientes de la ecuacion general de la recta ($Ax + By + C = 0$) para la recta 'L' dada.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public static void LineCoefficient(  
    LineType L,  
    out double A,  
    out double B,  
    out double C  
)
```

Parameters

L

Type: [is2GraphObject LineType](#)

Representa el segmento.

A

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor del coeficiente A.

B

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor del coeficiente B.

C

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor del coeficiente C.

Return Value

Type:

Devuelve un tipo void. Por referencia mediante los parámetros A, B, C se obtienen los coeficientes de la ecuación.

▲ See Also

Reference

[is2GraphObj Class](#)

[LineCoefficient Overload](#)

[is2GraphObject Namespace](#)

is2GraphObj LineCoefficient Method (SegmentType, Double , Double , Double)

Determina los coeficientes de la ecuacion general de la recta ($Ax + By + C = 0$) para el segmento 'S' dado.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public static void LineCoefficient(  
    SegmentType S,  
    out double A,  
    out double B,  
    out double C  
)
```

Parameters

S

Type: [is2GraphObject SegmentType](#)

Representa el segmento.

A

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor del coeficiente A.

B

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor del coeficiente B.

C

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor del coeficiente C.

Return Value

Type:

Devuelve un tipo void. Por referencia mediante los parámetros A, B, C se obtienen los coeficientes de la ecuación.

See Also

Reference

[is2GraphObj Class](#)

[LineCoefficient Overload](#)

[is2GraphObject Namespace](#)

is2GraphObj LineLineAngle Method

Determina el ángulo según su tipo (agudo u obtuso) que se forma entre dos líneas "L1" y "L2" dadas.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static double LineLineAngle(  
    LineType L1,  
    LineType L2,  
    bool agude = true  
)
```

Parameters

L1

Type: [is2GraphObject LineType](#)
Primera línea.

L2

Type: [is2GraphObject LineType](#)
Segunda línea.

agude (Optional)

Type: [System Boolean](#)

Indica el tipo de ángulo que se quiere calcular. Si es **true** se calcula el valor para el ángulo agudo, por el contrario si es **false** se calcula el valor del angulo obtuso.

Return Value

Type: [Double](#)

Devuelve el ángulo en grados (agudo u obtuso) que se forma entre las líneas dadas.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject LineType](#)

is2GraphObj LineLineIntersect Method

Calcula el punto de intercepción entre dos líneas.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool LineLineIntersect(  
    LineType L1,  
    LineType L2,  
    out PointType P  
)
```

Parameters

L1

Type: [is2GraphObject LineType](#)
Primera línea.

L2

Type: [is2GraphObject LineType](#)
Segunda línea.

P

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia el punto de intercepción entre las dos líneas, si existe.
Nota: Si las líneas no se intersectan el parámetro "P" retorna NULL.

Return Value

Type: [Boolean](#)

Devuelve **true** si ambas líneas se intercertan en el plano. En caso contrario devuelve **false**.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject LineType](#)

[is2GraphObject PointType](#)

is2GraphObj LineTangentToCircle Method

Determina las rectas tangentes a la circunferencia "C" dada y que pasan por el punto "P".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static void LineTangentToCircle(  
    CircleType C,  
    PointType P,  
    out LineType L1,  
    out LineType L2  
)
```

Parameters

C

Type: [is2GraphObject CircleType](#)

Circunferencia para que le se quieren calcular las rectas tangentes y que pasan por el punto dado.

P

Type: [is2GraphObject PointType](#)

Punto por el que pasan las rectas tangentes a la circunferencia.

L1

Type: [is2GraphObject LineType](#)

Parámetro de salida (out). Retorna por referencia la primera de las líneas tangentes a la circunferencia.

L2

Type: [is2GraphObject LineType](#)

Parámetro de salida (out). Retorna por referencia la segunda de las líneas tangentes a la circunferencia.

Return Value

Type:

Devuelve un tipo void. Por referencia devuelve las dos líneas que cumplen con la propiedad de pasar por el punto dado y ser tangente a la circunferencia.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObject CircleType
is2GraphObject LineType
is2GraphObject PointType

is2GraphObj MakeMPQx Method

Resuelve un sistema de ecuaciones de 2 con 2 por el método de sustitución, despejando 'X' en la ecuación lineal para luego sustituirla en la ecuación cuadrática. De lo que se obtiene un polinomio de la forma: $mx^2 + px + q$.

Nota: Los parámetros A, B, C, A1, B1, C1 se interpretan de la siguiente forma:

- A, B, C: coeficiente de una ecuación lineal de la forma $Ax + By + C = 0$
- A1, B1, C1: coeficientes de una ecuación cuadrática $X^2 + y^2 + A1x + B1y + C1 = 0$

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static void MakeMPQx(  
    double A,  
    double B,  
    double C,  
    double A1,  
    double B1,  
    double C1,  
    out double m,  
    out double p,  
    out double q  
)
```

Parameters

A

Type: [System Double](#)

Coeficiente de la abcisa (X) de la ecuación lineal.

B

Type: [System Double](#)

Coeficiente de la ordenada (Y) de la ecuación lineal.

C

Type: [System Double](#)

Coeficiente independiente de la ecuación lineal.

A1

Type: [System Double](#)

Coeficiente de la abcisa (X) de la ecuación cuadrática.

B1

Type: [System Double](#)

Coeficiente de la ordenada (Y) de la ecuación cuadrática.

C1

Type: [System Double](#)

Coeficiente independiente de la ecuación cuadrática.

m

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor del

coeficiente cuadrático

p

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor

q

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor

Return Value

Type:

Devuelve un tipo void. Por referencia mediante los parámetros m, p, q se obtienen los coeficientes para un polinomio de la forma $mx^2 + px + q$.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj MakeMPQy Method

Resuelve un sistema de ecuaciones de 2 con 2 por el método de sustitución, despejando 'Y' en la ecuación lineal para luego sustituirla en la ecuación cuadrática. De lo que se obtiene un polinomio de la forma: $mx^2 + px + q$. **Nota:** Los parámetros A, B, C, A1, B1, C1 se interpretan de la siguiente forma:

- A, B, C: coeficiente de una ecuación lineal de la forma $Ax + By + C = 0$ - A1, B1, C1: coeficientes de una ecuación cuadrática $X^2 + y^2 + A1x + B1y + C1 = 0$

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static void MakeMPQy(  
    double A,  
    double B,  
    double C,  
    double A1,  
    double B1,  
    double C1,  
    out double m,  
    out double p,  
    out double q  
)
```

Parameters

A

Type: [System Double](#)

[Missing <param name="A"/> documentation for

"M:is2GraphObject.is2GraphObj.MakeMPQy(System.Double,System.Double,System.D

B

Type: [System Double](#)

[Missing <param name="B"/> documentation for

"M:is2GraphObject.is2GraphObj.MakeMPQy(System.Double,System.Double,System.D

C

Type: [System Double](#)

[Missing <param name="C"/> documentation for

"M:is2GraphObject.is2GraphObj.MakeMPQy(System.Double,System.Double,System.D

A1

Type: [System Double](#)

[Missing <param name="A1"/> documentation for

"M:is2GraphObject.is2GraphObj.MakeMPQy(System.Double,System.Double,System.D

B1

Type: [System Double](#)

[Missing <param name="B1"/> documentation for

"M:is2GraphObject.is2GraphObj.MakeMPQy(System.Double,System.Double,System.D

C1

Type: [System Double](#)

[Missing <param name="C1"/> documentation for

"M:is2GraphObject.is2GraphObj.MakeMPQy(System.Double,System.Double,System.D

m

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor

p

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor

q

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor

Return Value

Type:

Devuelve un tipo void. Por referencia mediante los parámetros m, p, q se obtienen los coeficientes para un polinomio de la forma $mx^2 + px + q$.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj MaxXSegment Method

Determina el mayor valor de la coordenada X que contiene el segmento 'S' dado.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static double MaxXSegment(  
    SegmentType S  
)
```

Parameters

S

Type: [is2GraphObject SegmentType](#)

Representa el segmento al que se le determinará su mayor X.

Return Value

Type: [Double](#)

Devuelve un double que representa el mayor valor de la coordenada X del segmento.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject SegmentType](#)

is2GraphObj MaxYSegment Method

Determina el mayor valor de la coordenada Y que contiene el segmento 'S' dado.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static double MaxYSegment(  
    SegmentType S  
)
```

Parameters

S

Type: [is2GraphObject SegmentType](#)

Representa el segmento al que se le determinará su mayor Y.

Return Value

Type: [Double](#)

Devuelve un double que representa el mayor valor de la coordenada Y del segmento.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject SegmentType](#)

is2GraphObj MayorEstricto Method

Determina si el primer parámetro es estrictamente mayor que el segundo.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool MayorEstricto(  
    double a,  
    double b  
)
```

Parameters

a

Type: [System Double](#)

Primer valor decimal.

b

Type: [System Double](#)

Segundo valor decimal.

Return Value

Type: [Boolean](#)

Devuelve **true** si se cumple que "a" es mayor estricto que "b", en caso contrario devuelve **false**.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj MayorOrEqual Method

Comprueba si el primer parámetro es mayor o igual que el segundo.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool MayorOrEqual(  
    double a,  
    double b  
)
```

Parameters

a

Type: [System Double](#)
Primer valor decimal.

b

Type: [System Double](#)
Segundo valor decimal.

Return Value

Type: [Boolean](#)

Devuelve **true** si "a" es mayor o igual que "b", en caso contrario devuelve **false**.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj MenorEstricto Method

Determina si el primer parámetro es estrictamente menor que el segundo.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool MenorEstricto(  
    double a,  
    double b  
)
```

Parameters

a

Type: [System Double](#)

Primer valor decimal.

b

Type: [System Double](#)

Segundo valor decimal.

Return Value

Type: [Boolean](#)

Devuelve **true** si se cumple que "a" es menor estricto que "b", en caso contrario devuelve **false**.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj MenorOrEqual Method

Comprueba si el parámetro 'a' es menor o igual que 'b'.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool MenorOrEqual(  
    double a,  
    double b  
)
```

Parameters

a

Type: [System Double](#)
Primer valor decimal.

b

Type: [System Double](#)
Segundo valor decimal.

Return Value

Type: [Boolean](#)

Devuelve **true** si "a" es menor o igual que "b", en caso contrario devuelve **false**.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj

MidPointBetweenPoint Method

Determina el punto medio o punto equidistante entre 2 puntos "P1" y "P2" dados.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static PointType MidPointBetweenPoint(  
    PointType P1,  
    PointType P2  
)
```

Parameters

P1

Type: [is2GraphObject PointType](#)
Primer punto.

P2

Type: [is2GraphObject PointType](#)
Segundo punto.

Return Value

Type: [PointType](#)

Devuelve un punto que es equidistante a los puntos "P1" y "P2" dados.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj MinXSegment Method

Determina el menor valor de la coordenada X que contiene el segmento 'S' dado.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static double MinXSegment(  
    SegmentType S  
)
```

Parameters

S

Type: [is2GraphObject SegmentType](#)

Representa el segmento al que se le determinará su menor X.

Return Value

Type: [Double](#)

Devuelve un double que representa el menor valor de la coordenada X del segmento.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject SegmentType](#)

is2GraphObj MinYSegment Method

Determina el menor valor de la coordenada Y que contiene el segmento 'S' dado.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static double MinYSegment(  
    SegmentType S  
)
```

Parameters

S

Type: [is2GraphObject SegmentType](#)

Representa el segmento al que se le determinará su menor Y.

Return Value

Type: [Double](#)

Devuelve un double que representa el menor valor de la coordenada Y del segmento.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject SegmentType](#)

is2GraphObj NearInXAxis Method

Determina cuál de los puntos "P1" y "P2" dados esta más cercano a "pbase" sobre el eje de las abcisas (Eje X).

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static PointType NearInXAxis(  
    PointType pbase,  
    PointType P1,  
    PointType P2  
)
```

Parameters

pbase

Type: [is2GraphObject PointType](#)

Punto base o de referencia respecto al cuál se calculará la cercanía sobre el eje X.

P1

Type: [is2GraphObject PointType](#)

Primer punto.

P2

Type: [is2GraphObject PointType](#)

Segundo punto.

Return Value

Type: [PointType](#)

Devuelve el punto más cercado sobre el eje de las X al punto base dado.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj NearInYAxis Method

Determina cuál de los puntos "P1" y "P2" dados esta más cercano a "pbase" sobre el eje de las ordenadas (Eje Y).

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static PointType NearInYAxis(  
    PointType pbase,  
    PointType P1,  
    PointType P2  
)
```

Parameters

pbase

Type: [is2GraphObject PointType](#)

Punto base o de referencia respecto al cuál se calculará la cercanía sobre el eje Y.

P1

Type: [is2GraphObject PointType](#)

Primer punto.

P2

Type: [is2GraphObject PointType](#)

Segundo punto.

Return Value

Type: [PointType](#)

Devuelve el punto más cercado sobre el eje de las Y al punto base dado.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj NearToOrigen Method

Determina cuál de los puntos "P1" y "P2" dados está más cercano al origen de coordenadas [0; 0].

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static PointType NearToOrigen(  
    PointType P1,  
    PointType P2  
)
```

Parameters

P1

Type: [is2GraphObject PointType](#)
Primer punto.

P2

Type: [is2GraphObject PointType](#)
Segundo punto.

Return Value

Type: [PointType](#)

Devuelve el punto más cercano al origen de coordenadas.

▲ See Also

Reference



[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObjNearToPoint Method

▲ Overload List

	Name	Description
	NearToPoint(PointType, ListPointType)	Determina que punto de una lista de puntos dada esta más cercano al punto "pbase".
	NearToPoint(PointType, PointType, PointType)	Determina cuál de los puntos "P1" y "P2" dados está más cercano al punto "pbase".

[Top](#)

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj NearToPoint Method (PointType, List PointType)

Determina que punto de una lista de puntos dada esta más cercano al punto "pbase".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static PointType NearToPoint(  
    PointType pbase,  
    List<PointType> list  
)
```

Parameters

pbase

Type: [is2GraphObject PointType](#)

Punto base o de referencia respecto al cuál se calculará la carcanía.

list

Type: [System.Collections.Generic List PointType](#)

Lista de puntos a comprobar.

Return Value

Type: [PointType](#)

Devuelve el punto más cercano al punto "pbase".

See Also

Reference

[is2GraphObj Class](#)

[NearToPoint Overload](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj NearToPoint Method (PointType, PointType, PointType)

Determina cuál de los puntos "P1" y "P2" dados está más cercano al punto "pbase".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static PointType NearToPoint(  
    PointType pbase,  
    PointType P1,  
    PointType P2  
)
```

Parameters

pbase

Type: [is2GraphObject PointType](#)

Punto base o de referencia respecto al cuál se calculará la cercanía.

P1

Type: [is2GraphObject PointType](#)

Primer punto.

P2

Type: [is2GraphObject PointType](#)

Segundo punto.

Return Value

Type: [PointType](#)

Devuelve el punto más cercano al punto base dado.

▲ See Also

Reference

[is2GraphObj Class](#)

[NearToPoint Overload](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj NormalizeAngle Method

Normaliza un valor de ángulo.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static double NormalizeAngle(  
    double angle  
)
```

Parameters

angle

Type: [System Double](#)

Valor de ángulo en grados.

Return Value

Type: [Double](#)

Devuelve ángulo en grados normalizado.

▲ **Remarks** El proceso de Normalización garantiza que un valor de angulo no exceda los 360 grados, para lo cuál se calcula ??

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj OppositeAngle Method

Determina dado un ángulo, el valor del ángulo opuesto en el sistema de ejes de coordenadas.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static double OppositeAngle(  
    double angle  
)
```

Parameters

angle

Type: [System Double](#)

Valor de ángulo en grados.

Return Value

Type: [Double](#)

Devuelve el valor de ángulo opuesto al ángulo dado.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj ParallelLineAt Method

Determina una línea que es paralela a la línea "L" dada y que pasa por el punto "P".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static LineType ParallelLineAt(  
    LineType L,  
    PointType P  
)
```

Parameters

L

Type: [is2GraphObject LineType](#)

Línea a partir de la cual se determina la otra línea que es paralela a esta.

P

Type: [is2GraphObject PointType](#)

Punto por el que pasa la línea paralela a calcular.

Return Value

Type: [LineType](#)

Devuelve un línea que es paralela a la linea "L" dada y que pasa por el punto "P".

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject LineType](#)

[is2GraphObject PointType](#)

is2GraphObj PerperdicularLineAt Method

Determina una línea que es perpendicular a la línea "L" dada y que pasa por el punto "P".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static LineType PerpendicularLineAt(  
    LineType L,  
    PointType P  
)
```

Parameters

L

Type: [is2GraphObject LineType](#)

Línea a partir de la cual se determina la otra línea que es perpendicular a esta.

P

Type: [is2GraphObject PointType](#)

Punto por el que pasa la línea perpendicular a calcular.

Return Value

Type: [LineType](#)

Devuelve un línea que es perpendicular a la línea "L" dada y que pasa por el punto "P".

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject LineType](#)

[is2GraphObject PointType](#)

is2GraphObj PointInArc Method

Determina si el punto "P" dado pertenece al arco "A".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool PointInArc(  
    PointType P,  
    ArcType A  
)
```

Parameters

P

Type: [is2GraphObject PointType](#)

Punto que se quiere comprobar si pertenece al arco dado.

A

Type: [is2GraphObject ArcType](#)

Arco para el cuál se quiere comprobar la pertenencia de un punto.

Return Value

Type: [Boolean](#)

Devuelve **true** si el punto dado pertenece al arco. En caso contrario devuelve **false**.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject ArcType](#)

[is2GraphObject PointType](#)

is2GraphObj PointInCircle Method

Determina si el punto "P" dado pertenece al circunferencia "C".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool PointInCircle(  
    PointType P,  
    CircleType C  
)
```

Parameters

P

Type: [is2GraphObject PointType](#)

Punto que se quiere comprobar si pertenece a la circunferencia dada.

C

Type: [is2GraphObject CircleType](#)

Circunferencia para la cuál se quiere comprobar la pertenencia de un punto.

Return Value

Type: [Boolean](#)

Devuelve **true** si el punto dado pertenece a la circunferencia. En caso contrario devuelve **false**.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject CircleType](#)

[is2GraphObject PointType](#)

is2GraphObj PointInLine Method

Determina si el punto "P" dado pertenece la línea "L".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool PointInLine(  
    PointType P,  
    LineType L  
)
```

Parameters

P

Type: [is2GraphObject PointType](#)

Punto que se quiere comprobar si pertenece a la línea dada.

L

Type: [is2GraphObject LineType](#)

Línea a la que se quiere comprobar la pertenencia de un punto.

Return Value

Type: [Boolean](#)

Devuelve **true** si el punto dado pertenece a la línea. En caso contrario devuelve **false**.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject LineType](#)

[is2GraphObject PointType](#)

is2GraphObj PointInOctant Method

Determina sobre que octante se ubica el punto "P" dado.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static Octante PointInOctant(  
    PointType P  
)
```

Parameters

P

Type: [is2GraphObject PointType](#)

Punto a comprobar.

Return Value

Type: [Octante](#)

Devuelve el octante en que se ubica el punto dado.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj PointInQuadrant Method

Determina el cuadrante en el que se ubica el punto dado.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static Cuadrante PointInQuadrant(  
    PointType P  
)
```

Parameters

P

Type: [is2GraphObject PointType](#)

Representa el punto a comprobar.

Return Value

Type: [Cuadrante](#)

Devuelve un tipo enum que representa el cuadrante en que se ubica el punto dado.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj PointInSegment Method

Determina si el punto "P" dado pertenece al segmento "S".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool PointInSegment(  
    PointType P,  
    SegmentType S  
)
```

Parameters

P

Type: [is2GraphObject PointType](#)

Punto que se quiere comprobar si pertenece al segmento dada.

S

Type: [is2GraphObject SegmentType](#)

Segmento al cuál se le quiere comprobar la pertenencia de un punto.

Return Value

Type: [Boolean](#)

Devuelve **true** si el punto dado pertenece al segmento. En caso contrario devuelve **false**.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject SegmentType](#)

[is2GraphObject PointType](#)

is2GraphObj PointLineDistance Method

Determina la distancia un punto "P" a una línea "L" dados.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static double PointLineDistance(  
    PointType P,  
    LineType L  
)
```

Parameters

P

Type: [is2GraphObject PointType](#)
Representa el punto.

L

Type: [is2GraphObject LineType](#)
Representa la línea.

Return Value

Type: [Double](#)

Devuelve la distancia de un punto a una línea.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj PointPointAngle Method

Determina el valor del ángulo que se forma entre segmento que describen los puntos "P1" y "P2" dados y el eje de las abcisas (Eje X).

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▀ **Syntax**

C#

[Copy](#)

```
public static double PointPointAngle(  
    PointType P1,  
    PointType P2  
)
```

Parameters

P1

Type: [is2GraphObject PointType](#)
Primer punto.

P2

Type: [is2GraphObject PointType](#)
Segundo punto.

Return Value

Type: [Double](#)

Devuelve el ángulo en grados del segmento que se forma entre los punto "P1" y "P2".

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj PointPointDistance Method

Calcula la distancia entre los puntos "P1" y "P2" dados.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static double PointPointDistance(  
    PointType P1,  
    PointType P2  
)
```

Parameters

P1

Type: [is2GraphObject PointType](#)
Primer punto.

P2

Type: [is2GraphObject PointType](#)
Segundo punto.

Return Value

Type: [Double](#)

Devuelve un double que expresa el valor de la distancia entre dos puntos.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj PointPointSlope Method

Calcula la pendiente de la linea que pasa por los puntos "P1" y "P2" dados.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static double PointPointSlope(  
    PointType P1,  
    PointType P2  
)
```

Parameters

P1

Type: [is2GraphObject PointType](#)

Representa uno de los puntos por donde pasa la línea a la que se le quiere determinar su de pendiente.

P2

Type: [is2GraphObject PointType](#)

Representa el otro punto por donde pasa la línea.

Return Value

Type: [Double](#)

Devuelve un double que expresa el valor de la pendiente de la recta. Devuelve un valor NaN (Not a Number) si ambos puntos tienen iguales la coordenada X, lo que significa que la recta es vertical (es decir, es paralela al eje de las ordenadas).

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj PolarPoint Method

Ubica un punto 2D mediante coordenadas polares partir de un punto base, un ángulo y una distancia.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static PointType PolarPoint(  
    PointType pbase,  
    double angle,  
    double distance  
)
```

Parameters

pbase

Type: [is2GraphObject PointType](#)

Punto de referencia a partir del cuál se calcula el nuevo punto mediante el sistema de coordenadas polares.

angle

Type: [System Double](#)

Componente angular para el sistema de coordenadas polares.

distance

Type: [System Double](#)

Componente distancia para el sistema de coordenadas polares.

Return Value

Type: [PointType](#)

Devuelve el punto ubicado por coordenadas polares.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj PolarToCartesian Method

Convierte coordenadas Polares a coordenadas Cartesianas.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static void PolarToCartesian(  
    double angle,  
    double dist,  
    out double dX,  
    out double dY,  
    PointType pbase = null  
)
```

Parameters

angle

Type: [System Double](#)

Representa el valor del ángulo dado en grados en el sistema de coordenadas polares.

dist

Type: [System Double](#)

Representa el valor de distancia (radio) en el sistema de coordenadas polares.

dX

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia la componente dX del punto.

dY

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia la componente dY del punto.

pbase (**Optional**)

Type: [is2GraphObject PointType](#)

Define el punto base que se usa como origen para calcular las componentes dX y dY. Si "pbase" es igual a null se asume como punto base el Origen del Sistema de Coordenadas (0,0,0).

Return Value

Type:

Devuelve un tipo void. Por referencia se devuelve las componentes dX y dY.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj

PolygonLineIntersect Method

[Missing <summary> documentation for

"M:is2GraphObject.is2GraphObj.PolygonLineIntersect(is2GraphObject.PolygonType,is2G

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public static byte PolygonLineIntersect(  
    PolygonType E,  
    LineType L,  
    out PointType P1,  
    out PointType P2  
)
```

Parameters

E

Type: [is2GraphObject PolygonType](#)

[Missing <param name="E"/> documentation for

"M:is2GraphObject.is2GraphObj.PolygonLineIntersect(is2GraphObject.PolygonType,i

L

Type: [is2GraphObject LineType](#)

[Missing <param name="L"/> documentation for

"M:is2GraphObject.is2GraphObj.PolygonLineIntersect(is2GraphObject.PolygonType,i

P1

Type: [is2GraphObject PointType](#)

[Missing <param name="P1"/> documentation for

"M:is2GraphObject.is2GraphObj.PolygonLineIntersect(is2GraphObject.PolygonType,i

P2

Type: [is2GraphObject PointType](#)

[Missing <param name="P2"/> documentation for

"M:is2GraphObject.is2GraphObj.PolygonLineIntersect(is2GraphObject.PolygonType,i

Return Value

Type: [Byte](#)

[Missing <returns> documentation for

"M:is2GraphObject.is2GraphObj.PolygonLineIntersect(is2GraphObject.PolygonType,is2G

See Also

Reference

[is2GraphObj Class](#)

is2GraphObject Namespace

is2GraphObjRadToGrad Method

Convierte de Radianes a Grados.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public static double RadToGrad(  
    double radian  
)
```

Parameters

radian

Type: [SystemDouble](#)

Valor de ángulo en radianes.

Return Value

Type: [Double](#)

Devuelve un valor de ángulo en grados.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj

RectangleLineIntersect Method

[Missing <summary> documentation for
"M:is2GraphObject.is2GraphObj.RectangleLineIntersect(is2GraphObject.RectangleType,

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public static byte RectangleLineIntersect(  
    RectangleType E,  
    LineType L,  
    out PointType P1,  
    out PointType P2  
)
```

Parameters

E

Type: [is2GraphObject RectangleType](#)

[Missing <param name="E"/> documentation for

"M:is2GraphObject.is2GraphObj.RectangleLineIntersect(is2GraphObject.RectangleTy

L

Type: [is2GraphObject LineType](#)

[Missing <param name="L"/> documentation for

"M:is2GraphObject.is2GraphObj.RectangleLineIntersect(is2GraphObject.RectangleTy

P1

Type: [is2GraphObject PointType](#)

[Missing <param name="P1"/> documentation for

"M:is2GraphObject.is2GraphObj.RectangleLineIntersect(is2GraphObject.RectangleTy

P2

Type: [is2GraphObject PointType](#)

[Missing <param name="P2"/> documentation for

"M:is2GraphObject.is2GraphObj.RectangleLineIntersect(is2GraphObject.RectangleTy

Return Value

Type: [Byte](#)

[Missing <returns> documentation for

"M:is2GraphObject.is2GraphObj.RectangleLineIntersect(is2GraphObject.RectangleTy

See Also

Reference

[is2GraphObj Class](#)

is2GraphObject Namespace

is2GraphObj RootMPQ Method

Calcula las raíces de polinomios de 2do orden que tienen la forma:
 $mx^2 + px + q$ por el método del Discriminante.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static byte RootMPQ(  
    double m,  
    double p,  
    double q,  
    out double r1,  
    out double r2  
)
```

Parameters

m

Type: [System Double](#)

[Missing <param name="m"/> documentation for
"M:is2GraphObject.is2GraphObj.RootMPQ(System.Double,System.Double,System.Do

p

Type: [System Double](#)

[Missing <param name="p"/> documentation for
"M:is2GraphObject.is2GraphObj.RootMPQ(System.Double,System.Double,System.Do

q

Type: [System Double](#)

[Missing <param name="q"/> documentation for
"M:is2GraphObject.is2GraphObj.RootMPQ(System.Double,System.Double,System.Do

r1

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor

r2

Type: [System Double](#)

Parámetro de salida (out). Retorna por referencia el valor

Return Value

Type: [Byte](#)

Devuelve un tipo void. Por referencia mediante los parámetros "r1" y "r2" se obtienen las raíces del polinomio calculadas.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj RotateArc Method

Rota en el plano XY el arco 'A' sobre un punto 'pbase' dado según un ángulo especificado.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static ArcType RotateArc(  
    CircleType C,  
    double angle,  
    PointType pbase = null  
)
```

Parameters

C

Type: [is2GraphObject CircleType](#)

Representa el arco a rotar.

angle

Type: [System Double](#)

Representa el ángulo de rotacion.

pbase (Optional)

Type: [is2GraphObject PointType](#)

Representa el punto sobre el que se hará la rotacion.

Return Value

Type: [ArcType](#)

Devuelve el arco resultante de la rotación.

▲ **Remarks** Nota: Si se omite parámetro 'pbase' se asume que la rotacion se hace sobre el Origen de Coordenadas [0; 0]

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject CircleType](#)

[is2GraphObject PointType](#)

is2GraphObj RotateCircle Method

Rota en el plano XY el circulo 'C' sobre un punto 'pbase' dado según un ángulo especificado.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static CircleType RotateCircle(  
    CircleType C,  
    double angle,  
    PointType pbase = null  
)
```

Parameters

C

Type: [is2GraphObject CircleType](#)

Representa el círculo a rotar.

angle

Type: [System Double](#)

Representa el ángulo de rotacion.

pbase (Optional)

Type: [is2GraphObject PointType](#)

Representa el punto sobre el que se hará la rotacion.

Return Value

Type: [CircleType](#)

Devuelve el círculo resultante de la rotación.

▲ **Remarks** Nota: Si se omite parámetro 'pbase' se asume que la rotacion se hace sobre el Origen de Coordenadas [0; 0]

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject CircleType](#)

[is2GraphObject PointType](#)

is2GraphObj RotatePoint Method

Rota en el plano XY el punto 'P' sobre un punto 'pbase' dado según un ángulo especificado.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static PointType RotatePoint(  
    PointType P,  
    double angle,  
    PointType pbase = null  
)
```

Parameters

P

Type: [is2GraphObject PointType](#)

Representa el punto a rotar.

angle

Type: [System Double](#)

Representa el ángulo de rotacion.

pbase (Optional)

Type: [is2GraphObject PointType](#)

Representa el punto sobre el que se hará la rotacion.

Return Value

Type: [PointType](#)

Devuelve el punto resultante de la rotación.

▲ **Remarks** Nota: Si se omite parámetro 'pbase' se asume que la rotacion se hace sobre el Origen de Coordenadas [0; 0]

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj RotateSegment Method

Rota en el plano XY el segmento 'S' sobre un punto 'pbase' dado según un ángulo especificado.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static SegmentType RotateSegment(  
    SegmentType S,  
    double angle,  
    PointType pbase = null  
)
```

Parameters

S

Type: [is2GraphObject SegmentType](#)

Representa el segmento a rotar.

angle

Type: [System Double](#)

Representa el ángulo de rotacion.

pbase (Optional)

Type: [is2GraphObject PointType](#)

Representa el punto sobre el que se hará la rotacion.

Return Value

Type: [SegmentType](#)

Devuelve el segmento resultante de la rotación.

▲ **Remarks** Nota: Si se omite parámetro 'pbase' se asume que la rotacion se hace sobre el Origen de Coordenadas [0; 0]

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

[is2GraphObject SegmentType](#)

is2GraphObj SegmentsApparentIntersect Method

Determina la intercepción aparente en el plano entre dos segmentos.

Nota: La intersección aparente indica

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool SegmentsApparentIntersect(  
    SegmentType S1,  
    SegmentType S2,  
    out PointType P  
)
```

Parameters

S1

Type: [is2GraphObject SegmentType](#)

Primer segmento.

S2

Type: [is2GraphObject SegmentType](#)

Segundo segmento.

P

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto de intercepción entre los dos segmentos, si existe.

Nota: Si los segmentos no se intersectan por ser colineales o paralelos el parámetro "P" retorna NULL.

Return Value

Type: [Boolean](#)

Devuelve **true** si ambos segmentos se intercertan en el plano. En caso contrario devuelve **false**.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject SegmentType](#)

[is2GraphObject PointType](#)

is2GraphObj

SegmentSegmentAngle Method

Determina el ángulo según su tipo (agudo u obtuso) que se forma entre dos segmentos "S1" y "S2" dadas.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static double SegmentSegmentAngle(  
    SegmentType S1,  
    SegmentType S2,  
    bool agude = true  
)
```

Parameters

S1

Type: [is2GraphObject SegmentType](#)

Primer segmento.

S2

Type: [is2GraphObject SegmentType](#)

Segundo segmento.

agude (Optional)

Type: [System Boolean](#)

Indica el tipo de ángulo que se quiere calcular. Si es **true** se calcula el valor para el ángulo agudo, por el contrario si es **false** se calcula el valor del angulo obtuso.

Return Value

Type: [Double](#)

Devuelve el ángulo en grados (agudo u obtuso) que se forma entre los segmentos dados.

▲ See Also

Reference



[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject SegmentType](#)

is2GraphObjSegmentSegmentCham Method

▲ Overload List

	Name	Description
	SegmentSegmentChamfer(SegmentType, SegmentType, ChamferDistant, ChamferAngle, PointType, PointType)	
	SegmentSegmentChamfer(SegmentType, SegmentType, ChamferDistant, ChamferDistant, PointType, PointType)	

[Top](#)

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj
SegmentSegmentChamfer
Method (SegmentType,
SegmentType, ChamferDistantte,
ChamferAngle, PointType ,
PointType)

[Missing <summary> documentation for
"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.SegmentTyp

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static int SegmentSegmentChamfer(  
    SegmentType S1,  
    SegmentType S2,  
    ChamferDistant dist,  
    ChamferAngle angle,  
    out PointType P1,  
    out PointType P2  
)
```

Parameters

S1

Type: [is2GraphObject SegmentType](#)

[Missing <param name="S1"/> documentation for

"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.Segment"

S2

Type: [is2GraphObject SegmentType](#)

[Missing <param name="S2"/> documentation for

"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.Segment"

dist

Type: [is2GraphObject ChamferDistant](#)

[Missing <param name="dist"/> documentation for

"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.Segment"

angle

Type: [is2GraphObject ChamferAngle](#)

[Missing <param name="angle"/> documentation for

"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.Segment"

P1

Type: [is2GraphObject PointType](#)

[Missing <param name="P1"/> documentation for

"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.Segment"

P2

Type: [is2GraphObject PointType](#)

[Missing <param name="P2"/> documentation for

"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.Segment"

Return Value

Type: [Int32](#)

[Missing <returns> documentation for

"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.SegmentTyp

▲ See Also

Reference

[is2GraphObj Class](#)

[SegmentSegmentChamfer Overload](#)

[is2GraphObject Namespace](#)

is2GraphObj
SegmentSegmentChamfer
Method (SegmentType,
SegmentType, ChamferDistanto,
ChamferDistanto, PointType ,
PointType)

[Missing <summary> documentation for
"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.SegmentTyp

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public static int SegmentSegmentChamfer(  
    SegmentType S1,  
    SegmentType S2,  
    ChamferDistante dist1,  
    ChamferDistante dist2,  
    out PointType P1,  
    out PointType P2  
)
```

Parameters

S1

Type: [is2GraphObject SegmentType](#)

[Missing <param name="S1"/> documentation for

"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.Segment"

S2

Type: [is2GraphObject SegmentType](#)

[Missing <param name="S2"/> documentation for

"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.Segment"

dist1

Type: [is2GraphObject ChamferDistante](#)

[Missing <param name="dist1"/> documentation for

"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.Segment"

dist2

Type: [is2GraphObject ChamferDistante](#)

[Missing <param name="dist2"/> documentation for

"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.Segment"

P1

Type: [is2GraphObject PointType](#)

[Missing <param name="P1"/> documentation for

"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.Segment"

P2

Type: [is2GraphObject PointType](#)

[Missing <param name="P2"/> documentation for

"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.Segment"

Return Value

Type: [Int32](#)

[Missing <returns> documentation for

"M:is2GraphObject.is2GraphObj.SegmentSegmentChamfer(is2GraphObject.SegmentTyp

▲ See Also

Reference

[is2GraphObj Class](#)



[SegmentSegmentChamfer Overload](#)

[is2GraphObject Namespace](#)

is2GraphObj

SegmentSegmentFillet Method

▲ Overload List

	Name	Description
	SegmentSegmentFillet(SegmentType, SegmentType, Double, Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos segmentos de rectas "S1" y "S2" dados, con radio determinado por "r_fillet".
	SegmentSegmentFillet(SegmentType, SegmentType, Double, PointType , PointType , Boolean, Boolean)	Calcula el fillet (EMPALME) entre dos segmentos de rectas "S1" y "S2" dados, con radio determinado por "r_fillet".

[Top](#)

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj

SegmentSegmentFillet Method

(SegmentType, SegmentType, Double, Boolean, Boolean)

Calcula el fillet (EMPALME) entre dos segmentos de rectas "S1" y "S2" dados, con radio determinado por "r_fillet".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static ArcType SegmentSegmentFillet(  
    SegmentType S1,  
    SegmentType S2,  
    double r_fillet,  
    bool s1_right_up = true,  
    bool s2_right_up = true  
)
```

Parameters

S1

Type: [is2GraphObject SegmentType](#)

Representa el 1er segmento.

S2

Type: [is2GraphObject SegmentType](#)

Representa el 2do segmento.

r_fillet

Type: [System Double](#)

Representa el radio del fillet.

s1_right_up (Optional)

Type: [System Boolean](#)

Determina hacia que lado del segmento "S1" se calcula el fillet.

Nota: Si el valor es **true**, el fillet se calcula hacia la derecha-o-arriba del segmento. Por el contrario si el valor es **false**, el fillet se calcula hacia la izquierda-o-abajo del segmento. En ambos casos su aplicación es sobre el segmento "S1".

s2_right_up (Optional)

Type: [System Boolean](#)

Determina hacia que lado del segmento "S2" se calcula el fillet.

Nota: Si el valor es **true**, el fillet se calcula hacia la derecha-o-arriba del segmento. Por el contrario si el valor es **false**, el fillet se calcula hacia la izquierda-o-abajo del segmento. En ambos casos su aplicación es sobre el segmento "S2".

Return Value

Type: [ArcType](#)

Devuelve un tipo Arco que representa el fillet entre los dos segmentos dados.

▲ See Also

Reference

[is2GraphObj Class](#)

[SegmentSegmentFillet Overload](#)

[is2GraphObject Namespace](#)

[is2GraphObject SegmentType](#)

is2GraphObj

SegmentSegmentFillet Method (SegmentType, SegmentType, Double, PointType , PointType , PointType , Boolean, Boolean)

Calcula el fillet (EMPALME) entre dos segmentos de rectas "S1" y "S2" dados, con radio determinado por "r_fillet".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool SegmentSegmentFillet(  
    SegmentType S1,  
    SegmentType S2,  
    double r_fillet,  
    out PointType P1,  
    out PointType P2,  
    out PointType Pc,  
    bool s1_right_up = true,  
    bool s2_right_up = true  
)
```

Parameters

S1

Type: [is2GraphObject SegmentType](#)
Representa el 1er segmento.

S2

Type: [is2GraphObject SegmentType](#)
Representa el 2do segmento.

r_fillet

Type: [System Double](#)
Representa el radio del fillet.

P1

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia el punto del arc-fillet que pertenece al segmento "S1".

P2

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia el punto del arc-fillet que pertenece al segmento "S2".

Pc

Type: [is2GraphObject PointType](#)
Parámetro de salida (out). Retorna por referencia el punto centro del arc-fillet.

s1_right_up (Optional)

Type: [System Boolean](#)

Determina hacia que lado del segmento "S1" se calcula el fillet.

Si el valor es **true**, el fillet se calcula hacia la derecha-o-arriba del segmento. Por el contrario si el valor es **false**, el fillet se calcula hacia la izquierda-o-abajo del segmento. En ambos casos su aplicación es sobre el segmento "S1".

s2_right_up (Optional)

Type: [System Boolean](#)

Determina hacia que lado del segmento "S2" se calcula el fillet.

Nota: Si el valor es **true**, el fillet se calcula hacia la derecha-o-arriba del segmento. Por el contrario si el valor es **false**, el fillet se calcula hacia la izquierda-o-abajo del segmento. En ambos casos su aplicación es sobre el segmento "S2".

Return Value

Type: [Boolean](#)

Nota: Devuelve un tipo void. Por referencia mediante los parámetros P1, P2 y Pc se obtienen los 3 puntos del arc-fillet.

▲ **Remarks** Nota: "P1" pertenece al segmento "S1", "P2" pertenece al segmento "S2" y "Pc" es el centro del arco.

▲ See Also

Reference

[is2GraphObj Class](#)

[SegmentSegmentFillet Overload](#)

[is2GraphObject Namespace](#)

[is2GraphObject SegmentType](#)

[is2GraphObject PointType](#)

is2GraphObj SegmentSegmentFilletSave Method

[Missing <summary> documentation for
"M:is2GraphObject.is2GraphObj.SegmentSegmentFilletSave(is2GraphObject.SegmentTy

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public static ArcType SegmentSegmentFilletSave(  
    ref SegmentType S1,  
    ref SegmentType S2,  
    double r_fillet,  
    bool s1_right_up = true,  
    bool s2_right_up = true  
)
```

Parameters

S1

Type: [is2GraphObject SegmentType](#)

[Missing <param name="S1"/> documentation for
"M:is2GraphObject.is2GraphObj.SegmentSegmentFilletSave(is2GraphObject.Segmen

S2

Type: [is2GraphObject SegmentType](#)

[Missing <param name="S2"/> documentation for
"M:is2GraphObject.is2GraphObj.SegmentSegmentFilletSave(is2GraphObject.Segmen

r_fillet

Type: [System Double](#)

[Missing <param name="r_fillet"/> documentation for
"M:is2GraphObject.is2GraphObj.SegmentSegmentFilletSave(is2GraphObject.Segmen

s1_right_up (Optional)

Type: [System Boolean](#)

[Missing <param name="s1_right_up"/> documentation for
"M:is2GraphObject.is2GraphObj.SegmentSegmentFilletSave(is2GraphObject.Segmen

s2_right_up (Optional)

Type: [System Boolean](#)

[Missing <param name="s2_right_up"/> documentation for
"M:is2GraphObject.is2GraphObj.SegmentSegmentFilletSave(is2GraphObject.Segmen

Return Value

Type: [ArcType](#)

[Missing <returns> documentation for
"M:is2GraphObject.is2GraphObj.SegmentSegmentFilletSave(is2GraphObject.SegmentTy

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj

SegmentsRealIntersect Method

Determina la intercepción real en el plano entre dos segmentos.

Nota: La intersección real indica que ambos segmentos tienen realmente un punto en común.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static bool SegmentsRealIntersect(  
    SegmentType S1,  
    SegmentType S2,  
    out PointType P  
)
```

Parameters

S1

Type: [is2GraphObject SegmentType](#)

Primer segmento.

S2

Type: [is2GraphObject SegmentType](#)

Segundo segmento.

P

Type: [is2GraphObject PointType](#)

Parámetro de salida (out). Retorna por referencia el punto de intercepción entre los dos segmentos, si existe.

Nota: Si los segmentos no se intersectan por ser colineales o paralelos entre si, el parámetro "P" retorna NULL.

Return Value

Type: [Boolean](#)

Devuelve **true** si ambos segmentos se intercertan realmente en el plano. En caso contrario devuelve **false**.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject SegmentType](#)

[is2GraphObject PointType](#)

is2GraphObj SetPresicion Method

Establece la precision (posiciones decimales) que tiene en cuenta is2Graph para comprobar el valor del Cero Real.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static void SetPresicion(  
    uint d  
)
```

Parameters

d

Type: [System UInt32](#)

Es un número entero entre 1 - 10 que establece la precisión sobre valores decimales con la que trabajara Is2Graph.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObjSphericalPoint Method

Ubica un punto 3D mediante coordenadas esféricas.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public static PointType SphericalPoint(  
    PointType pbase  
)
```

Parameters

pbase

Type: [is2GraphObjectPointType](#)

[Missing <param name="pbase"/> documentation for
"M:is2GraphObject.is2GraphObj.SphericalPoint(is2GraphObject.PointType)"]

Return Value

Type: [PointType](#)

[Missing <returns> documentation for
"M:is2GraphObject.is2GraphObj.SphericalPoint(is2GraphObject.PointType)"]

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObjectPointType](#)

is2GraphObj SwapArc Method

Intercambia los dos arcos pasados por parámetro.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static void SwapArc(  
    ref ArcType A1,  
    ref ArcType A2  
)
```

Parameters

A1

Type: [is2GraphObject ArcType](#)

Parámetro de salida (ref). Primer arco.

A2

Type: [is2GraphObject ArcType](#)

Parámetro de salida (ref). Segundo arco.

Return Value

Type:

La función retorna un tipo void. Por referencia quedan intercambiados ambos arcos.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject ArcType](#)

is2GraphObj SwapCircle Method

Intercambia las dos circunferencias pasados por parámetro.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static void SwapCircle(  
    ref CircleType C1,  
    ref CircleType C2  
)
```

Parameters

C1

Type: [is2GraphObject CircleType](#)

Parámetro de salida (ref). Primera circunferencia.

C2

Type: [is2GraphObject CircleType](#)

Parámetro de salida (ref). Segunda circunferencia.

Return Value

Type:

La función retorna un tipo void. Por referencia quedan intercambiados ambos círculos.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject CircleType](#)

is2GraphObj SwapLine Method

Intercambia las dos líneas pasados por parámetro.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static void SwapLine(  
    ref LineType L1,  
    ref LineType L2  
)
```

Parameters

L1

Type: [is2GraphObject LineType](#)

Parámetro de salida (ref). Primera línea.

L2

Type: [is2GraphObject LineType](#)

Parámetro de salida (ref). Segunda línea.

Return Value

Type:

La función retorna un tipo void. Por referencia quedan intercambiados ambas lineas.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject LineType](#)

is2GraphObj SwapPoint Method

Intercambia los dos puntos pasados por parámetro.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static void SwapPoint(  
    ref PointType P1,  
    ref PointType P2  
)
```

Parameters

P1

Type: [is2GraphObject PointType](#)

Parámetro de salida (ref). Primer punto.

P2

Type: [is2GraphObject PointType](#)

Parámetro de salida (ref). Segundo punto.

Return Value

Type:

La función retorna un tipo void. Por referencia quedan intercambiados ambos puntos.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj SwapSegment Method

Intercambia los dos segmentos pasados por parámetro.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static void SwapSegment(  
    ref SegmentType S1,  
    ref SegmentType S2  
)
```

Parameters

S1

Type: [is2GraphObject SegmentType](#)

Parámetro de salida (ref). Primer segmento.

S2

Type: [is2GraphObject SegmentType](#)

Parámetro de salida (ref). Segundo segmento.

Return Value

Type:

La función retorna un tipo void. Por referencia quedan intercambiados ambos segmentos.

See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject SegmentType](#)

is2GraphObj SwapValue Method

Intercambia los dos valores pasados por parámetro.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public static void SwapValue(  
    ref double a,  
    ref double b  
)
```

Parameters

a

Type: [System Double](#)

Parámetro de salida (ref). Primer valor decimal.

b

Type: [System Double](#)

Parámetro de salida (ref). Segundo valor decimal.

Return Value

Type:

La función retorna un tipo void. Por referencia quedan intercambiados ambos valores.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObj TranslateArc Method

Traslada el arco 'A' las distancias definas por las componentes dx, dy, dz.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static ArcType TranslateArc(  
    ArcType A,  
    double dx,  
    double dy = 0,  
    double dz = 0  
)
```

Parameters

A

Type: [is2GraphObject ArcType](#)
Representa el arco a trasladar.

dx

Type: [System Double](#)
Representa la componente de la translacion sobre el eje X.

dy (**Optional**)

Type: [System Double](#)
Representa la componente de la translacion sobre el eje Y.

dz (**Optional**)

Type: [System Double](#)
Representa la componente de la translacion sobre el eje Z.

Return Value

Type: [ArcType](#)
Devuelve el arco resultante de la translacion sobre las componentes dx, dy, dz.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject ArcType](#)

is2GraphObj TranslateCircle Method

Traslada el circulo 'C' las distancias definas por las componentes dx, dy, dz.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static CircleType TranslateCircle(  
    CircleType C,  
    double dx,  
    double dy = 0,  
    double dz = 0  
)
```

Parameters

C

Type: [is2GraphObject CircleType](#)

Representa el circulo a trasladar.

dx

Type: [System Double](#)

Representa la componente de la translacion sobre el eje X.

dy (**Optional**)

Type: [System Double](#)

Representa la componente de la translacion sobre el eje Y.

dz (**Optional**)

Type: [System Double](#)

Representa la componente de la translacion sobre el eje Z.

Return Value

Type: [CircleType](#)

Devuelve el circulo resultante de la translacion sobre las componentes dx, dy, dz.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject CircleType](#)

is2GraphObj TranslatePoint Method

Traslada el punto 'P' las distancias definidas por las componentes dx, dy, dz.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static PointType TranslatePoint(  
    PointType P,  
    double dx,  
    double dy = 0,  
    double dz = 0  
)
```

Parameters

P

Type: [is2GraphObject PointType](#)

Representa el punto a trasladar.

dx

Type: [System Double](#)

Representa la componente de la translacion sobre el eje X.

dy (**Optional**)

Type: [System Double](#)

Representa la componente de la translacion sobre el eje Y.

dz (**Optional**)

Type: [System Double](#)

Representa la componente de la translacion sobre el eje Z.

Return Value

Type: [PointType](#)

Devuelve el punto resultante de la translacion sobre las componentes dx, dy, dz.

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

is2GraphObj TranslateSegment Method

Traslada el segmento 'S' las distancias definas por las componentes dx, dy, dz.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public static SegmentType TranslateSegment(  
    SegmentType S,  
    double dx,  
    double dy = 0,  
    double dz = 0  
)
```

Parameters

S

Type: [is2GraphObject SegmentType](#)

Representa el segmento a trasladar.

dx

Type: [System Double](#)

Representa la componente de la translacion sobre el eje X.

dy (**Optional**)

Type: [System Double](#)

Representa la componente de la translacion sobre el eje Y.

dz (**Optional**)

Type: [System Double](#)

Representa la componente de la translacion sobre el eje Z.

Return Value

Type: [SegmentType](#)

Devuelve el segmento resultante de la translacion sobre las componentes dx, dy, dz.

▲ See Also

Reference

[is2GraphObj Class](#)


[is2GraphObject Namespace](#)

[is2GraphObject SegmentType](#)

is2GraphObj Properties

The [is2GraphObj](#) type exposes the following members.

▲ Properties

	Name	Description
	OrigenXYZ	Propiedad de solo lectura. Devuelve el Origen de Sistema de Coordenadas.

[Top](#)

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

is2GraphObjOrigenXYZ Property

Propiedad de solo lectura. Devuelve el Origen de Sistema de Coordenadas.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public static PointType OrigenXYZ { get; }
```

Property Value

Type: [PointType](#)

▲ See Also

Reference

[is2GraphObj Class](#)

[is2GraphObject Namespace](#)

LineType Class

Representa un tipo Línea. Esta línea se define por un punto y un ángulo, dando como resultado una línea que pasa por el punto 'P', forma un angulo con el eje de las abcisas determinado por 'Angle' y tiene longitud infinita.

▲ Inheritance Hierarchy [System Object](#) [is2GraphObject](#)
[LineType](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public class LineType
```





The `LineType` type exposes the following members.



▲ Constructors

	Name	Description
	<code>LineType</code>	Constructor por defecto.
	<code>LineType(PointType, Double)</code>	Contructor que toma por parámetro un punto y un ángulo.

[Top](#)


▲ Methods

	Name	Description
	<code>Equals</code>	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	<code>Finalize</code>	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	<code>GetHashCode</code>	Serves as the default hash function. (Inherited from Object .)
	<code>GetType</code>	Gets the Type of the current instance. (Inherited from Object .)

	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)


[Top](#)

▲ Fields

	Name	Description
	P	Representa el punto por donde pasa la línea.

[Top](#)

▲ Properties

	Name	Description
	Angle	Representa el ángulo de inclinación de la línea con respecto al eje de las abcisas (Eje X).

[Top](#)

▲ See Also

Reference

[is2GraphObject Namespace](#)

LineType Constructor

▲ Overload List

	Name	Description
	LineType	Constructor por defecto.
	LineType(PointType, Double)	Contructor que toma por parámetro un punto y un ángulo.

[Top](#)

▲ See Also

Reference

[LineType Class](#)

[is2GraphObject Namespace](#)

LineType Constructor

Constructor por defecto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public LineType()
```

▲ See Also

Reference

[LineType Class](#)

[LineType Overload](#)

[is2GraphObject Namespace](#)

LineType Constructor (PointType, Double)

Contructor que toma por parámetro un punto y un ángulo.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public LineType(  
    PointType p,  
    double angle  
)
```

Parameters

p

Type: [is2GraphObject PointType](#)

Punto por donde pasa la línea.

angle

Type: [System Double](#)

Ángulo de inclinación de la línea con respecto al eje de las abscisas (Eje X).

▲ See Also

Reference

[LineType Class](#)


[LineType Overload](#)

[is2GraphObject Namespace](#)

LineType Fields

The [LineType](#) type exposes the following members.

▲ Fields

	Name	Description
	P	Representa el punto por donde pasa la línea.

[Top](#)

▲ See Also

Reference

[LineType Class](#)

[is2GraphObject Namespace](#)

LineTypeP Field

Representa el punto por donde pasa la línea.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType P
```

Field Value

Type: [PointType](#)

▲ See Also

Reference







[LineType Class](#)

[is2GraphObject Namespace](#)

LineType Methods

The [LineType](#) type exposes the following members.

▲ Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

[Top](#)

▲ See Also

Reference


[LineType Class](#)

[is2GraphObject Namespace](#)

LineType Properties

The [LineType](#) type exposes the following members.

▲ Properties

	Name	Description
	Angle	Representa el ángulo de inclinación de la línea con respecto al eje de las abcisas (Eje X).

[Top](#)

▲ See Also

Reference

[LineType Class](#)

[is2GraphObject Namespace](#)

LineTypeAngle Property

Representa el ángulo de inclinación de la línea con respecto al eje de las abcisas (Eje X).

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Angle { get; set; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[LineType Class](#)

[is2GraphObject Namespace](#)

Octante Enumeration

Define cada una de las ocho porciones en la que descompone un sistema de 3 planos interceptados en el espacio (Frontal, Horizontal, Lateral).

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public enum Octante
```

▲ Members

	Member name	Value	Description
	I	1	Representa el 1er octante [+X +Y +Z] para un sistema de 3 planos.
	II	2	Representa el 2do octante [+X -Y +Z] para un sistema de 3 planos.
	III	3	Representa el 3ro octante [+X -Y -Z] para un sistema de 3 planos.
	IV	4	Representa el 4to octante [+X +Y -Z] para un sistema de 3 planos.
	V	5	Representa el 5to octante [-X +Y +Z] para un sistema de 3 planos.
	VI	6	Representa el 6to octante [-X -Y +Z] para un sistema de 3 planos.
	VII	7	Representa el 7mo octante [-X -Y -Z] para un sistema de 3 planos.
	VIII	8	Representa el 8vo octante [-X +Y -Z] para un sistema de 3 planos.

▲ See Also

Reference

[is2GraphObject Namespace](#)

Plane Enumeration

Define los 3 planos que se forman por la intersección de los ejes de coordenadas X-Y, Y-Z y X-Z.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public enum Plane
```

▲ Members

Member name	Value	Description
XY	0	Representa el plano XY
YZ	1	Representa el plano YZ
XZ	2	Representa el plano XZ

▲ See Also

Reference

[is2GraphObject Namespace](#)

PlaneType Class

Representa un tipo Plano.

▲ Inheritance Hierarchy [System Object](#) [is2GraphObject](#)
[PlaneType](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax


C#

[Copy](#)

```
public class PlaneType
```






The `PlaneType` type exposes the following members.

▲ Constructors

Name	Description
 PlaneType	

[Top](#)

▲ Methods

Name	Description
 Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
 Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
 GetHashCode	Serves as the default hash function. (Inherited from Object .)
 GetType	Gets the Type of the current instance. (Inherited from Object .)
 MemberwiseClone	Creates a shallow copy of the current Object .

(Inherited from [Object](#).)



[ToString](#)

Returns a string that represents the current object.
(Inherited from [Object](#).)

[Top](#)

▲ Fields

	Name	Description
	Angle	Representa el angulo que forma ??.
	DirVector	Define un vector dirección en el plano.
	OnePoint	Define un punto en el plano.

[Top](#)

▲ See Also

Reference

[is2GraphObject Namespace](#)

PlaneType Constructor

[Missing <summary> documentation for "M:is2GraphObject.PlaneType.#ctor"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PlaneType()
```

▲ See Also

Reference




[PlaneType Class](#)

[is2GraphObject Namespace](#)

PlaneType Fields

The [PlaneType](#) type exposes the following members.

▲ Fields

	Name	Description
	Angle	Representa el angulo que forma ??.
	DirVector	Define un vector dirección en el plano.
	OnePoint	Define un punto en el plano.

[Top](#)

▲ See Also

Reference

[PlaneType Class](#)

[is2GraphObject Namespace](#)

PlaneTypeAngle Field

Representa el angulo que forma ??.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Angle
```

Field Value

Type: [Double](#)

▲ See Also

Reference

[PlaneType Class](#)

[is2GraphObject Namespace](#)

PlaneTypeDirVector Field

Define un vector dirección en el plano.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType DirVector
```

Field Value

Type: [PointType](#)

▲ See Also

Reference

[PlaneType Class](#)

[is2GraphObject Namespace](#)

PlaneTypeOnePoint Field

Define un punto en el plano.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType OnePoint
```

Field Value

Type: [PointType](#)

▲ See Also

Reference







[PlaneType Class](#)

[is2GraphObject Namespace](#)

PlaneType Methods

The [PlaneType](#) type exposes the following members.

▲ Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

[Top](#)

▲ See Also

Reference

[PlaneType Class](#)

[is2GraphObject Namespace](#)

PointLinePosition Enumeration

[Missing <summary> documentation for "T:is2GraphObject.PointLinePosition"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public enum PointLinePosition
```

▲ Members

	Member name	Value	Description
	Member	0	Indica que el punto pertenece a la línea
	Up	1	Indica que el punto está sobre la línea.
	Left	2	Indica que el punto está a la izquierda de la línea.
	Bottom	4	Indica que el punto está debajo de la línea.
	Right	8	Indica que el punto está a la derecha de la línea.

▲ See Also

Reference

[is2GraphObject Namespace](#)

PointType Class

Representa un tipo Punto que puede ser definido tanto en el plano como el espacio.

▲ Inheritance Hierarchy [System Object](#) [is2GraphObject](#)
[PointType](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax




C#

Copy

```
public class PointType
```




The `PointType` type exposes the following members.

Constructors

	Name	Description
	<code>PointType</code>	Constructor por defecto.
	<code>PointType(Double, Double)</code>	Contructor que toma 2 parámetros, [x, y].
	<code>PointType(Double, Double, Double)</code>	Contructor que toma 3 parámetros, [x, y, z].

[Top](#)

Methods

	Name	Description
	<code>Equals</code>	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	<code>Finalize</code>	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	<code>GetHashCode</code>	Serves as the default hash function. (Inherited from Object .)



	GetType	Gets the Type of the current instance. (Inherited from Object .)
💡💜	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
☰💜	ToString	Returns a string that represents the current object. (Inherited from Object .)

[Top](#)

▲ Fields

	Name	Description
💎	cX	Representa la coordenada X del punto.
💎	cY	Representa la coordenada Y del punto.
💎	cZ	Representa la coordenada Z del punto.

[Top](#)




▲ See Also

Reference

[is2GraphObject Namespace](#)

PointType Constructor

▲ Overload List

	Name	Description
	PointType	Constructor por defecto.
	PointType(Double, Double)	Constructor que toma 2 parámetros, [x, y].
	PointType(Double, Double, Double)	Constructor que toma 3 parámetros, [x, y, z].

[Top](#)

▲ See Also

Reference

[PointType Class](#)

[is2GraphObject Namespace](#)

PointType Constructor

Constructor por defecto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType()
```

▲ See Also

Reference

[PointType Class](#)

[PointType Overload](#)

[is2GraphObject Namespace](#)

PointType Constructor (Double, Double)

Contructor que toma 2 parámetros, [x, y].

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType(  
    double x,  
    double y  
)
```

Parameters

x

Type: [SystemDouble](#)

Coordenada X del punto.

y

Type: [SystemDouble](#)

Coordenada Y del punto.

▲ See Also

Reference

[PointType Class](#)

[PointType Overload](#)

[is2GraphObject Namespace](#)

PointType Constructor (Double, Double, Double)

Contructor que toma 3 parámetros, [x, y, z].

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType(  
    double x,  
    double y,  
    double z  
)
```

Parameters

x

Type: [System Double](#)

Coordenada X del punto.

y

Type: [System Double](#)

Coordenada Y del punto.

z

Type: [System Double](#)

Coordenada Z del punto.

▲ See Also

Reference

[PointType Class](#)




[PointType Overload](#)

[is2GraphObject Namespace](#)

PointType Fields

The [PointType](#) type exposes the following members.

▲ Fields

	Name	Description
	cX	Representa la coordenada X del punto.
	cY	Representa la coordenada Y del punto.
	cZ	Representa la coordenada Z del punto.

[Top](#)

▲ See Also

Reference

[PointType Class](#)

[is2GraphObject Namespace](#)

PointTypecX Field

Representa la coordenada X del punto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double cX
```

Field Value

Type: [Double](#)

▲ See Also

Reference

[PointType Class](#)

[is2GraphObject Namespace](#)

PointTypecY Field

Representa la coordenada Y del punto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double cY
```

Field Value

Type: [Double](#)

▲ See Also

Reference

[PointType Class](#)

[is2GraphObject Namespace](#)

PointTypecZ Field

Representa la coordenada Z del punto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double cZ
```

Field Value

Type: [Double](#)

▲ See Also

Reference







[PointType Class](#)

[is2GraphObject Namespace](#)

PointType Methods

The [PointType](#) type exposes the following members.

▲ Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

[Top](#)

▲ See Also

Reference

[PointType Class](#)

[is2GraphObject Namespace](#)

PolarPointType Class

Representa un tipo Punto Polar el cual es definido mediante coordenadas polares.

▲ Inheritance Hierarchy [System Object](#) [is2GraphObject](#)
[PolarPointType](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax



C#

[Copy](#)

```
public class PolarPointType
```





The **PolarPointType** type exposes the following members.

Constructors



	Name	Description
	PolarPointType	Constructor por defecto.
	PolarPointType(PointType, Double, Double)	Constructor que toma 3 parámetro, [punto base, ángulo y radio].

[Top](#)

Methods



	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance.

(Inherited from [Object](#).)

	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)





[Top](#)

Fields

	Name	Description
	Angle	Representa la componente angular del punto polar.
	Radius	Representa la componente de longitud del punto polar.

[Top](#)

Properties

	Name	Description
	cX	Propiedad de solo lectura. Devuelve la coordenada X del punto polar.
	cY	Propiedad de solo lectura. Devuelve la coordenada Y del punto polar.
	cZ	Propiedad de solo lectura. Devuelve la coordenada Z del punto.
	ReferencePoint	Propiedad de solo lectura. Devuelve el punto de referencia que se uso como Centro u Origen para calcular el punto

polar.

[Top](#)



▲ [See Also](#)

Reference

[is2GraphObject Namespace](#)

PolarPointType Constructor

▲ Overload List

	Name	Description
	PolarPointType	Constructor por defecto.
	PolarPointType(PointType, Double, Double)	Constructor que toma 3 parámetro, [punto base, ángulo y radio].

[Top](#)

▲ See Also

Reference

[PolarPointType Class](#)

[is2GraphObject Namespace](#)

PolarPointType Constructor

Constructor por defecto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PolarPointType()
```

▲ See Also

Reference

[PolarPointType Class](#)

[PolarPointType Overload](#)

[is2GraphObject Namespace](#)

PolarPointType Constructor (PointType, Double, Double)

Constructor que toma 3 parámetro, [punto base, ángulo y radio].

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public PolarPointType(  
    PointType pbase,  
    double angle,  
    double radius  
)
```

Parameters

pbase

Type: [is2GraphObject PointType](#)

Punto base a partir del cuál se calcula el punto polar.

angle

Type: [System Double](#)

Componente angular del punto polar.

radius

Type: [System Double](#)

Componente de longitud del punto polar.

▲ **Remarks** El valor del radio se considera siempre positivo, por lo que establecer un valor de radio negativo no tiene ninguna influencia en el punto polar calculado.

▲ **See Also**

Reference

[PolarPointType Class](#)



[PolarPointType Overload](#)

[is2GraphObject Namespace](#)

PolarPointType Fields

The [PolarPointType](#) type exposes the following members.

▲ Fields

	Name	Description
	Angle	Representa la componente angular del punto polar.
	Radius	Representa la componente de longitud del punto polar.

[Top](#)

▲ See Also

Reference

[PolarPointType Class](#)

[is2GraphObject Namespace](#)

PolarPointTypeAngle Field

Representa la componente angular del punto polar.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Angle
```

Field Value

Type: [Double](#)

▲ See Also

Reference

[PolarPointType Class](#)

[is2GraphObject Namespace](#)

PolarPointTypeRadius Field

Representa la componente de longitud del punto polar.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Radius
```

Field Value

Type: [Double](#)

▲ See Also

Reference







[PolarPointType Class](#)

[is2GraphObject Namespace](#)

PolarPointType Methods

The [PolarPointType](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

[Top](#)

▲ See Also

Reference





[PolarPointType Class](#)

[is2GraphObject Namespace](#)

PolarPointType Properties

The [PolarPointType](#) type exposes the following members.

▲ Properties

	Name	Description
	cX	Propiedad de solo lectura. Devuelve la coordenada X del punto polar.
	cY	Propiedad de solo lectura. Devuelve la coordenada Y del punto polar.
	cZ	Propiedad de solo lectura. Devuelve la coordenada Z del punto.
	ReferencePoint	Propiedad de solo lectura. Devuelve el punto de referencia que se uso como Centro u Origen para calcular el punto polar.

[Top](#)

▲ See Also

Reference

[PolarPointType Class](#)

[is2GraphObject Namespace](#)

PolarPointTypecX Property

Propiedad de solo lectura. Devuelve la coordenada X del punto polar.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double cX { get; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[PolarPointType Class](#)

[is2GraphObject Namespace](#)

PolarPointType.cY Property

Propiedad de solo lectura. Devuelve la coordenada Y del punto polar.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double cY { get; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[PolarPointType Class](#)

[is2GraphObject Namespace](#)

PolarPointTypecZ Property

Propiedad de solo lectura. Devuelve la coordenada Z del punto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double cZ { get; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[PolarPointType Class](#)

[is2GraphObject Namespace](#)

PolarPointTypeReferencePoint Property

Propiedad de solo lectura. Devuelve el punto de referencia que se uso como Centro u Origen para calcular el punto polar.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType ReferencePoint { get; }
```

Property Value

Type: [PointType](#)

▲ See Also

Reference

[PolarPointType Class](#)

[is2GraphObject Namespace](#)

PolygonException Class

Representa una exception de tipo Polygon Error.

▲ Inheritance Hierarchy [System Object System](#)

[Exception](#)

[is2GraphObject PolygonException](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**


C#

[Copy](#)

```
public class PolygonException : Exception
```





The `PolygonException` type exposes the following members.

▲ Constructors





	Name	Description
	PolygonException	Constructor por defecto.
	PolygonException(String)	Representa una exception de tipo Arc Error.

[Top](#)

▲ Methods




	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash






function.
(Inherited from [Object](#).)

	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

[Top](#)


▲ Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)

	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

[Top](#)

▲ Events

Name		Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

[Top](#)

▲ See Also

Reference

is2GraphObject Namespace

PolygonException Constructor

▲ Overload List

	Name	Description
	PolygonException	Constructor por defecto.
	PolygonException(String)	Representa una exception de tipo Arc Error.

[Top](#)

▲ See Also

Reference

[PolygonException Class](#)

[is2GraphObject Namespace](#)

PolygonException Constructor

Constructor por defecto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PolygonException()
```

▲ See Also

Reference

[PolygonException Class](#)

[PolygonException Overload](#)

[is2GraphObject Namespace](#)

PolygonException Constructor (String)

Representa una exception de tipo Arc Error.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PolygonException(  
    string msg  
)
```

Parameters

msg

Type: [SystemString](#)

[Missing <param name="msg"/> documentation for
"M:is2GraphObject.PolygonException.#ctor(System.String)"]

▲ See Also

Reference

[PolygonException Class](#)

[PolygonException Overload](#)







[is2GraphObject Namespace](#)

[SystemString](#)

PolygonException Methods

The [PolygonException](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current

instance.
(Inherited from [Exception.](#))



[MemberwiseClone](#)

Creates a shallow copy of the current [Object](#).
(Inherited from [Object.](#))



[ToString](#)

Creates and returns a string representation of the current exception.
(Inherited from [Exception.](#))

[Top](#)

▲ See Also

Reference








[PolygonException Class](#)

[is2GraphObject Namespace](#)

PolygonException Properties

The [PolygonException](#) type exposes the following members.

▲ Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the

immediate frames on the call stack.
(Inherited from [Exception](#).)



[TargetSite](#)

Gets the method that throws the current exception.
(Inherited from [Exception](#).)

[Top](#)

▲ See Also

Reference


[PolygonException Class](#)

[is2GraphObject Namespace](#)

PolygonException Events

The [PolygonException](#) type exposes the following members.

▲ Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

[Top](#)

▲ See Also

Reference

[PolygonException Class](#)

[is2GraphObject Namespace](#)

PolygonType Class

Representa un tipo Polígono regular convexo.

▲ Inheritance Hierarchy [System Object](#) [is2GraphObject](#)
[PolygonType](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax










C#

Copy

```
public class PolygonType
```

The `PolygonType` type exposes the following members.

Methods


	Name	Description
	Area	
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	isPointInside	
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Perimeter	
		

[ToString](#)

Returns a string that represents the current object.
(Inherited from [Object](#).)


[Top](#)

▲ Fields

Name		Description
	Center	

[Top](#)

▲ Properties

Name		Description
	Radius	

[Top](#)

▲ See Also


Reference

[is2GraphObject Namespace](#)

PolygonType Fields

The [PolygonType](#) type exposes the following members.

▲ Fields

Name		Description
	Center	

[Top](#)

▲ See Also

Reference

[PolygonType Class](#)

[is2GraphObject Namespace](#)

PolygonTypeCenter Field

[Missing <summary> documentation for "F:is2GraphObject.PolygonType.Center"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType Center
```

Field Value

Type: [PointType](#)

▲ See Also

Reference









[PolygonType Class](#)

[is2GraphObject Namespace](#)

PolygonType Methods

The [PolygonType](#) type exposes the following members.

▲ Methods

	Name	Description
	Area	
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	isPointInside	
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Perimeter	



[ToString](#)

Returns a string that represents the current object.
(Inherited from [Object](#).)

[Top](#)

▲ See Also

Reference

[PolygonType Class](#)

[is2GraphObject Namespace](#)

PolygonTypeArea Method

[Missing <summary> documentation for "M:is2GraphObject.PolygonType.Area"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Area()
```

Return Value

Type: [Double](#)

[Missing <returns> documentation for "M:is2GraphObject.PolygonType.Area"]

▲ See Also

Reference

[PolygonType Class](#)

[is2GraphObject Namespace](#)

PolygonTypeisPointInside Method

[Missing <summary> documentation for
"M:is2GraphObject.PolygonType.isPointInside(is2GraphObject.PointType)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public bool isPointInside(  
    PointType P  
)
```

Parameters

P

Type: [is2GraphObjectPointType](#)

[Missing <param name="P"/> documentation for
"M:is2GraphObject.PolygonType.isPointInside(is2GraphObject.PointType)"]

Return Value

Type: [Boolean](#)

[Missing <returns> documentation for
"M:is2GraphObject.PolygonType.isPointInside(is2GraphObject.PointType)"]

▲ See Also

Reference

[PolygonType Class](#)

[is2GraphObject Namespace](#)

PolygonTypePerimeter Method

[Missing <summary> documentation for "M:is2GraphObject.PolygonType.Perimeter"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Perimeter()
```

Return Value

Type: [Double](#)

[Missing <returns> documentation for "M:is2GraphObject.PolygonType.Perimeter"]

▲ See Also

Reference


[PolygonType Class](#)

[is2GraphObject Namespace](#)

PolygonType Properties

The [PolygonType](#) type exposes the following members.

▲ Properties

Name	Description
 Radius	

[Top](#)

▲ See Also

Reference

[PolygonType Class](#)

[is2GraphObject Namespace](#)

PolygonTypeRadius Property

[Missing <summary> documentation for "P:is2GraphObject.PolygonType.Radius"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Radius { get; set; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[PolygonType Class](#)

[is2GraphObject Namespace](#)

PolylineElement Class

Clase fachada que representa el elemento de una polilínea.

▲ Inheritance Hierarchy [System Object](#) [is2GraphObject](#)
[PolylineElement](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax




C#

[Copy](#)

```
public class PolylineElement
```



The `PolylineElement` type exposes the following members.

▲ Constructors




	Name	Description
	PolylineElement(ArcType)	Constuctor que define el elemento como un Tipo Arco.
	PolylineElement(PointType)	Constuctor que define el elemento como un Tipo Punto.
	PolylineElement(SegmentType)	Constuctor que define el elemento como un Tipo Segmento.

[Top](#)

▲ Methods





	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)



	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

[Top](#)

▲ Properties

	Name	Description
	Arc	Propiedad de solo lectura. Devuelve un Tipo Arco.
	Point	Propiedad de solo lectura. Devuelve un Tipo Punto.
	Segment	Propiedad de solo lectura. Devuelve un Tipo Segmento.
	TypeElement	Devuelve un enum que indica el tipo de elemento que contiene, el cual puede ser: un punto, un segmento o un arco.

[Top](#)




▲ See Also

Reference

is2GraphObject Namespace

PolylineElement Constructor

▲ Overload List

	Name	Description
	PolylineElement(ArcType)	Constuctor que define el elemento como un Tipo Arco.
	PolylineElement(PointType)	Constuctor que define el elemento como un Tipo Punto.
	PolylineElement(SegmentType)	Constuctor que define el elemento como un Tipo Segmento.

[Top](#)

▲ See Also

Reference

[PolylineElement Class](#)

[is2GraphObject Namespace](#)

PolylineElement Constructor (ArcType)

Constructor que define el elemento como un Tipo Arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PolylineElement(  
    ArcType a  
)
```

Parameters

a

Type: [is2GraphObjectArcType](#)

[Missing <param name="a"/> documentation for
"M:is2GraphObject.PolylineElement.#ctor(is2GraphObject.ArcType)"]

▲ See Also

Reference

[PolylineElement Class](#)

[PolylineElement Overload](#)

[is2GraphObject Namespace](#)

PolylineElement Constructor (PointType)

Constructor que define el elemento como un Tipo Punto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PolylineElement(  
    PointType p  
)
```

Parameters

p

Type: [is2GraphObjectPointType](#)

Representa el pun

▲ See Also

Reference

[PolylineElement Class](#)

[PolylineElement Overload](#)

[is2GraphObject Namespace](#)

PolylineElement Constructor (SegmentType)

Constructor que define el elemento como un Tipo Segmento.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PolylineElement(  
    SegmentType s  
)
```

Parameters

s

Type: [is2GraphObjectSegmentType](#)

[Missing <param name="s"/> documentation for

"M:is2GraphObject.PolylineElement.#ctor(is2GraphObject.SegmentType)"]

▲ See Also

Reference

[PolylineElement Class](#)







[PolylineElement Overload](#)

[is2GraphObject Namespace](#)

PolylineElement Methods

The [PolylineElement](#) type exposes the following members.

▲ Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

[Top](#)

▲ See Also

Reference





[PolylineElement Class](#)

[is2GraphObject Namespace](#)

PolylineElement Properties

The [PolylineElement](#) type exposes the following members.

▲ Properties

	Name	Description
	Arc	Propiedad de solo lectura. Devuelve un Tipo Arco.
	Point	Propiedad de solo lectura. Devuelve un Tipo Punto.
	Segment	Propiedad de solo lectura. Devuelve un Tipo Segmento.
	TypeElement	Devuelve un enum que indica el tipo de elemento que contiene, el cual puede ser: un punto, un segmento o un arco.

[Top](#)

▲ See Also

Reference

[PolylineElement Class](#)

[is2GraphObject Namespace](#)

PolylineElementArc Property

Propiedad de solo lectura. Devuelve un Tipo Arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public ArcType Arc { get; }
```

Property Value

Type: [ArcType](#)

▲ See Also

Reference

[PolylineElement Class](#)

[is2GraphObject Namespace](#)

PolylineElementPoint Property

Propiedad de solo lectura. Devuelve un Tipo Punto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType Point { get; }
```

Property Value

Type: [PointType](#)

▲ See Also

Reference

[PolylineElement Class](#)

[is2GraphObject Namespace](#)

PolylineElementSegment Property

Propiedad de solo lectura. Devuelve un Tipo Segmento.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public SegmentType Segment { get; }
```

Property Value

Type: [SegmentType](#)

▲ See Also

Reference

[PolylineElement Class](#)

[is2GraphObject Namespace](#)

PolylineElementTypeElement Property

Devuelve un enum que indica el tipo de elemento que contiene, el cual puede ser: un punto, un segmento o un arco.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PolylineElementType TypeElement { get; }
```

Property Value

Type: [PolylineElementType](#)

▲ See Also

Reference

[PolylineElement Class](#)

[is2GraphObject Namespace](#)

PolylineElementType Enumeration

[Missing <summary> documentation for "T:is2GraphObject.PolylineElementType"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public enum PolylineElementType
```

▲ Members

Member name	Value	Description
Point	0	
Segment	1	
Arc	2	

▲ See Also

Reference

[is2GraphObject Namespace](#)

PolylineException Class

Representa una exception para el tipo Polyline Error.

▲ Inheritance Hierarchy [System Object System](#)

[Exception](#)

[is2GraphObject PolylineException](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public class PolylineException : Exception
```





The `PolylineException` type exposes the following members.

▲ Constructors





	Name	Description
	PolylineException	Constructor por defecto.
	PolylineException(String)	Constructor que toma como parámetro un tipo String .

[Top](#)

▲ Methods




	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash






function.
(Inherited from [Object](#).)

	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current instance. (Inherited from Exception .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Creates and returns a string representation of the current exception. (Inherited from Exception .)

[Top](#)


▲ Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)

	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the immediate frames on the call stack. (Inherited from Exception .)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception .)

[Top](#)

▲ Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

[Top](#)

▲ See Also

Reference

is2GraphObject Namespace

PolylineException Constructor

▲ Overload List

	Name	Description
	PolylineException	Constructor por defecto.
	PolylineException(String)	Constructor que toma como parámetro un tipo String.

[Top](#)

▲ See Also

Reference

[PolylineException Class](#)

[is2GraphObject Namespace](#)

PolylineException Constructor

Constructor por defecto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PolylineException()
```

▲ See Also

Reference

[PolylineException Class](#)

[PolylineException Overload](#)

[is2GraphObject Namespace](#)

PolylineException Constructor (String)

Constructor que toma como parámetro un tipo String.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PolylineException(  
    string msg  
)
```

Parameters

msg

Type: [SystemString](#)

Representa una cadena de caracteres que indica la naturaleza de la exception.

▲ See Also

Reference

[PolylineException Class](#)

[PolylineException Overload](#)







[is2GraphObject Namespace](#)

[SystemString](#)

PolylineException Methods

The [PolylineException](#) type exposes the following members.

Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception .)
	GetType	Gets the runtime type of the current

instance.
(Inherited from [Exception.](#))



[MemberwiseClone](#)

Creates a shallow copy of the current [Object](#).
(Inherited from [Object.](#))



[ToString](#)

Creates and returns a string representation of the current exception.
(Inherited from [Exception.](#))

[Top](#)

▲ See Also

Reference








[PolylineException Class](#)

[is2GraphObject Namespace](#)

PolylineException Properties

The [PolylineException](#) type exposes the following members.

▲ Properties

	Name	Description
	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception .)
	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception .)
	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception .)
	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception .)
	Message	Gets a message that describes the current exception. (Inherited from Exception .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception .)
	StackTrace	Gets a string representation of the

immediate frames on the call stack.
(Inherited from [Exception](#).)



[TargetSite](#)

Gets the method that throws the current exception.
(Inherited from [Exception](#).)

[Top](#)

▲ See Also

Reference


[PolylineException Class](#)

[is2GraphObject Namespace](#)

PolylineException Events

The [PolylineException](#) type exposes the following members.

▲ Events

	Name	Description
	SerializeObjectState	Occurs when an exception is serialized to create an exception state object that contains serialized data about the exception. (Inherited from Exception .)

[Top](#)

▲ See Also

Reference

[PolylineException Class](#)

[is2GraphObject Namespace](#)

PolylineType Class

Representa un tipo Polilínea.

▲ Inheritance Hierarchy [System Object](#) [is2GraphObject](#)
[PolylineType](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax


C#

[Copy](#)

```
public class PolylineType
```



The `PolylineType` type exposes the following members.

▲ Constructors

	Name	Description
	PolylineType	Constructor por defecto.

[Top](#)

▲ Methods

	Name	Description
	AddVertex	Agrega un vertice a la Polyline.
	Clear	Borra todos los vértices que contiene la polilinea.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Explode	Genera una explosión de los elementos que conforman la polilinea.
	Finalize	Destructor de la clase. (Overrides Object Finalize .)
	getByIndex	
		

	<code>getFirstElement</code>	Determina cuál es el primer elemento de la polilínea.
	<code>getFirstVertex</code>	Determina cuál es el primer punto de la polilínea.
	<code>GetHashCode</code>	Serves as the default hash function. (Inherited from Object .)
	<code>getLastElement</code>	Determina cuál es el último elemento de la polilínea.
	<code>getLastVertex</code>	Determina cuál es el último punto de la polilínea.
	<code>getNoItems</code>	Determina el número de items que contiene la polilínea.
	<code>getNoVertex</code>	Determina el número de vértices que contiene la polilínea.
	<code>GetType</code>	Gets the Type of the current instance. (Inherited from Object .)
	<code>JoinEntity(ArcType)</code>	Agrega un arco de circunferencia a la polilínea.
	<code>JoinEntity(PolylineType)</code>	Junta la polilínea "pl" dada a la polilínea actual.
	<code>JoinEntity(SegmentType)</code>	Agrega un segmento de recta a la polilínea.
	<code>MemberwiseClone</code>	Creates a shallow copy of the current Object . (Inherited from Object .)



[ToString](#)

Returns a string that represents the current object.
(Inherited from [Object](#).)

[Top](#)

▲ [See Also](#)

Reference

[is2GraphObject Namespace](#)

PolylineType Constructor

Constructor por defecto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PolylineType()
```

▲ See Also

Reference








[PolylineType Class](#)

[is2GraphObject Namespace](#)

PolylineType Methods

The [PolylineType](#) type exposes the following members.

▲ Methods

	Name	Description
	AddVertex	Agrega un vertice a la Polyline.
	Clear	Borra todos los vértices que contiene la polilinea.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Explode	Genera una explosión de los elementos que conforman la polilinea.
	Finalize	Destructor de la clase. (Overrides Object Finalize .)
	getByIndex	
	getFirstElement	Determina cuál es el primer elemento de la polilínea.
	getFirstVertex	Determina cuál es el primer punto de la polilínea.
	GetHashCode	Serves as the default hash function. (Inherited from Object .)

	getLastElement	Determina cuál es el último elemento de la polilínea.
	getLastVertex	Determina cuál es el último punto de la polilínea.
	getNoItems	Determina el número de items que contiene la polilínea.
	getNoVertex	Determina el número de vértices que contiene la polilínea.
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	JoinEntity(ArcType)	Agrega un arco de circunferencia a la polilínea.
	JoinEntity(PolylineType)	Junta la polilínea "pl" dada a la polilínea actual.
	JoinEntity(SegmentType)	Agrega un segmento de recta a la polilínea.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

[Top](#)

▲ See Also

Reference

[PolylineType Class](#)

[is2GraphObject Namespace](#)

PolylineTypeAddVertex Method

Agrega un vertice a la Polyline.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public void AddVertex(  
    PointType vertex  
)
```

Parameters

vertex

Type: [is2GraphObjectPointType](#)

Vértice que se agregara a la polilínea.

▲ See Also

Reference

[PolylineType Class](#)

[is2GraphObject Namespace](#)

PolylineTypeClear Method

Borra todos los vértices que contiene la polilínea.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public void Clear()
```

▲ See Also

Reference

[PolylineType Class](#)

[is2GraphObject Namespace](#)

PolylineTypeExplode Method

Genera una explosión de los elementos que conforman la polilínea.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PolylineElement[] Explode()
```

Return Value

Type: [PolylineElement](#)

Retorna un array de PolylineElement con los elementos de la polilínea. Si la polilínea esta vacia retorna null.

▲ See Also

Reference

[PolylineType Class](#)

[is2GraphObject Namespace](#)

PolylineTypeFinalize Method

Destructor de la clase.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
protected override void Finalize()
```

▲ See Also

Reference

[PolylineType Class](#)

[is2GraphObject Namespace](#)

PolylineType getByIndex Method

[Missing <summary> documentation for
"M:is2GraphObject.PolylineType.getByIndex(System.Int32)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public PolylineElement getByIndex(  
    int index  
)
```

Parameters

index

Type: [System Int32](#)

Indice del vertice a devolver.

Nota: El primer vértice de la polilínea tiene indice 0.

Return Value

Type: [PolylineElement](#)

Devuelve un punto de la polilínea segun su índice. Si la polilínea esta vacía devuelve null.

▲ See Also

Reference

[PolylineType Class](#)

[is2GraphObject Namespace](#)

PolylineType getFirstElement Method

Determina cuál es el primer elemento de la polilínea.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public PolylineElement getFirstElement()
```

Return Value

Type: [PolylineElement](#)

Devuelve el primer elemento de la polilínea. Si la polilínea esta vacía devuelve null.

▲ **Remarks** Un elemento de la polilínea puede ser: un punto, un segmento, un arco.

▲ **See Also**

Reference

[PolylineType Class](#)

[is2GraphObject Namespace](#)

PolylineType.getFirstVertex Method

Determina cuál es el primer punto de la polilínea.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType getFirstVertex()
```

Return Value

Type: [PointType](#)

Devuelve el primer punto de la polilínea. Si la polilínea esta vacía devuelve null.

▲ See Also

Reference

[PolylineType Class](#)

[is2GraphObject Namespace](#)

PolylineType getLastElement Method

Determina cuál es el último elemento de la polilínea.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public PolylineElement getLastElement()
```

Return Value

Type: [PolylineElement](#)

Devuelve el último elemento de la polilínea. Si la polilínea esta vacía devuelve null.

▲ **Remarks** Un elemento de la polilínea puede ser: un punto, un segmento, un arco.

▲ **See Also**

Reference

[PolylineType Class](#)

[is2GraphObject Namespace](#)

PolylineTypegetLastVertex Method

Determina cuál es el último punto de la polilínea.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType getLastVertex()
```

Return Value

Type: [PointType](#)

Devuelve el último punto de la polilínea. Si la polilínea esta vacía devuelve null.

▲ See Also

Reference

[PolylineType Class](#)

[is2GraphObject Namespace](#)

PolylineType.getItems Method

Determina el número de items que contiene la polilínea.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public int getNoItems()
```

Return Value

Type: [Int32](#)

Devuelve el número de items de la polilínea.

▲ See Also

Reference

[PolylineType Class](#)

[is2GraphObject Namespace](#)

PolylineTypegetNoVertex Method

Determina el número de vértices que contiene la polilínea.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public int getNoVertex()
```

Return Value

Type: [Int32](#)

Devuelve el número de vértices de la polilínea.

▲ See Also




Reference

[PolylineType Class](#)

[is2GraphObject Namespace](#)

PolylineTypeJoinEntity Method

▲ Overload List

	Name	Description
	JoinEntity(ArcType)	Agrega un arco de circunferencia a la polilínea.
	JoinEntity(PolylineType)	Junta la polilínea "pl" dada a la polilínea actual.
	JoinEntity(SegmentType)	Agrega un segmento de recta a la polilínea.

[Top](#)

▲ See Also

Reference

[PolylineType Class](#)

[is2GraphObject Namespace](#)

PolylineTypeJoinEntity Method (ArcType)

Agrega un arco de circunferencia a la polilínea.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public void JoinEntity(  
    ArcType A  
)
```

Parameters

A

Type: [is2GraphObjectArcType](#)
Arco de circunferencia a juntar.

Return Value

Type:

▲ See Also

Reference

[PolylineType Class](#)

[JoinEntity Overload](#)

[is2GraphObject Namespace](#)

PolylineTypeJoinEntity Method (PolylineType)

Junta la polilínea "pl" dada a la polilínea actual.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public void JoinEntity(  
    PolylineType pl  
)
```

Parameters

pl

Type: [is2GraphObjectPolylineType](#)

Polilínea a juntar.

▲ See Also

Reference

[PolylineType Class](#)

[JoinEntity Overload](#)

[is2GraphObject Namespace](#)

PolylineTypeJoinEntity Method (SegmentType)

Agrega un segmento de recta a la polilínea.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public void JoinEntity(  
    SegmentType S  
)
```

Parameters

S

Type: [is2GraphObjectSegmentType](#)

[Missing <param name="S"/> documentation for

"M:is2GraphObject.PolylineType.JoinEntity(is2GraphObject.SegmentType)"]

Return Value

Type:

▲ See Also

Reference

[PolylineType Class](#)

[JoinEntity Overload](#)

[is2GraphObject Namespace](#)

RectangleType Class

Representa un tipo Rectángulo.

▲ Inheritance Hierarchy [System Object](#) [is2GraphObject](#)
[RectangleType](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax



C#

[Copy](#)

```
public class RectangleType
```






The **RectangleType** type exposes the following members.







▲ Constructors

	Name	Description
	RectangleType(PointType, PointType)	
	RectangleType(PointType, Double, Double)	

[Top](#)


▲ Methods

	Name	Description
	Area	
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetSides	

	GetType	Gets the Type of the current instance. (Inherited from Object .)
	GetVertex	
	isPointInside	
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Perimeter	
	ToString	Returns a string that represents the current object. (Inherited from Object .)

[Top](#)

▲ Fields

	Name	Description
	Lcorner	
	Rcorner	

[Top](#)



▲ See Also

Reference

[is2GraphObject Namespace](#)

RectangleType Constructor

▲ Overload List

	Name	Description
	RectangleType(PointType, PointType)	
	RectangleType(PointType, Double, Double)	

[Top](#)

▲ See Also

Reference

[RectangleType Class](#)

[is2GraphObject Namespace](#)

RectangleType Constructor (PointType, PointType)

[Missing <summary> documentation for
"M:is2GraphObject.RectangleType.#ctor(is2GraphObject.PointType,is2GraphObject.Point"

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public RectangleType(  
    PointType pL,  
    PointType pR  
)
```

Parameters

pL

Type: [is2GraphObjectPointType](#)

[Missing <param name="pL"/> documentation for
"M:is2GraphObject.RectangleType.#ctor(is2GraphObject.PointType,is2GraphObject.Poi

pR

Type: [is2GraphObjectPointType](#)

[Missing <param name="pR"/> documentation for
"M:is2GraphObject.RectangleType.#ctor(is2GraphObject.PointType,is2GraphObject.Poi

▲ See Also

Reference

[RectangleType Class](#)

RectangleType Overload
is2GraphObject Namespace

RectangleType Constructor (PointType, Double, Double)

[Missing <summary> documentation for
"M:is2GraphObject.RectangleType.#ctor(is2GraphObject.PointType,System.Double,Syst

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public RectangleType(  
    PointType pL,  
    double Xoffset,  
    double Yoffset  
)
```

Parameters

pL

Type: [is2GraphObject PointType](#)

[Missing <param name="pL"/> documentation for

"M:is2GraphObject.RectangleType.#ctor(is2GraphObject.PointType,System.Double,Sy

Xoffset

Type: [System Double](#)

[Missing <param name="Xoffset"/> documentation for

"M:is2GraphObject.RectangleType.#ctor(is2GraphObject.PointType,System.Double,Sy

Yoffset

Type: [System Double](#)

[Missing <param name="Yoffset"/> documentation for

"M:is2GraphObject.RectangleType.#ctor(is2GraphObject.PointType,System.Double,Sy

▲ See Also

Reference

[RectangleType Class](#)


[RectangleType Overload](#)

[is2GraphObject Namespace](#)

RectangleType Fields

The [RectangleType](#) type exposes the following members.

▲ Fields

	Name	Description
	Lcorner	
	Rcorner	

[Top](#)

▲ See Also

Reference

[RectangleType Class](#)

[is2GraphObject Namespace](#)

RectangleTypeLcorner Field

[Missing <summary> documentation for "F:is2GraphObject.RectangleType.Lcorner"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType Lcorner
```

Field Value

Type: [PointType](#)

▲ See Also

Reference

[RectangleType Class](#)

[is2GraphObject Namespace](#)

RectangleTypeRcorner Field

[Missing <summary> documentation for "F:is2GraphObject.RectangleType.Rcorner"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType Rcorner
```

Field Value

Type: [PointType](#)

▲ See Also

Reference










[RectangleType Class](#)

[is2GraphObject Namespace](#)

RectangleType Methods

The [RectangleType](#) type exposes the following members.

Methods

	Name	Description
	Area	
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetSides	
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	GetVertex	
	isPointInside	
	MemberwiseClone	Creates a shallow copy of the current Object .

(Inherited from [Object](#).)



[Perimeter](#)



[ToString](#)

Returns a string that represents the current object.
(Inherited from [Object](#).)

[Top](#)

▲ See Also

Reference

[RectangleType Class](#)

[is2GraphObject Namespace](#)

RectangleTypeArea Method

[Missing <summary> documentation for "M:is2GraphObject.RectangleType.Area"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Area()
```

Return Value

Type: [Double](#)

[Missing <returns> documentation for "M:is2GraphObject.RectangleType.Area"]

▲ See Also

Reference

[RectangleType Class](#)

[is2GraphObject Namespace](#)

RectangleType GetSides Method

[Missing <summary> documentation for
"M:is2GraphObject.RectangleType.GetSides(is2GraphObject.SegmentType@,is2GraphO

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public void GetSides(  
    out SegmentType s1,  
    out SegmentType s2,  
    out SegmentType s3,  
    out SegmentType s4  
)
```

Parameters

s1

Type: [is2GraphObject SegmentType](#)

[Missing <param name="s1"/> documentation for

"M:is2GraphObject.RectangleType.GetSides(is2GraphObject.SegmentType@,is2Grapl

s2

Type: [is2GraphObject SegmentType](#)

[Missing <param name="s2"/> documentation for

"M:is2GraphObject.RectangleType.GetSides(is2GraphObject.SegmentType@,is2Grapl

s3

Type: [is2GraphObject SegmentType](#)

[Missing <param name="s3"/> documentation for

"M:is2GraphObject.RectangleType.GetSides(is2GraphObject.SegmentType@,is2Grapl

s4

Type: [is2GraphObject SegmentType](#)

[Missing <param name="s4"/> documentation for

"M:is2GraphObject.RectangleType.GetSides(is2GraphObject.SegmentType@,is2Grapl

See Also

Reference

[RectangleType Class](#)

[is2GraphObject Namespace](#)

RectangleType GetVertex Method

[Missing <summary> documentation for
"M:is2GraphObject.RectangleType.GetVertex(is2GraphObject.PointType@,is2GraphObje

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public void GetVertex(  
    out PointType v1,  
    out PointType v2,  
    out PointType v3,  
    out PointType v4  
)
```

Parameters

v1

Type: [is2GraphObject PointType](#)

[Missing <param name="v1"/> documentation for

"M:is2GraphObject.RectangleType.GetVertex(is2GraphObject.PointType@,is2GraphO

v2

Type: [is2GraphObject PointType](#)

[Missing <param name="v2"/> documentation for

"M:is2GraphObject.RectangleType.GetVertex(is2GraphObject.PointType@,is2GraphO

v3

Type: [is2GraphObject PointType](#)

[Missing <param name="v3"/> documentation for

"M:is2GraphObject.RectangleType.GetVertex(is2GraphObject.PointType@,is2GraphO

v4

Type: [is2GraphObject PointType](#)

[Missing <param name="v4"/> documentation for

"M:is2GraphObject.RectangleType.GetVertex(is2GraphObject.PointType@,is2GraphO

See Also

Reference

[RectangleType Class](#)

[is2GraphObject Namespace](#)

RectangleTypeIsPointInside Method

[Missing <summary> documentation for
"M:is2GraphObject.RectangleType.isPointInside(is2GraphObject.PointType)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public bool isPointInside(  
    PointType P  
)
```

Parameters

P

Type: [is2GraphObjectPointType](#)

[Missing <param name="P"/> documentation for
"M:is2GraphObject.RectangleType.isPointInside(is2GraphObject.PointType)"]

Return Value

Type: [Boolean](#)

[Missing <returns> documentation for
"M:is2GraphObject.RectangleType.isPointInside(is2GraphObject.PointType)"]

▲ See Also

Reference

[RectangleType Class](#)

[is2GraphObject Namespace](#)

RectangleTypePerimeter Method

[Missing <summary> documentation for "M:is2GraphObject.RectangleType.Perimeter"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Perimeter()
```

Return Value

Type: [Double](#)

[Missing <returns> documentation for "M:is2GraphObject.RectangleType.Perimeter"]

▲ See Also

Reference

[RectangleType Class](#)

[is2GraphObject Namespace](#)

SegmentType Class

Representa un tipo Segmento. Un segmento se define por dos puntos.

▲ Inheritance Hierarchy [System Object](#) [is2GraphObject](#)
[SegmentType](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax



C#

[Copy](#)

```
public class SegmentType
```




The `SegmentType` type exposes the following members.

▲ Constructors

	Name	Description
	SegmentType	Constructor por defecto.
	SegmentType(PointType, PointType)	Constructor que toma 2 parámetros: dos puntos. Representan los puntos inicio y final del segmento respectivamente.

[Top](#)




▲ Methods

	Name	Description
	CheckRelativePosition	Comprueba si un punto "P" dado, cumple con la condición de posición relativa al segmento establecida por "condition".
	ConvertToLine	Convierte el segmento en un LineType.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)





	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
💡	GetHashCode	Serves as the default hash function. (Inherited from Object .)
💡	GetType	Gets the Type of the current instance. (Inherited from Object .)
💡	isParallelTo	Determina si el segmento es paralelo a otro segmento 'S' dado.
💡	isPerpendicularTo	Determina si el segmento es perpendicular a otro segmento 'S' dado.
💡	isSecanteTo	Determina si el segmento es secante a otro segmento 'S' dado.
💡💡	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
💡	PointInSegment	Determina si el punto 'P' pasado por parámetro pertenece a Segmento.
💡	PointMayorX	Determina cuál es el punto del segmento que tiene mayor X.
💡	PointMayorY	Determina cuál es el punto del

segmento que tiene mayor Y.

	PointMenorX	Determina cuál es el punto del segmento que tiene menor X.
	PointMenorY	Determina cuál es el punto del segmento que tiene menor Y.
	ToString	Returns a string that represents the current object. (Inherited from Object .)




[Top](#)

▲ Fields





	Name	Description
	EndPoint	Representa el punto de final del segmento.
	StartPoint	Representa el punto de inicio del segmento.

[Top](#)

▲ Properties

	Name	Description
	Angle	Propiedad de solo lectura. Devuelve el ángulo que forma el segmento con respecto al eje de las Abcisas (Eje X).
	isHorizontal	Propiedad de solo lectura. Devuelve true si el segmento es Horizontal. La característica verticalidad se da si el segmento es paralelo al eje de las Abcisas (Eje X).
	isObliquo	Propiedad de solo lectura. Devuelve true si el segmento es Obliquo a los ejes de

coordenadas. Se define el segmento como obliquo si no es paralelo a ninguno de los ejes de coordenadas.

	isVertical	Propiedad de solo lectura. Devuelve true si el segmento es Vertical. La característica verticalidad se da si el segmento es paralelo al eje de las Ordenadas (Eje Y).
	Longitude	Propiedad de solo lectura. Devuelve la longitud del segmento.
	MidPoint	Propiedad de solo lectura. Devuelve el punto medio del segmento.
	Slope	Propiedad de solo lectura. Devuelve el valor de la pendiente del segmento. Nota: La propiedad devuelve NaN si el segmento es vertical, o sea, si es paralelo al eje de laSs Ordenadas (Eje Y).

[Top](#)

▲ [See Also](#)

Reference

[is2GraphObject Namespace](#)

SegmentType Constructor

▲ Overload List

	Name	Description
	SegmentType	Constructor por defecto.
	SegmentType(PointType, PointType)	Constructor que toma 2 parámetros: dos puntos. Representan los puntos inicio y final del segmento respectivamente.

[Top](#)

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentType Constructor

Constructor por defecto.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public SegmentType()
```

▲ See Also

Reference

[SegmentType Class](#)

[SegmentType Overload](#)

[is2GraphObject Namespace](#)

SegmentType Constructor (PointType, PointType)

Constructor que toma 2 parámetros: dos puntos. Representan los puntos inicio y final del segmento respectivamente.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public SegmentType(  
    PointType startPoint,  
    PointType endPoint  
)
```

Parameters

startPoint

Type: [is2GraphObject PointType](#)

Punto de inicio del segmento.

endPoint

Type: [is2GraphObject PointType](#)

Punto final del segmento.

▲ See Also

Reference

[SegmentType Class](#)



[SegmentType Overload](#)

[is2GraphObject Namespace](#)

SegmentType Fields

The [SegmentType](#) type exposes the following members.

▲ Fields

	Name	Description
	EndPoint	Representa el punto de final del segmento.
	StartPoint	Representa el punto de inicio del segmento.

[Top](#)

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentTypeEndPoint Field

Representa el punto de final del segmento.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType EndPoint
```

Field Value

Type: [PointType](#)

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentTypeStartPoint Field

Representa el punto de inicio del segmento.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType StartPoint
```

Field Value

Type: [PointType](#)

▲ See Also

Reference







[SegmentType Class](#)











[is2GraphObject Namespace](#)

SegmentType Methods

The [SegmentType](#) type exposes the following members.

Methods

	Name	Description
	CheckRelativePosition	Comprueba si un punto "P" dado, cumple con la condición de posición relativa al segmento establecida por "condition".
	ConvertToLine	Convierte el segmento en un LineType.
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)

	isParallelTo	Determina si el segmento es paralelo a otro segmento 'S' dado.
	isPerpendicularTo	Determina si el segmento es perpendicular a otro segmento 'S' dado.
	isSecanteTo	Determina si el segmento es secante a otro segmento 'S' dado.
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	PointInSegment	Determina si el punto 'P' pasado por parámetro pertenece a Segmento.
	PointMayorX	Determina cuál es el punto del segmento que tiene mayor X.
	PointMayorY	Determina cuál es el punto del segmento que tiene mayor Y.
	PointMenorX	Determina cuál es el punto del segmento que tiene menor X.
	PointMenorY	Determina cuál es el punto del segmento que tiene menor Y.
	ToString	Returns a string that represents the current object. (Inherited from Object .)

[Top](#)

▲ [See Also](#)

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentType

CheckRelativePosition Method

Comprueba si un punto "P" dado, cumple con la condición de posición relativa al segmento establecida por "condition".

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public bool CheckRelativePosition(  
    PointType P,  
    PointLinePosition condition  
)
```

Parameters

P

Type: [is2GraphObject PointType](#)

[Missing <param name="P"/> documentation for
"M:is2GraphObject.SegmentType.CheckRelativePosition(is2GraphObject.PointType,is

condition

Type: [is2GraphObject PointLinePosition](#)

[Missing <param name="condition"/> documentation for
"M:is2GraphObject.SegmentType.CheckRelativePosition(is2GraphObject.PointType,is

Return Value

Type: [Boolean](#)

Devuelve **true** si se cumple ▲ [See Also](#)

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentTypeConvertToLine Method

Convierte el segmento en un LineType.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public LineType ConvertToLine()
```

Return Value

Type: [LineType](#)

Devuelve el tipo linea que pasa por los dos puntos del segmento.

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentType isParallelTo Method

Determina si el segmento es paralelo a otro segmento 'S' dado.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public bool isParallelTo(  
    SegmentType S  
)
```

Parameters

S

Type: [is2GraphObject SegmentType](#)

Segmento contra el que se comprueba la propiedad de paralelismo.

Return Value

Type: [Boolean](#)

Devuelve **true** si el segmento que invoca al método es paralelo al segmento "S" dado, en caso contrario devuelve **false**.

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentType isPerpendicularTo Method

Determina si el segmento es perpendicular a otro segmento 'S' dado.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public bool isPerpendicularTo(  
    SegmentType S  
)
```

Parameters

S

Type: [is2GraphObject SegmentType](#)

Segmento contra el que se comprueba la propiedad de perpendicularidad.

Return Value

Type: [Boolean](#)

Devuelve **true** si el segmento que invoca al método es perpendicular al segmento "S" dado, en caso contrario devuelve **false**.

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentType isSecanteTo Method

Determina si el segmento es secante a otro segmento 'S' dado.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public bool isSecanteTo(  
    SegmentType S  
)
```

Parameters

S

Type: [is2GraphObject SegmentType](#)

Segmento contra el que se comprueba la propiedad de intersección.

Return Value

Type: [Boolean](#)

Devuelve **true** si el segmento que invoca al método es secante al segmento "S" dado, en caso contrario devuelve **false**.

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentType PointInSegment Method

Determina si el punto 'P' pasado por parámetro pertenece a Segmento.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public bool PointInSegment(  
    PointType P  
)
```

Parameters

P

Type: [is2GraphObject PointType](#)

Punto para el que se quiere comprobar la pertenencia al segmento.

Return Value

Type: [Boolean](#)

Devuelve **true** si el punto pertenece al segmento en caso contrario devuelve **false**.

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

[is2GraphObject PointType](#)

SegmentTypePointMayorX Method

Determina cuál es el punto del segmento que tiene mayor X.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType PointMayorX()
```

Return Value

Type: [PointType](#)

Devuelve el punto de mayor X en el segmento.

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentTypePointMayorY Method

Determina cuál es el punto del segmento que tiene mayor Y.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType PointMayorY()
```

Return Value

Type: [PointType](#)

Devuelve el punto de mayor Y en el segmento.

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentTypePointMenorX

Method

Determina cuál es el punto del segmento que tiene menor X.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType PointMenorX()
```

Return Value

Type: [PointType](#)

Devuelve el punto de menor X en el segmento.

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentTypePointMenorY Method

Determina cuál es el punto del segmento que tiene menor Y.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType PointMenorY()
```

Return Value

Type: [PointType](#)

Devuelve el punto de menor Y en el segmento.

▲ See Also

Reference








[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentType Properties

The [SegmentType](#) type exposes the following members.

▲ Properties

	Name	Description
	Angle	Propiedad de solo lectura. Devuelve el ángulo que forma el segmento con respecto al eje de las Abcisas (Eje X).
	isHorizontal	Propiedad de solo lectura. Devuelve true si el segmento es Horizontal. La característica verticalidad se da si el segmento es paralelo al eje de las Abcisas (Eje X).
	isObliquo	Propiedad de solo lectura. Devuelve true si el segmento es Obliquo a los ejes de coodenadas. Se define el segmento como obliquo si no es paralelo a ninguno de los ejes de coordenadas.
	isVertical	Propiedad de solo lectura. Devuelve true si el segmento es Vertical. La característica verticalidad se da si el segmento es paralelo al eje de las Ordenadas (Eje Y).
	Longitude	Propiedad de solo lectura. Devuelve la longitud del segmento.
	MidPoint	Propiedad de solo lectura. Devuelve el punto medio del segmento.
		

Slope

Propiedad de solo lectura. Devuelve el valor de la pendiente del segmento.

Nota: La propiedad devuelve NaN si el segmento es vertical, o sea, si es paralelo al eje de las Ordenadas (Eje Y).

[Top](#)

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentTypeAngle Property

Propiedad de solo lectura. Devuelve el ángulo que forma el segmento con respecto al eje de las Abcisas (Eje X).

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Angle { get; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentType isHorizontal Property

Propiedad de solo lectura. Devuelve **true** si el segmento es Horizontal. La característica verticalidad se da si el segmento es paralelo al eje de las Abcisas (Eje X).

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public bool isHorizontal { get; }
```

Property Value

Type: [Boolean](#)

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentType isObliquo Property

Propiedad de solo lectura. Devuelve **true** si el segmento es Obliquo a los ejes de coodenadas. Se define el segmento como obliquo si no es paralelo a ninguno de los ejes de coordenadas.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public bool isObliquo { get; }
```

Property Value

Type: [Boolean](#)

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentType isVertical Property

Propiedad de solo lectura. Devuelve **true** si el segmento es Vertical. La característica verticalidad se da si el segmento es paralelo al eje de las Ordenadas (Eje Y).

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public bool isVertical { get; }
```

Property Value

Type: [Boolean](#)

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentTypeLongitude Property

Propiedad de solo lectura. Devuelve la longitud del segmento.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Longitude { get; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentTypeMidPoint Property

Propiedad de solo lectura. Devuelve el punto medio del segmento.

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public PointType MidPoint { get; }
```

Property Value

Type: [PointType](#)

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SegmentType Slope Property

Propiedad de solo lectura. Devuelve el valor de la pendiente del segmento.

Nota: La propiedad devuelve NaN si el segmento es vertical, o sea, si es paralelo al eje de laSs Ordenadas (Eje Y).

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ **Syntax**

C#

[Copy](#)

```
public double Slope { get; }
```

Property Value

Type: [Double](#)

▲ See Also

Reference

[SegmentType Class](#)

[is2GraphObject Namespace](#)

SphericalPointType Class

[Missing <summary> documentation for "T:is2GraphObject.SphericalPointType"]

▲ Inheritance Hierarchy [System Object](#) [is2GraphObject](#)
[SphericalPointType](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax


C#

[Copy](#)

```
public class SphericalPointType
```






The `SphericalPointType` type exposes the following members.

Constructors

Name	Description
 SphericalPointType	

[Top](#)

Methods

Name	Description
 Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
 Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
 GetHashCode	Serves as the default hash function. (Inherited from Object .)
 GetType	Gets the Type of the current instance. (Inherited from Object .)
 MemberwiseClone	Creates a shallow copy of the current Object .

(Inherited from [Object](#).)



[ToString](#)

Returns a string that represents the current object.
(Inherited from [Object](#).)

[Top](#)

▲ [See Also](#)

Reference

[is2GraphObject Namespace](#)

SphericalPointType Constructor

[Missing <summary> documentation for "M:is2GraphObject.SphericalPointType.#ctor"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public SphericalPointType()
```

▲ See Also

Reference







[SphericalPointType Class](#)

[is2GraphObject Namespace](#)

SphericalPointType Methods

The [SphericalPointType](#) type exposes the following members.

▲ Methods

	Name	Description
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	ToString	Returns a string that represents the current object. (Inherited from Object .)

[Top](#)

▲ See Also

Reference

[SphericalPointType Class](#)

[is2GraphObject Namespace](#)

TriangleType Class

Representa un tipo Triángulo.

▲ Inheritance Hierarchy [System Object](#) [is2GraphObject](#)
[TriangleType](#)

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax


C#

Copy

```
public class TriangleType
```








The `TriangleType` type exposes the following members.

Constructors












Name	Description
 TriangleType	

[Top](#)

Methods

Name	Description
 Area	
 CircumscriptCicle	
 Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
 Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
 GetAlturas	
 GetBisectrices	
 GetHashCode	Serves as the default hash function.

(Inherited from [Object](#).)

	GetMediatrices	
	GetSides	
	GetType	Gets the Type of the current instance. (Inherited from Object .)
	GetVertex	
	InscriptCicle	
	isPointInside	
	MemberwiseClone	Creates a shallow copy of the current Object . (Inherited from Object .)
	Perimeter	
	ToString	Returns a string that represents the current object. (Inherited from Object .)
	TypeByAngle	
	TypeBySide	

[Top](#)

▲ [See Also](#)

Reference

[is2GraphObject Namespace](#)

TriangleType Constructor

[Missing <summary> documentation for
"M:is2GraphObject.TriangleType.#ctor(is2GraphObject.PointType,is2GraphObject.PointTy

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public TriangleType(  
    PointType v1,  
    PointType v2,  
    PointType v3  
)
```

Parameters

v1

Type: [is2GraphObjectPointType](#)

[Missing <param name="v1"/> documentation for
"M:is2GraphObject.TriangleType.#ctor(is2GraphObject.PointType,is2GraphObject.Point

v2

Type: [is2GraphObjectPointType](#)

[Missing <param name="v2"/> documentation for
"M:is2GraphObject.TriangleType.#ctor(is2GraphObject.PointType,is2GraphObject.Point

v3

Type: [is2GraphObjectPointType](#)

[Missing <param name="v3"/> documentation for
"M:is2GraphObject.TriangleType.#ctor(is2GraphObject.PointType,is2GraphObject.Point

▲ See Also

Reference











[TriangleType Class](#)

[is2GraphObject Namespace](#)

TriangleType Methods

The [TriangleType](#) type exposes the following members.

Methods

	Name	Description
	Area	
	CircumscriptCicle	
	Equals	Determines whether the specified object is equal to the current object. (Inherited from Object .)
	Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. (Inherited from Object .)
	GetAlturas	
	GetBisectrices	
	GetHashCode	Serves as the default hash function. (Inherited from Object .)
	GetMediatrixes	
	GetSides	
	GetType	Gets the Type of the current instance.

(Inherited from [Object](#).)



[GetVertex](#)



[InscriptCicle](#)



[isPointInside](#)



[MemberwiseClone](#)

Creates a shallow copy of the current [Object](#).
(Inherited from [Object](#).)



[Perimeter](#)



[ToString](#)

Returns a string that represents the current object.
(Inherited from [Object](#).)



[TypeByAngle](#)



[TypeBySide](#)

[Top](#)

▲ See Also

Reference

[TriangleType Class](#)

[is2GraphObject Namespace](#)

TriangleTypeArea Method

[Missing <summary> documentation for "M:is2GraphObject.TriangleType.Area"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Area()
```

Return Value

Type: [Double](#)

[Missing <returns> documentation for "M:is2GraphObject.TriangleType.Area"]

▲ See Also

Reference

[TriangleType Class](#)

[is2GraphObject Namespace](#)

TriangleTypeCircunscriptCicle Method

[Missing <summary> documentation for
"M:is2GraphObject.TriangleType.CircunscriptCicle"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public CircleType CircunscriptCicle()
```

Return Value

Type: [CircleType](#)

[Missing <returns> documentation for
"M:is2GraphObject.TriangleType.CircunscriptCicle"]

▲ See Also

Reference

[TriangleType Class](#)

[is2GraphObject Namespace](#)

TriangleType GetAlturas Method

[Missing <summary> documentation for

"M:is2GraphObject.TriangleType.GetAlturas(is2GraphObject.SegmentType@,is2GraphOI

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public void GetAlturas(  
    out SegmentType s1,  
    out SegmentType s2,  
    out SegmentType s3  
)
```

Parameters

s1

Type: [is2GraphObject SegmentType](#)

[Missing <param name="s1"/> documentation for

"M:is2GraphObject.TriangleType.GetAlturas(is2GraphObject.SegmentType@,is2GraphObject.SegmentType@,is2GraphObject.SegmentType@)"]

s2

Type: [is2GraphObject SegmentType](#)

[Missing <param name="s2"/> documentation for

"M:is2GraphObject.TriangleType.GetAlturas(is2GraphObject.SegmentType@,is2GraphObject.SegmentType@,is2GraphObject.SegmentType@)"]

s3

Type: [is2GraphObject SegmentType](#)

[Missing <param name="s3"/> documentation for

"M:is2GraphObject.TriangleType.GetAlturas(is2GraphObject.SegmentType@,is2GraphObject.SegmentType@,is2GraphObject.SegmentType@)"]

See Also

Reference

[TriangleType Class](#)

[is2GraphObject Namespace](#)

TriangleType GetBisectrices Method

[Missing <summary> documentation for

"M:is2GraphObject.TriangleType.GetBisectrices(is2GraphObject.SegmentType@,is2Grap

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public void GetBisectrices(  
    out SegmentType s1,  
    out SegmentType s2,  
    out SegmentType s3  
)
```

Parameters

s1

Type: [is2GraphObject](#) [SegmentType](#)

[Missing <param name="s1"/> documentation for

"M:is2GraphObject.TriangleType.GetBisectrices(is2GraphObject.SegmentType@,is2G

s2

Type: [is2GraphObject](#) [SegmentType](#)

[Missing <param name="s2"/> documentation for

"M:is2GraphObject.TriangleType.GetBisectrices(is2GraphObject.SegmentType@,is2G

s3

Type: [is2GraphObject](#) [SegmentType](#)

[Missing <param name="s3"/> documentation for

"M:is2GraphObject.TriangleType.GetBisectrices(is2GraphObject.SegmentType@,is2G

▲ See Also

Reference

[TriangleType Class](#)

[is2GraphObject Namespace](#)

TriangleType GetMediatrices Method

[Missing <summary> documentation for

"M:is2GraphObject.TriangleType.GetMediatrices(is2GraphObject.SegmentType@,is2Gra

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public void GetMediatrices(  
    out SegmentType s1,  
    out SegmentType s2,  
    out SegmentType s3  
)
```

Parameters

s1

Type: [is2GraphObject](#) [SegmentType](#)

[Missing <param name="s1"/> documentation for

"M:is2GraphObject.TriangleType.GetMediatrices(is2GraphObject.SegmentType@,is2G

s2

Type: [is2GraphObject](#) [SegmentType](#)

[Missing <param name="s2"/> documentation for

"M:is2GraphObject.TriangleType.GetMediatrices(is2GraphObject.SegmentType@,is2G

s3

Type: [is2GraphObject](#) [SegmentType](#)

[Missing <param name="s3"/> documentation for

"M:is2GraphObject.TriangleType.GetMediatrices(is2GraphObject.SegmentType@,is2G

▲ See Also

Reference

[TriangleType Class](#)

[is2GraphObject Namespace](#)

TriangleType GetSides Method

[Missing <summary> documentation for

"M:is2GraphObject.TriangleType.GetSides(is2GraphObject.SegmentType@,is2GraphObj

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public void GetSides(  
    out SegmentType s1,  
    out SegmentType s2,  
    out SegmentType s3  
)
```

Parameters

s1

Type: [is2GraphObject SegmentType](#)

[Missing <param name="s1"/> documentation for

"M:is2GraphObject.TriangleType.GetSides(is2GraphObject.SegmentType@,is2GraphC

s2

Type: [is2GraphObject SegmentType](#)

[Missing <param name="s2"/> documentation for

"M:is2GraphObject.TriangleType.GetSides(is2GraphObject.SegmentType@,is2GraphC

s3

Type: [is2GraphObject SegmentType](#)

[Missing <param name="s3"/> documentation for

"M:is2GraphObject.TriangleType.GetSides(is2GraphObject.SegmentType@,is2GraphC

See Also

Reference

[TriangleType Class](#)

[is2GraphObject Namespace](#)

TriangleType GetVertex Method

[Missing <summary> documentation for

"M:is2GraphObject.TriangleType.GetVertex(is2GraphObject.PointType@,is2GraphObject

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:

1.0.0.0 (1.0.0.0) ▲ Syntax

C#

[Copy](#)

```
public void GetVertex(  
    out PointType v1,  
    out PointType v2,  
    out PointType v3  
)
```

Parameters

v1

Type: [is2GraphObject PointType](#)

[Missing <param name="v1"/> documentation for

"M:is2GraphObject.TriangleType.GetVertex(is2GraphObject.PointType@,is2GraphObj

v2

Type: [is2GraphObject PointType](#)

[Missing <param name="v2"/> documentation for

"M:is2GraphObject.TriangleType.GetVertex(is2GraphObject.PointType@,is2GraphObj

v3

Type: [is2GraphObject PointType](#)

[Missing <param name="v3"/> documentation for

"M:is2GraphObject.TriangleType.GetVertex(is2GraphObject.PointType@,is2GraphObj

See Also

Reference

[TriangleType Class](#)

[is2GraphObject Namespace](#)

TriangleTypeInscriptCicle Method

[Missing <summary> documentation for "M:is2GraphObject.TriangleType.InscriptCicle"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public CircleType InscriptCicle()
```

Return Value

Type: [CircleType](#)

[Missing <returns> documentation for "M:is2GraphObject.TriangleType.InscriptCicle"]

▲ See Also

Reference

[TriangleType Class](#)

[is2GraphObject Namespace](#)

TriangleTypeisPointInside Method

[Missing <summary> documentation for
"M:is2GraphObject.TriangleType.isPointInside(is2GraphObject.PointType)"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public bool isPointInside(  
    PointType P  
)
```

Parameters

P

Type: [is2GraphObjectPointType](#)

[Missing <param name="P"/> documentation for
"M:is2GraphObject.TriangleType.isPointInside(is2GraphObject.PointType)"]

Return Value

Type: [Boolean](#)

[Missing <returns> documentation for
"M:is2GraphObject.TriangleType.isPointInside(is2GraphObject.PointType)"]

▲ See Also

Reference

[TriangleType Class](#)

[is2GraphObject Namespace](#)

TriangleTypePerimeter Method

[Missing <summary> documentation for "M:is2GraphObject.TriangleType.Perimeter"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public double Perimeter()
```

Return Value

Type: [Double](#)

[Missing <returns> documentation for "M:is2GraphObject.TriangleType.Perimeter"]

▲ See Also

Reference

[TriangleType Class](#)

[is2GraphObject Namespace](#)

TriangleTypeTypeByAngle Method

[Missing <summary> documentation for "M:is2GraphObject.TriangleType.TypeByAngle"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public TriangleTypeTriangleTypeByAngle TypeByAngle
```

Return Value

Type: [TriangleTypeTriangleTypeByAngle](#)

[Missing <returns> documentation for "M:is2GraphObject.TriangleType.TypeByAngle"]

▲ See Also

Reference

[TriangleType Class](#)

[is2GraphObject Namespace](#)

TriangleTypeTypeBySide Method

[Missing <summary> documentation for "M:is2GraphObject.TriangleType.TypeBySide"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public TriangleTypeTriangleTypeBySide TypeBySide()
```

Return Value

Type: [TriangleTypeTriangleTypeBySide](#)

[Missing <returns> documentation for "M:is2GraphObject.TriangleType.TypeBySide"]

▲ See Also

Reference

[TriangleType Class](#)

[is2GraphObject Namespace](#)

TriangleTypeTriangleTypeByAngle Enumeration

[Missing <summary> documentation for
"T:is2GraphObject.TriangleType.TriangleTypeByAngle"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version:
1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public enum TriangleTypeByAngle
```

▲ See Also

Reference

[is2GraphObject Namespace](#)

TriangleTypeTriangleTypeBySide Enumeration

[Missing <summary> documentation for
"T:is2GraphObject.TriangleType.TriangleTypeBySide"]

Namespace: [is2GraphObject](#)

Assembly: is2GraphObj_net4.5 (in is2GraphObj_net4.5.dll) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

C#

[Copy](#)

```
public enum TriangleTypeBySide
```

▲ See Also

Reference

[is2GraphObject Namespace](#)