Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AdvancedImageConverter</td>
<td>Provides a converter for Image instances that can preserve the original format of images better than the ImageConverter class when converting Bitmap, Metafile and Icon images.</td>
</tr>
</tbody>
</table>

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
AdvancedImageConverter Class

Provides a converter for Image instances that can preserve the original format of images better than the ImageConverter class when converting Bitmap, Metafile and Icon images.

Inheritance Hierarchy

SystemObject  System.ComponentModel.TypeConverter
   System.Drawing.ImageConverter
   KGySoft.ComponentModel.AdvancedImageConverter

Namespace: KGySoft.ComponentModel

Syntax

```csharp
public class AdvancedImageConverter : ImageConverter
```

The AdvancedImageConverter type exposes the following members.

Constructors

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AdvancedImageConverter</td>
<td>Initializes a new instance of the AdvancedImageConverter class</td>
</tr>
</tbody>
</table>

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CanConvertFrom(Type)</td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td>CanConvertFrom(ITypeDescriptorContext, Type)</td>
<td>(Inherited from ImageConverter.)</td>
</tr>
<tr>
<td>CanConvertTo(Type)</td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td>Function</td>
<td>Inheritance</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>CanConvertTo(ITypeDescriptorContext, Type)</td>
<td>Inherited from ImageConverter.</td>
</tr>
<tr>
<td>ConvertFrom(Object)</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>ConvertFrom(ITypeDescriptorContext, CultureInfo, Object)</td>
<td>Converts a specified object to an Image.  (Overrides ImageConverter.ConvertFrom(ITypeDescriptorContext, CultureInfo, Object).)</td>
</tr>
<tr>
<td>ConvertFromInvariantString(String)</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>ConvertFromInvariantString(ITypeDescriptorContext, String)</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>ConvertFromString(String)</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>ConvertFromString(ITypeDescriptorContext, String)</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>ConvertFromString(ITypeDescriptorContext, CultureInfo, String)</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>ConvertTo(Object, Type)</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>ConvertTo(ITypeDescriptorContext, CultureInfo, Object, Type)</td>
<td>Converts an Image (or an object that can be Image) to the specified type.  (Overrides ImageConverter.ConvertTo(ITypeDescriptorContext, CultureInfo, Object, Type).)</td>
</tr>
<tr>
<td>ConvertToInvariantString(Object)</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>ConvertToInvariantString(ITypeDescriptorContext, Object)</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>ConvertToString(Object)</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>ConvertToString(ITypeDescriptorContext, Object)</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>CreateInstance(IDictionary)</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>CreateInstance(ITypeDescriptorContext, IDictionary)</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>Equals</td>
<td>Inherited from Object.</td>
</tr>
<tr>
<td>Finalize</td>
<td>Inherited from Object.</td>
</tr>
<tr>
<td>GetConvertFromException</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>GetConvertToException</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>GetCreateInstanceSupported</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>GetCreateInstanceSupported(ITypeDescriptorContext)</td>
<td>Inherited from TypeConverter.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>GetProperties(Object)</td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td>GetProperties(ITypeDescriptorContext, Object)</td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td>GetProperties(ITypeDescriptorContext, Object, Attribute)</td>
<td>(Inherited from ImageConverter.)</td>
</tr>
<tr>
<td>GetPropertiesSupported</td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td>GetPropertiesSupported(ITypeDescriptorContext)</td>
<td>(Inherited from ImageConverter.)</td>
</tr>
<tr>
<td>GetStandardValues</td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td>GetStandardValues(ITypeDescriptorContext)</td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td>GetStandardValuesExclusive</td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td>GetStandardValuesExclusive(ITypeDescriptorContext)</td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td>GetStandardValuesSupported</td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td>GetStandardValuesSupported(ITypeDescriptorContext)</td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td>GetType</td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>IsValid(Object)</td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td>IsValid(ITypeDescriptorContext, Object)</td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>SortProperties</td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td>ToString</td>
<td>(Inherited from Object.)</td>
</tr>
</tbody>
</table>

**See Also**

Reference
KGySoft.ComponentModel Namespace

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](kgysoftdocs) page.

Copyright © KGy SOFT. All rights reserved.
AdvancedImageConverter Constructor

Initializes a new instance of the `AdvancedImageConverter` class

**Namespace:** KGySoft.ComponentModel

**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public AdvancedImageConverter()
```

**See Also**

Reference

- `AdvancedImageConverter` Class
- KGySoft.ComponentModel Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
The `AdvancedImageConverter` type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="#" alt="CanConvertFrom" /> Type</td>
<td>(Inherited from <code>TypeConverter</code>.)</td>
</tr>
<tr>
<td><img src="#" alt="CanConvertFrom" /> ITypeDescriptorContext, Type</td>
<td>(Inherited from <code>ImageConverter</code>.)</td>
</tr>
<tr>
<td><img src="#" alt="CanConvertTo" /> Type</td>
<td>(Inherited from <code>TypeConverter</code>.)</td>
</tr>
<tr>
<td><img src="#" alt="CanConvertTo" /> ITypeDescriptorContext, Type</td>
<td>(Inherited from <code>ImageConverter</code>.)</td>
</tr>
<tr>
<td><img src="#" alt="ConvertFrom" /> Object</td>
<td>(Inherited from <code>TypeConverter</code>.)</td>
</tr>
<tr>
<td><img src="#" alt="ConvertFrom" /> ITypeDescriptorContext, CultureInfo, Object</td>
<td>Converts a specified object to an <code>Image</code>. (Overrider <code>ImageConverter.ConvertFrom(ITypeDescriptorContext, CultureInfo, Object)</code>.)</td>
</tr>
<tr>
<td><img src="#" alt="ConvertFromInvariantString" /> String</td>
<td>(Inherited from <code>TypeConverter</code>.)</td>
</tr>
<tr>
<td><img src="#" alt="ConvertFromInvariantString" /> ITypeDescriptorContext, String</td>
<td>(Inherited from <code>TypeConverter</code>.)</td>
</tr>
<tr>
<td><img src="#" alt="ConvertFromString" /> String</td>
<td>(Inherited from <code>TypeConverter</code>.)</td>
</tr>
<tr>
<td><img src="#" alt="ConvertFromString" /> ITypeDescriptorContext, String</td>
<td>(Inherited from <code>TypeConverter</code>.)</td>
</tr>
<tr>
<td><img src="#" alt="ConvertFromString" /> ITypeDescriptorContext, CultureInfo, String</td>
<td>(Inherited from <code>TypeConverter</code>.)</td>
</tr>
<tr>
<td><img src="#" alt="ConvertTo" /> Object, Type</td>
<td>(Inherited from <code>TypeConverter</code>.)</td>
</tr>
<tr>
<td><img src="#" alt="ConvertTo" /> ITypeDescriptorContext, CultureInfo, Object, Type</td>
<td>Converts an <code>Image</code> (or an object that can be <code>Image</code>) to the specified type. (Overrider <code>ImageConverter.ConvertTo(ITypeDescriptorContext, CultureInfo, Object, Type)</code>.)</td>
</tr>
<tr>
<td><img src="#" alt="ConvertToInvariantString" /> Object</td>
<td>(Inherited from <code>TypeConverter</code>.)</td>
</tr>
<tr>
<td><img src="#" alt="ConvertToInvariantString" /> ITypeDescriptorContext, Object</td>
<td>(Inherited from <code>TypeConverter</code>.)</td>
</tr>
<tr>
<td>Method</td>
<td>Inherited From</td>
</tr>
<tr>
<td>----------------------------------------------------</td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td>ConvertToString(Object)</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>ConvertToString(ITypeDescriptorContext, Object)</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>ConvertToString(ITypeDescriptorContext, CultureInfo, Object)</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>CreateInstance(IDictionary)</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>CreateInstance(ITypeDescriptorContext, IDictionary)</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>Equals</td>
<td>Object</td>
</tr>
<tr>
<td>Finalize</td>
<td>Object</td>
</tr>
<tr>
<td>GetConvertFromException</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>GetConvertToException</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>GetCreateInstanceSupported</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>GetCreateInstanceSupported(ITypeDescriptorContext)</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>Object</td>
</tr>
<tr>
<td>GetProperties(Object)</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>GetProperties(ITypeDescriptorContext, Object)</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>GetProperties(ITypeDescriptorContext, Object, Attribute)</td>
<td>ImageConverter</td>
</tr>
<tr>
<td>GetPropertiesSupported</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>GetPropertiesSupported(ITypeDescriptorContext)</td>
<td>ImageConverter</td>
</tr>
<tr>
<td>GetStandardValues</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>GetStandardValues(ITypeDescriptorContext)</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>GetStandardValuesExclusive</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>GetStandardValuesExclusive(ITypeDescriptorContext)</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>GetStandardValuesSupported</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>GetStandardValuesSupported(ITypeDescriptorContext)</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>GetType</td>
<td>Object</td>
</tr>
<tr>
<td>IsValid(Object)</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>IsValid(ITypeDescriptorContext, Object)</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>MemberwiseClone</td>
<td>Object</td>
</tr>
<tr>
<td>SortProperties</td>
<td>TypeConverter</td>
</tr>
<tr>
<td>ToString</td>
<td>Object</td>
</tr>
</tbody>
</table>
AdvancedImageConverter

**ConvertFrom Method**

**Overload List**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="convertfrom.png" alt="Icon" /> <strong>ConvertFrom(Object)</strong></td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td><img src="convertfrom.png" alt="Icon" /> <strong>ConvertFrom(ITypeDescriptorContext, CultureInfo, Object)</strong></td>
<td>Converts a specified object to an Image. (Overrides ImageConverterConvertFrom(ITypeDescriptorContext, CultureInfo, Object).)</td>
</tr>
</tbody>
</table>

**See Also**

Reference

AdvancedImageConverter Class
KGySoft.ComponentModel Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
AdvancedImageConverter

Converts a specified object to an Image.

Namespace: KGySoft.ComponentModel

Syntax

```csharp
public override Object ConvertFrom(
    ITypeDescriptorContext context,
    CultureInfo culture,
    Object value
)
```

Parameters

- **context**
  Type: System.ComponentModel.ITypeDescriptorContext
  An ITypeDescriptorContext that provides a format context. In this converter this parameter is ignored.

- **culture**
  Type: System.Globalization.CultureInfo
  A CultureInfo. In this converter this parameter is ignored.

- **value**
  Type: System.Object
  The Object to be converted.

Return Value

Type: Object
If this method succeeds, it returns the Image that it created by converting the specified object. Otherwise, it throws an exception.

See Also
Reference

AdvancedImageConverter Class
ConvertFrom Overload
KGySoft.ComponentModel Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
## AdvancedImageConverter ConvertTo Method

### Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConvertTo(Object, Type)</td>
<td>(Inherited from TypeConverter.)</td>
</tr>
<tr>
<td>ConvertTo(ITypeDescriptorContext, CultureInfo, Object, Type)</td>
<td>Converts an Image (or an object that can be cast to an Image) to the specified type. (Overides ImageConverterConvertTo(ITypeDescriptorContext, CultureInfo, Object, Type).)</td>
</tr>
</tbody>
</table>

### See Also

- **Reference**
  - AdvancedImageConverter Class
  - KGySoft.ComponentModel Namespace

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](#) page.

Copyright © KGy SOFT. All rights reserved.
AdvancedImageConverter ConvertTo Method (ITypeDescriptorContext, CultureInfo, Object, Type)

Converts an Image (or an object that can be cast to an Image) to the specified type.

Namespace: KGySoft.ComponentModel

**Syntax**

```csharp
public override Object ConvertTo(ITypeDescriptorContext context, CultureInfo culture, Object value, Type destinationType)
```

**Parameters**

- **context**
  - Type: System.ComponentModel.ITypeDescriptorContext
  - An ITypeDescriptorContext that provides a format context. In this converter this parameter is ignored.

- **culture**
  - Type: System.Globalization.CultureInfo
  - A CultureInfo. In this converter this parameter is ignored.

- **value**
  - Type: System.Object
  - The Image to convert.

- **destinationType**
  - Type: System.Type
  - The Type to convert the Image to. This type converter supports byte[] type.
Return Value
Type: Object
An Object that represents the converted value.

See Also

Reference
AdvancedImageConverter Class
ConvertTo Overload
KGySoft.ComponentModel Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
The **KGySoft.Drawing** namespace contains extension methods and types built around the types of the **System.Drawing** namespace. Among others, provides advanced support for the **Icon** type such as extracting, combining and converting multi-resolution icons, including hi-resolution ones, supports saving several **Image** formats without built-in encoders (TIFF, GIF, Icon, EMF and WMF), provides pixel format conversion with arbitrary palette and preserving transparency, etc.

### Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BitmapExtensions</td>
<td>Contains extension methods for the <strong>Bitmap</strong> type.</td>
</tr>
<tr>
<td>CursorHandle</td>
<td>Represents a windows cursor. The <strong>CursorHandle</strong> instance can be passed to the <strong>System.Windows.Forms.Cursor</strong> constructor to create a new cursor.</td>
</tr>
<tr>
<td>GraphicsExtensions</td>
<td>Contains extension methods for the <strong>Graphics</strong> type.</td>
</tr>
<tr>
<td>IconExtensions</td>
<td>Contains extension methods for the <strong>Icon</strong> type.</td>
</tr>
<tr>
<td>Icons</td>
<td>Provides some icon-related methods as well as properties returning general icons in multi resolution. Unlike <strong>SystemIcons</strong>, these icons should be disposed when not used any more.</td>
</tr>
<tr>
<td>ImageExtensions</td>
<td>Contains extension methods for the <strong>Image</strong> type.</td>
</tr>
<tr>
<td>MetafileExtensions</td>
<td>Contains extension methods for the <strong>Metafile</strong> type.</td>
</tr>
</tbody>
</table>

### Enumerations

<table>
<thead>
<tr>
<th>Enumeration</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>StockIcon</td>
<td>Represents the Windows stock icons can be retrieved by the <strong>Icons.GetStockIcon</strong> method in Windows Vista and above. See also the <strong>SHSTOCKICONID Enumeration</strong> at the Microsoft Docs site.</td>
</tr>
<tr>
<td>SystemIconSize</td>
<td>Represents the predefined system icon sizes.</td>
</tr>
</tbody>
</table>
Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Contains extension methods for the `Bitmap` type.

### Inheritance Hierarchy

```
System
    Object
    KGySoft.Drawing
        BitmapExtensions
```

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll)  
**Version:** 4.6.1

### Syntax

C# | VB | C++ | F#
---|----|-----|----

```csharp
public static class BitmapExtensions
```

The `BitmapExtensions` type exposes the following members.

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![ rosa ] CloneCurrentFrame</td>
<td>Creates a clone of the current frame of the provided <code>Bitmap</code> instance. Unlike the <code>Bitmap(Image)</code> constructor, this method preserves original pixel format, and unlike <code>Bitmap.Clone(Rectangle,PixelFormat)</code> method, this method returns a single frame image.</td>
</tr>
<tr>
<td>![ rosaline ] ExtractBitmaps</td>
<td>When <code>image</code> contains multiple pages, frames or multi-resolution sub-images, returns them as separated <code>Bitmap</code> instances. Otherwise, returns a new <code>Bitmap</code> with the copy of the original <code>image</code>.</td>
</tr>
<tr>
<td>![ oro ] GetColors</td>
<td>Gets the colors used in the defined <code>bitmap</code>. A limit can be defined in <code>maxColors</code>.</td>
</tr>
<tr>
<td>![ or ] Resize</td>
<td>Resizes the image with high quality. The result is always a 32 bit ARGB image.</td>
</tr>
<tr>
<td>![ or ] ToCursorHandle</td>
<td>Converts the provided <code>bitmap</code> to a <code>CursorHandle</code>, which can be passed to the <code>System.Windows.Forms.Cursor</code> constructor to create a new cursor.</td>
</tr>
</tbody>
</table>
See Also

Reference

KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
The **BitmapExtensions** type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CloneCurrentFrame</td>
<td>Creates a clone of the current frame of the provided Bitmap instance. Unlike the Bitmap(Image) constructor, this method preserves original pixel format, and unlike Bitmap.Clone(Rectangle,PixelFormat) method, this method returns a single frame image.</td>
</tr>
<tr>
<td>ExtractBitmaps</td>
<td>When image contains multiple pages, frames or multi-resolution sub-images, returns them as separated Bitmap instances. Otherwise, returns a new Bitmap with the copy of the original image.</td>
</tr>
<tr>
<td>GetColors</td>
<td>Gets the colors used in the defined bitmap. A limit can be defined in maxColors.</td>
</tr>
<tr>
<td>Resize</td>
<td>Resizes the image with high quality. The result is always a 32 bit ARGB image.</td>
</tr>
<tr>
<td>ToCursorHandle</td>
<td>Converts the provided bitmap to a CursorHandle, which can be passed to the System.Windows.Forms.Cursor constructor to create a new cursor.</td>
</tr>
</tbody>
</table>

## See Also

**Reference**

- **BitmapExtensions Class**
- **KGySoft.Drawing Namespace**

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](#) page.

Copyright © KGy SOFT. All rights reserved.
**BitmapExtensions.CloneCurrentFrame** Method

Creates a clone of the current frame of the provided `Bitmap` instance. Unlike the `Bitmap(Image)` constructor, this method preserves original pixel format, and unlike `Bitmap.Clone(Rectangle,PixelFormat)` method, this method returns a single frame image.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public static Bitmap CloneCurrentFrame(
    this Bitmap bitmap
)
```

**Parameters**

- `bitmap`  
  Type: `System.Drawing.Bitmap`  
  The bitmap to be cloned.

**Return Value**

Type: `Bitmap`  
A single frame `Bitmap` instance that has the same content and has the same pixel format as the current frame of the source bitmap.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type `Bitmap`. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

**See Also**

Reference
BitmapExtensions Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
When `image` contains multiple pages, frames or multi-resolution sub-images, returns them as separated `Bitmap` instances. Otherwise, returns a new `Bitmap` with the copy of the original `image`.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

#### C#  
```csharp
public static Bitmap[] ExtractBitmaps(
    this Bitmap image
)
```

**Parameters**

`image`  
Type: `System.Drawing.Bitmap`  
An `Image` instance, which may contain multiple pages, frames or multi-resolution sub-images.

**Return Value**

Type: `Bitmap`  
An array of `Bitmap` instances, which contains the images of the provided `image`.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type `Bitmap`. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

### See Also

**Reference**  
BitmapExtensions Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
### GetColors Method

Gets the colors used in the defined `bitmap`. A limit can be defined in `maxColors`.

**Namespace:** KGYSoft.Drawing  
**Assembly:** KGYSoft.Drawing (in KGYSoft.Drawing.dll) Version: 4.6.1

#### Syntax

```csharp
public static Color[] GetColors(
    this Bitmap bitmap,
    int maxColors = 0
)
```

#### Parameters

- **bitmap**  
  - Type: System.Drawing.Bitmap  
  - The bitmap to get its colors. When it is indexed, its palette is returned and `maxColors` is ignored.

- **maxColors** (Optional)  
  - Type: System.Int32  
  - A limit of the returned colors. This parameter is ignored for indexed bitmaps. Use 0 for no limit. This parameter is optional.  
  - Default value: 0.

#### Return Value

- Type: Color  
  - An array of Color entries.

#### Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type `Bitmap`. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).
Remarks
The method is optimized for Format32bppRgb and Format32bppArgb formats.

See Also
Reference
BitmapExtensions Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Resizes the image with high quality. The result is always a 32 bit ARGB image.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll)  
Version: 4.6.1

### Syntax

**C#**  
```csharp
public static Bitmap Resize(
    this Bitmap image,
    Size newSize,
    bool keepAspectRatio
)
```

**Parameters**

- **image**  
  Type: System.Drawing.Bitmap  
  The original image to resize

- **newSize**  
  Type: System.Drawing.Size  
  The requested new size.

- **keepAspectRatio**  
  Type: System.Boolean  
  Determines whether the source image should keep aspect ratio.

**Return Value**

Type: Bitmap  
A Bitmap instance with the new size.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type Bitmap. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension.
Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference

BitmapExtensions Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
**BitmapExtensions**

**ToCursorHandle Method**

Converts the provided *bitmap* to a *CursorHandle*, which can be passed to the *System.Windows.Forms.Cursor* constructor to create a new cursor.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

```csharp
public static CursorHandle ToCursorHandle(
    Bitmap bitmap,
    Point cursorHotspot = null
)
```

**Parameters**

*bitmap*

- Type: *System.Drawing.Bitmap*
  - The *Bitmap*, which should be converted to a cursor.

*cursorHotspot* (Optional)

- Type: *System.Drawing.Point*
  - The hotspot coordinates of the cursor. This parameter is optional.
  - Default value: 0; 0 (top-left corner)

**Return Value**

- Type: *CursorHandle*
  - A *CursorHandle* instance that can be used to create a *System.Windows.Forms.Cursor* instance.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type *Bitmap*. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).
**See Also**

**Reference**

- BitmapExtensions Class
- KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
CursorHandle Class

Represents a windows cursor. The CursorHandle instance can be passed to the System.Windows.Forms.Cursor constructor to create a new cursor.

Inheritance Hierarchy


Namespace: KGySoft.Drawing  

Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>VB</th>
<th>C++</th>
<th>F#</th>
</tr>
</thead>
<tbody>
<tr>
<td>public sealed class CursorHandle : SafeHandle</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The CursorHandle type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Close</td>
<td>(Inherited from SafeHandle.)</td>
</tr>
<tr>
<td>DangerousAddRef</td>
<td>(Inherited from SafeHandle.)</td>
</tr>
<tr>
<td>DangerousGetHandle</td>
<td>(Inherited from SafeHandle.)</td>
</tr>
<tr>
<td>DangerousRelease</td>
<td>(Inherited from SafeHandle.)</td>
</tr>
<tr>
<td>Dispose</td>
<td>(Inherited from SafeHandle.)</td>
</tr>
<tr>
<td>Equals</td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>GetType</td>
<td>(Inherited from Object.)</td>
</tr>
<tr>
<td>SetHandleAsInvalid</td>
<td>(Inherited from SafeHandle.)</td>
</tr>
</tbody>
</table>
## Operators

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(CursorHandle to IntPtr)</td>
<td>Performs an implicit conversion from <strong>CursorHandle</strong> to <strong>IntPtr</strong>.</td>
</tr>
</tbody>
</table>

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IsClosed</td>
<td>(Inherited from <strong>SafeHandle</strong>.)</td>
</tr>
<tr>
<td>IsInvalid</td>
<td>Gets whether the handle value is invalid. (Overrides <strong>SafeHandle.IsInvalid</strong>.)</td>
</tr>
</tbody>
</table>

## Remarks

A **CursorHandle** instance can be created from an **Icon** or **Bitmap** instance by using the **IconExtensions.ToCursorHandle** and **BitmapExtensions.ToCursorHandle** extension methods.

This class can be used to create a custom Windows Forms **System.Windows.Forms.Cursor**.

**Important**

Do keep a reference to this **CursorHandle** instance until the cursor is in use; otherwise, the cursor resources might be disposed too soon.

## See Also

**Reference**

**KGySoft.Drawing Namespace**

Find the complete KGy SOFT Libraries documentation at the **KGy SOFT Docs** page.

Copyright © KGy SOFT. All rights reserved.
The **CursorHandle** type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Close</td>
<td>(Inherited from <code>SafeHandle</code>.)</td>
</tr>
<tr>
<td>DangerousAddRef</td>
<td>(Inherited from <code>SafeHandle</code>.)</td>
</tr>
<tr>
<td>DangerousGetHandle</td>
<td>(Inherited from <code>SafeHandle</code>.)</td>
</tr>
<tr>
<td>DangerousRelease</td>
<td>(Inherited from <code>SafeHandle</code>.)</td>
</tr>
<tr>
<td>Dispose</td>
<td>(Inherited from <code>SafeHandle</code>.)</td>
</tr>
<tr>
<td>Equals</td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>GetHashCode</td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>GetType</td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
<tr>
<td>SetHandleAsInvalid</td>
<td>(Inherited from <code>SafeHandle</code>.)</td>
</tr>
<tr>
<td>ToString</td>
<td>(Inherited from <code>Object</code>.)</td>
</tr>
</tbody>
</table>

## See Also

**Reference**
- **CursorHandle Class**
- **KGySoft.Drawing Namespace**

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](https://kgysotf.com/docs) page.

Copyright © KGy SOFT. All rights reserved.
The **CursorHandle** type exposes the following members.

### Operators

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(CursorHandle to IntPtr)</td>
<td>Performs an implicit conversion from <strong>CursorHandle</strong> to <strong>IntPtr</strong>.</td>
</tr>
</tbody>
</table>

**See Also**

**Reference**

- **CursorHandle Class**
- **KGySoft.Drawing Namespace**

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](https://kgysoft.com) page.

Copyright © KGy SOFT. All rights reserved.
Perform an implicit conversion from `CursorHandle` to `IntPtr`.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

**C#**

```csharp
public static implicit operator IntPtr (CursorHandle cursorHandle)
```

**Parameters**

*cursorHandle*

Type: KGySoft.Drawing.CursorHandle  
The cursor handle.

**Return Value**

Type: `IntPtr`  
An `IntPtr` instance representing the native cursor handle.

### See Also

**Reference**  
*CursorHandle Class*  
*KGySoft.Drawing Namespace*

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.  
Copyright © KGy SOFT. All rights reserved.
The `CursorHandle` type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IsClosed</td>
<td>(Inherited from <code>SafeHandle</code>.)</td>
</tr>
<tr>
<td>IsInvalid</td>
<td>Gets whether the handle value is invalid. (Overrides <code>SafeHandle.IsInvalid</code>.)</td>
</tr>
</tbody>
</table>

## See Also

- **Reference**
  - `CursorHandle Class`
  - `KGySoft.Drawing Namespace`

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](#) page.

Copyright © KGy SOFT. All rights reserved.
CursorHandleIsInvalid Property

Gets whether the handle value is invalid.

Namespace: KGySoft.Drawing

Syntax

```csharp
public override bool IsInvalid { get; }
```

Property Value
Type: Boolean

See Also
Reference
CursorHandle Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
Contains extension methods for the Graphics type.

Inheritance Hierarchy

System\Object\ KGySoft.Drawing\GraphicsExtensions

Namespace: KGySoft.Drawing

Syntax

```
public static class GraphicsExtensions
```

The GraphicsExtensions type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DrawRoundedRectangle(Graphics, Pen, Rectangle, Int32)</td>
<td>Draws a rounded rectangle specified by a bounding Rectangle and a common corner radius value for each corners.</td>
</tr>
<tr>
<td>FillRoundedRectangle(Graphics, Brush, Rectangle, Int32)</td>
<td>Fills a rounded rectangle specified by a bounding Rectangle and a common corner radius value for each corners.</td>
</tr>
<tr>
<td>FillRoundedRectangle(Graphics, Brush, Rectangle, Int32, Int32, Int32)</td>
<td>Fills a rounded rectangle specified by a bounding Rectangle and four custom corner radius values.</td>
</tr>
<tr>
<td>ToBitmap</td>
<td>Copies the Graphics object provided in graphics parameter to a Bitmap instance.</td>
</tr>
</tbody>
</table>
See Also

Reference
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
The **GraphicsExtensions** type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>📚 DrawRoundedRectangle(Graphics, Pen, Rectangle, Int32)</td>
<td>Draws a rounded rectangle specified by a bounding Rectangle and a common corner radius value for each corners.</td>
</tr>
<tr>
<td>📚 FillRoundedRectangle(Graphics, Brush, Rectangle, Int32)</td>
<td>Fills a rounded rectangle specified by a bounding Rectangle and a common corner radius value for each corners.</td>
</tr>
<tr>
<td>📚 FillRoundedRectangle(Graphics, Brush, Rectangle, Int32, Int32, Int32, Int32)</td>
<td>Fills a rounded rectangle specified by a bounding Rectangle and four custom corner radius values.</td>
</tr>
<tr>
<td>📚 ToBitmap</td>
<td>Copies the Graphics object provided in graphics parameter to a Bitmap instance.</td>
</tr>
</tbody>
</table>

## See Also

**Reference**

- **GraphicsExtensions Class**
- **KGySoft.Drawing Namespace**

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](#) page.

Copyright © KGy SOFT. All rights reserved.
GraphicsExtensions\DrawRoundedRectangle Method

Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DrawRoundedRectangle(Graphics, Pen, Rectangle, Int32)</td>
<td>Draws a rounded rectangle specified by a bounding Rectangle and a common corner radius value for each corners.</td>
</tr>
</tbody>
</table>

See Also

Reference

GraphicsExtensions Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Draws a rounded rectangle specified by a bounding `Rectangle` and a common corner radius value for each corners.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll)  
**Version:** 4.6.1

### Syntax

```csharp
public static void DrawRoundedRectangle(
    this Graphics graphics,
    Pen pen,
    Rectangle bounds,
    int cornerRadius
)
```

### Parameters

**graphics**  
Type: `System.Drawing.Graphics`  
The `Graphics` instance to draw on.

**pen**  
Type: `System.DrawingPen`  
The `Pen` instance to be used for the drawing.

**bounds**  
Type: `System.DrawingRectangle`  
A `Rectangle` that bounds the rounded rectangle.

**cornerRadius**  
Type: `SystemInt32`  
Size of the corner radius for each corners.
Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type Graphics. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference
GraphicsExtensions Class
DrawRoundedRectangle Overload
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Draws a rounded rectangle specified by a bounding Rectangle and four corner radius values.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

```
public static void DrawRoundedRectangle(
    this Graphics graphics,
    Pen pen,
    Rectangle bounds,
    int radiusTopLeft,
    int radiusTopRight,
    int radiusBottomRight,
    int radiusBottomLeft
)
```

**Parameters**

- **graphics**
  - Type: System.Drawing.Graphics
  - The Graphics instance to draw on.

- **pen**
  - Type: System.Drawing.Pen
  - The Pen instance to be used for the drawing.

- **bounds**
  - Type: System.Drawing.Rectangle
  - A Rectangle that bounds the rounded rectangle.
radiusTopLeft
  Type: SystemInt32
  Size of the top-left radius.

radiusTopRight
  Type: SystemInt32
  Size of the top-right radius.

radiusBottomRight
  Type: SystemInt32
  Size of the bottom-right radius.

radiusBottomLeft
  Type: SystemInt32
  Size of the bottom-left radius.

Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type Graphics. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference
GraphicsExtensions Class
DrawRoundedRectangle Overload
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FillRoundedRectangle(Graphics, Brush, Rectangle, Int32)</td>
<td>Fills a rounded rectangle specified by a bounding Rectangle and a common corner radius value for each corners.</td>
</tr>
<tr>
<td>FillRoundedRectangle(Graphics, Brush, Rectangle, Int32, Int32, Int32, Int32)</td>
<td>Fills a rounded rectangle specified by a bounding Rectangle and four custom corner radius values.</td>
</tr>
</tbody>
</table>

See Also

Reference

GraphicsExtensions Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
GraphicsExtensions FillRoundedRectangle Method (Graphics, Brush, Rectangle, Int32)

Fills a rounded rectangle specified by a bounding Rectangle and a common corner radius value for each corners.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public static void FillRoundedRectangle(
    this Graphics graphics,
    Brush brush,
    Rectangle bounds,
    int cornerRadius
)
```

**Parameters**

- `graphics`  
  Type: System.Drawing.Graphics  
  The Graphics instance to draw on.

- `brush`  
  Type: System.Drawing.Brush  
  The Brush instance to be used for the drawing.

- `bounds`  
  Type: System.Drawing.Rectangle  
  A Rectangle that bounds the rounded rectangle.

- `cornerRadius`  
  Type: System.Int32  
  Size of the corner radius for each corners.

**Usage Note**
In Visual Basic and C#, you can call this method as an instance method on any object of type Graphics. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference

GraphicsExtensions Class
Fill RoundedRectangle Overload
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
Fills a rounded rectangle specified by a bounding Rectangle and four custom corner radius values.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

```csharp
public static void FillRoundedRectangle(
    this Graphics graphics,
    Brush brush,
    Rectangle bounds,
    int radiusTopLeft,
    int radiusTopRight,
    int radiusBottomRight,
    int radiusBottomLeft
)
```

**Parameters**

- **graphics**  
  Type: System.Drawing.Graphics  
  The Graphics instance to draw on.

- **brush**  
  Type: System.Drawing.Brush  
  The Brush instance to be used for the drawing.

- **bounds**  
  Type: System.Drawing.Rectangle  
  A Rectangle that bounds the rounded rectangle.

- **radiusTopLeft**
Type: `SystemInt32`  
Size of the top-left radius.

`radiusTopRight`  
Type: `SystemInt32`  
Size of the top-right radius.

`radiusBottomRight`  
Type: `SystemInt32`  
Size of the bottom-right radius.

`radiusBottomLeft`  
Type: `SystemInt32`  
Size of the bottom-left radius.

Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type `Graphics`. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods (Visual Basic)](https://docs.microsoft.com/en-us/dotnet/framework/design/extension-methods) or [Extension Methods (C# Programming Guide)](https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/).
**GraphicsExtensions**

**ToBitmap Method**

Copies the Graphics object provided in *graphics* parameter to a Bitmap instance.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```
public static Bitmap ToBitmap(
    this Graphics graphics,
    bool visibleClipOnly
)
```

**Parameters**

- **graphics**
  - Type: System.Drawing.Graphics  
  - The Graphics instance to be converted.

- **visibleClipOnly**
  - Type: System.Boolean  
  - When true, the result will contain only the area represented by VisibleClipBounds property. When false, the result will contain the image of the whole container source (when a container object is found), where the visible clip bounds can be identified by VisibleClipBounds in pixels.

**Return Value**

- Type: Bitmap  
  - A Bitmap object that contains the image content of the source Graphics object, or, when the required area of graphics is empty.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type Graphics. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension.
Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference
GraphicsExtensions Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
IconExtensions Class

Contains extension methods for the Icon type.

Inheritance Hierarchy

System\Object KGYSoft.Drawing\IconExtensions

Namespace: KGYSoft.Drawing

Syntax

```
public static class IconExtensions
```

The IconExtensions type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Combine(Icon, Bitmap)</td>
<td>Combines an Icon instance with the provided images into a multi-resolution Icon instance.</td>
</tr>
<tr>
<td>Combine(Icon, Icon)</td>
<td>Combines an Icon instance with the provided icons into a multi-resolution Icon instance.</td>
</tr>
<tr>
<td>Combine(Icon, Bitmap, Color)</td>
<td>Combines an Icon instance with the provided image into a multi-resolution Icon instance.</td>
</tr>
<tr>
<td>ExtractBitmap(Icon, Boolean)</td>
<td>Extracts the first image from an Icon instance. If the icon has only one image consider to use ToAlphaBitmap, which is faster.</td>
</tr>
<tr>
<td>ExtractBitmap(Icon, Size, Boolean)</td>
<td>Extracts the first image of specified size from an Icon instance.</td>
</tr>
<tr>
<td>ExtractBitmap(Icon, Int32, Boolean)</td>
<td>Extracts the image of specified index from an Icon instance.</td>
</tr>
<tr>
<td>ExtractBitmap(Icon, Size, PixelFormat, Boolean)</td>
<td>Extracts the image of specified size and pixel format from an Icon instance.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ExtractBitmaps(Icon, Boolean)</td>
<td>Extracts every image from an Icon instance.</td>
</tr>
<tr>
<td>ExtractBitmaps(Icon, PixelFormat, Boolean)</td>
<td>Extracts every image of specified pixel format from an Icon instance.</td>
</tr>
<tr>
<td>ExtractBitmaps(Icon, Size, Boolean)</td>
<td>Extracts every image of specified size from an Icon instance.</td>
</tr>
<tr>
<td>ExtractIcon(Icon, Size)</td>
<td>Extracts the first icon of specified size from an Icon instance. Unless the Icon constructors, this method works as expected.</td>
</tr>
<tr>
<td>ExtractIcon(Icon, Int32)</td>
<td>Extracts the icon of specified index from an Icon instance.</td>
</tr>
<tr>
<td>ExtractIcon(Icon, Size, PixelFormat)</td>
<td>Extracts the icon of specified size and pixel format from an Icon instance. Unless the Icon constructors, this method works as expected.</td>
</tr>
<tr>
<td>ExtractIcons(Icon)</td>
<td>Extracts every icon from an Icon instance as separated instances.</td>
</tr>
<tr>
<td>ExtractIcons(Icon, PixelFormat)</td>
<td>Extracts every icon of specified pixel format from an Icon instance as separated instances.</td>
</tr>
<tr>
<td>ExtractIcons(Icon, Size)</td>
<td>Extracts every icon of specified size from an Icon instance as separated instances.</td>
</tr>
<tr>
<td>ExtractNearestBitmap</td>
<td>Extracts the nearest image of specified size and pixel format from an Icon instance.</td>
</tr>
<tr>
<td>ExtractNearestIcon</td>
<td>Extracts the nearest icon of specified size and pixel format from an Icon instance. Unless the Icon constructors, this method works as expected.</td>
</tr>
<tr>
<td>GetImagesCount</td>
<td>Gets the number of images in the icon.</td>
</tr>
<tr>
<td>SaveHighQuality</td>
<td>Saves the icon into the specified stream. Unlike Icon.Save, this method can save every icon with high quality, even SystemIcons members, and icons created by the Icon.FromHandle method.</td>
</tr>
<tr>
<td>ToAlphaBitmap</td>
<td>Converts the specified icon to a Bitmap. While Icon.ToBitmap may return a wrong result when icon contains semi-transparent pixels, this method returns an image, in which alpha channel is always correctly applied for the image.</td>
</tr>
<tr>
<td>ToCursorHandle</td>
<td>Converts the provided icon to a CursorHandle, which can be passed to the System.Windows.Forms.Cursor constructor to create a new cursor.</td>
</tr>
<tr>
<td>ToMultiResBitmap</td>
<td>Converts the icon to a Bitmap instance, which contains every image of the icon. When the returned Bitmap is used to create another Bitmap or is drawn into a Graphics, the best-fitting image is automatically applied.</td>
</tr>
</tbody>
</table>
See Also

KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
The **IconExtensions** type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Combine(Icon, Bitmap)</td>
<td>Combines an Icon instance with the provided images into a multi-resolution Icon instance.</td>
</tr>
<tr>
<td>Combine(Icon, Icon)</td>
<td>Combines an Icon instance with the provided icons into a multi-resolution Icon instance.</td>
</tr>
<tr>
<td>Combine(Icon, Bitmap, Color)</td>
<td>Combines an Icon instance with the provided image into a multi-resolution Icon instance.</td>
</tr>
<tr>
<td>ExtractBitmap(Icon, Boolean)</td>
<td>Extracts the first image from an Icon instance. If the icon has only one image consider to use ToAlphaBitmap, which is faster.</td>
</tr>
<tr>
<td>ExtractBitmap(Icon, Size, Boolean)</td>
<td>Extracts the first image of specified size from an Icon instance.</td>
</tr>
<tr>
<td>ExtractBitmap(Icon, Int32, Boolean)</td>
<td>Extracts the image of specified index from an Icon instance.</td>
</tr>
<tr>
<td>ExtractBitmap(Icon, Size, PixelFormat, Boolean)</td>
<td>Extracts the image of specified size and pixel format from an Icon instance.</td>
</tr>
<tr>
<td>ExtractBitmaps(Icon, Boolean)</td>
<td>Extracts every image from an Icon instance.</td>
</tr>
<tr>
<td>ExtractBitmaps(Icon, PixelFormat, Boolean)</td>
<td>Extracts every image of specified pixel format from an Icon instance.</td>
</tr>
<tr>
<td>ExtractBitmaps(Icon, Size, Boolean)</td>
<td>Extracts every image of specified size from an Icon instance.</td>
</tr>
<tr>
<td>ExtractIcon(Icon, Size)</td>
<td>Extracts the first icon of specified size from an Icon instance. Unless the Icon constructors, this method works as expected.</td>
</tr>
<tr>
<td>ExtractIcon(Icon, Int32)</td>
<td>Extracts the icon of specified index from an Icon instance.</td>
</tr>
<tr>
<td>ExtractIcon(Icon, Size, PixelFormat)</td>
<td>Extracts the icon of specified size and pixel format from an Icon instance. Unless the Icon constructors, this method works as</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>ExtractIcons(Icon)</td>
<td>Extracts every icon from an Icon instance as separated instances.</td>
</tr>
<tr>
<td>ExtractIcons(Icon, PixelFormat)</td>
<td>Extracts every icon of specified pixel format from an Icon instance as separated instances.</td>
</tr>
<tr>
<td>ExtractIcons(Icon, Size)</td>
<td>Extracts every icon of specified size from an Icon instance as separated instances.</td>
</tr>
<tr>
<td>ExtractNearestBitmap</td>
<td>Extracts the nearest image of specified size and pixel format from an Icon instance.</td>
</tr>
<tr>
<td>ExtractNearestIcon</td>
<td>Extracts the nearest icon of specified size and pixel format from an Icon instance. Unless the Icon constructors, this method works as expected.</td>
</tr>
<tr>
<td>GetImagesCount</td>
<td>Gets the number of images in the Icon.</td>
</tr>
<tr>
<td>SaveHighQuality</td>
<td>Saves the Icon into the specified stream. Unlike Icon.Save, this method can save every icon with high quality, even SystemIcons members, and icons created by the Icon.FromHandle method.</td>
</tr>
<tr>
<td>ToAlphaBitmap</td>
<td>Converts the specified icon to a Bitmap. While Icon.ToBitmap may return a wrong result when icon contains semi-transparent pixels, this method returns an image, in which alpha channel is always correctly applied for the image.</td>
</tr>
<tr>
<td>ToCursorHandle</td>
<td>Converts the provided icon to a CursorHandle, which can be passed to the System.Windows.Forms.Cursor constructor to create a new cursor.</td>
</tr>
<tr>
<td>ToMultiResBitmap</td>
<td>Converts the icon to a Bitmap instance, which contains every image of the icon. When the returned Bitmap is used to create another Bitmap or is drawn into a Graphics, the best-fitting image is automatically applied.</td>
</tr>
</tbody>
</table>

**See Also**

Reference

IconExtensions Class

KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
## IconExtensions.Combine Method

### Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Combine(Icon, Bitmap)</td>
<td>Combines an Icon instance with the provided images into a multi-resolution Icon instance.</td>
</tr>
<tr>
<td>Combine(Icon, Icon)</td>
<td>Combines an Icon instance with the provided icons into a multi-resolution Icon instance.</td>
</tr>
<tr>
<td>Combine(Icon, Bitmap, Color)</td>
<td>Combines an Icon instance with the provided image into a multi-resolution Icon instance.</td>
</tr>
</tbody>
</table>

---

### See Also

- **Reference**
  - IconExtensions Class
  - KGySoft.Drawing Namespace

---

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](#) page.

Copyright © KGy SOFT. All rights reserved.
**IconExtensionsCombine Method (Icon, Bitmap)**

Combines an `Icon` instance with the provided `images` into a multi-resolution `Icon` instance.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll)  
Version: 4.6.1

### Syntax

```csharp
public static Icon Combine(
    this Icon icon,
    params Bitmap[] images
)
```

### Parameters

- **icon**  
  Type: `System.DrawingIcon`  
  The icon to combine with other images.

- **images**  
  Type: `System.DrawingBitmap`  
  The images to be added to the `icon`. Images can be non-squared ones.

### Return Value

Type: `Icon`  
An `Icon` instance that contains every image of the source `images`.

### Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type `Icon`. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods (Visual Basic)](C#) or [Extension Methods (C# Programming Guide)](C#).

### Remarks

`icon` may already contain multiple icons.
See Also

Reference

IconExtensions Class
Combine Overload
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
IconExtensions

Combine Method
(Icon, Icon)

Combines an Icon instance with the provided icons into a multi-resolution Icon instance.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll)  
**Version:** 4.6.1

**Syntax**

```csharp
public static Icon Combine(  
    this Icon icon,  
    params Icon[] icons  
)
```

**Parameters**

- `icon`
  - Type: System.DrawingIcon
  - The icon to combine with other icons.

- `icons`
  - Type: System.DrawingIcon
  - The icons to be combined with the specified icon.

**Return Value**

- Type: Icon
  - An Icon instance that contains every image of the source icons.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type Icon. When you use instance method syntax to call this method, omit the first parameter. For more information, see [Extension Methods (Visual Basic)](https://docs.microsoft.com/en-us/dotnet/visual-basic/programming-guide/extension-methods) or [Extension Methods (C# Programming Guide)](https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/extension-methods).

**Remarks**

Both icon and elements of icons may contain multiple icons.
See Also

Reference
IconExtensions Class
Combine Overload
KGYSoft.Drawing Namespace

Find the complete KGY SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGY SOFT. All rights reserved.
IconExtensions.Combine Method
(Icon, Bitmap, Color)

Combines an Icon instance with the provided image into a multi-resolution Icon instance.

Namespace: KGySoft.Drawing

Syntax

public static Icon Combine(
    this Icon icon,
    Bitmap image,
    Color transparentColor
)

Parameters

icon
Type: System.Drawing.Icon
The icon to combine with other images.

image
Type: System.Drawing.Bitmap
The image to be added to the icon. Can be a non-squared one.

transparentColor
Type: System.Drawing.Color
A color that represents the transparent color in image.

Return Value
Type: Icon
An Icon instance that contains the source image.

Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type Icon. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension
Remarks

*icon* may already contain multiple icons.

See Also

Reference

IconExtensions Class
Combine Overload
KgySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
### IconExtensions\ExtractBitmap Method

#### Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="ExtractBitmap(Icon, Boolean)" /></td>
<td>Extracts the first image from an <a href="#">Icon</a> instance. If the icon has only one image consider to use <a href="#">ToAlphaBitmap</a>, which is faster.</td>
</tr>
<tr>
<td><img src="image" alt="ExtractBitmap(Icon, Size, Boolean)" /></td>
<td>Extracts the first image of specified size from an <a href="#">Icon</a> instance.</td>
</tr>
<tr>
<td><img src="image" alt="ExtractBitmap(Icon, Int32, Boolean)" /></td>
<td>Extracts the image of specified index from an <a href="#">Icon</a> instance.</td>
</tr>
<tr>
<td><img src="image" alt="ExtractBitmap(Icon, Size, PixelFormat, Boolean)" /></td>
<td>Extracts the image of specified size and pixel format from an <a href="#">Icon</a> instance.</td>
</tr>
</tbody>
</table>

#### See Also

**Reference**

- IconExtensions Class
- KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
Extracts the first image from an Icon instance. If the icon has only one image consider to use ToAlphaBitmap, which is faster.

**Namespace:** KGySoft.Drawing

**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public static Bitmap ExtractBitmap(
    this Icon icon,
    bool keepOriginalFormat = false
)
```

**Parameters**

- **icon**
  Type: System.DrawingIcon
  The icon that may contain multiple images.

- **keepOriginalFormat** (Optional)
  Type: SystemBoolean
  If , keeps the original image format stored in the icon. Possible transparent pixels of non-32 bpp ARGB formats may be black. If , always returns 32 bpp images with transparency. This parameter is optional.
  Default value:  

**Return Value**

Type: Bitmap

An Bitmap instance, which was extracted from the icon, or if no image was found in the icon.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type Icon. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension
Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference
IconExtensions Class
ExtractBitmap Overload
KGySoft.Drawing Namespace
IconExtensionsToAlphaBitmap(Icon)

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Extracts the first image of specified size from an `Icon` instance.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

```csharp
public static Bitmap ExtractBitmap(
    this Icon icon,
    Size size,
    bool keepOriginalFormat = false
)
```

**Parameters**

- `icon`  
  Type: `System.Drawing.Icon`  
  The icon that may contain multiple images.

- `size`  
  Type: `System.Drawing.Size`  
  The required icon size to retrieve.

- `keepOriginalFormat` (Optional)  
  Type: `System.Boolean`  
  If true, keeps the original image format stored in the `icon`. Possible transparent pixels of non-32 bpp ARGB formats may be black. If false, always returns 32 bpp images with transparency. This parameter is optional.
  Default value: false

**Return Value**

Type: `Bitmap`  
An `Bitmap` instance, which was extracted from the `icon`, or if no icon found with the specified size.
In Visual Basic and C#, you can call this method as an instance method on any object of type Icon. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

**See Also**

**Reference**

IconExtensions Class
ExtractBitmap Overload
KGySoft.Drawing Namespace
IconExtensionsExtractIcon(Icon, Size)

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
IconExtensionsExtractBitmap

Method (Icon, Int32, Boolean)

Extracts the image of specified index from an Icon instance.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public static Bitmap ExtractBitmap(
    this Icon icon,
    int index,
    bool keepOriginalFormat = false
)
```

**Parameters**

- **icon**  
  Type: **System.DrawingIcon**  
  The icon that may contain multiple images.

- **index**  
  Type: **System.Int32**  
  The zero-based index of the icon image to retrieve.

- **keepOriginalFormat** (Optional)  
  Type: **System.Boolean**  
  If , keeps the original image format stored in the icon. Possible transparent pixels of non-32 bpp ARGB formats may be black. If , always returns 32 bpp images with transparency. This parameter is optional.  
  Default value: .

**Return Value**

Type: **Bitmap**  
An Bitmap instance, which was extracted from the icon, or if the specified index was too large.

**Usage Note**
In Visual Basic and C#, you can call this method as an instance method on any object of type Icon. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

**See Also**

**Reference**

IconExtensions Class
ExtractBitmap Overload
KGySoft.Drawing Namespace
IconExtensionsExtractIcon(Icon, Int32)

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
IconExtensionsExtractBitmap Method
(Icon, Size, PixelFormat, Boolean)

Extracts the image of specified size and pixel format from an Icon instance.

Namespace: KGySoft.Drawing

Syntax

**C#**
```csharp
public static Bitmap ExtractBitmap(
    this Icon icon,
    Size size,
    PixelFormat pixelFormat,
    bool keepOriginalFormat = false
)
```

**Parameters**

*icon*
- Type: System.Drawing.Icon
  - The icon that may contain multiple images.

*size*
- Type: System.Drawing.Size
  - The required icon size to retrieve.

*pixelFormat*
- Type: System.Drawing.Imaging.PixelFormat
  - The required pixel format to retrieve

*keepOriginalFormat* (Optional)
- Type: System.Boolean
  - If , keeps the original image format stored in the icon. Possible transparent pixels of non-32 bpp ARGB formats may be black. If , always returns 32 bpp images with transparency. This parameter is optional.
  - Default value: .
Return Value
Type: Bitmap
An Bitmap instance, which was extracted from the icon, or if no icon found with the specified size and format.

Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type Icon. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also
Reference
IconExtensions Class
ExtractBitmap Overload
KGySoft.Drawing Namespace
IconExtensionsExtractIcon(Icon, Size, PixelFormat)

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
IconExtensions
ExtractBitmaps Method

Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExtractBitmaps(Icon, Boolean)</td>
<td>Extracts every image from an Icon instance.</td>
</tr>
<tr>
<td>ExtractBitmaps(Icon, PixelFormat, Boolean)</td>
<td>Extracts every image of specified pixel format from an Icon instance.</td>
</tr>
<tr>
<td>ExtractBitmaps(Icon, Size, Boolean)</td>
<td>Extracts every image of specified size from an Icon instance.</td>
</tr>
</tbody>
</table>

See Also

Reference

IconExtensions Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
Extracts every image from an Icon instance.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

**C#**

```csharp
public static Bitmap[] ExtractBitmaps(
    this Icon icon,
    bool keepOriginalFormat = false
)
```

**Parameters**

**icon**
- Type: `System.Drawing.Icon`
  - The icon that may contain multiple images.

`keepOriginalFormat` (Optional)
- Type: `System.Boolean`
  - If `true`, keeps the original image format stored in the `icon`. Possible transparent pixels of non-32 bpp ARGB formats may be black. If `false`, always returns 32 bpp images with transparency. This parameter is optional.
  - Default value: `false`

**Return Value**

- Type: `Bitmap`
  - An array of `Bitmap` instances, which were extracted from the `icon`.

### Usage Note

See Also

Reference

IconExtensions Class
ExtractBitmaps Overload
KGySoft.Drawing Namespace
IconExtensionsExtractIcons(Icon)

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Extracts every image of specified pixel format from an Icon instance.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

```
public static Bitmap[] ExtractBitmaps(
    this Icon icon,
    PixelFormat pixelFormat,
    bool keepOriginalFormat = false
)
```

#### Parameters

- **icon**
  - Type: **System.Drawing.Icon**
  - The icon that may contain multiple images.

- **pixelFormat**
  - Type: **System.Drawing.Imaging.PixelFormat**
  - The required pixel format to retrieve

- **keepOriginalFormat** (Optional)
  - Type: **SystemBoolean**
  - If true, keeps the original image format stored in the icon. Possible transparent pixels of non-32 bpp ARGB formats may be black. If true, always returns 32 bpp images with transparency. This parameter is optional.
    - Default value: false

#### Return Value

- **Type:** **Bitmap**
- An array of Bitmap instances, which were extracted from the icon.

### Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type `Icon`. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

**See Also**

**Reference**

- IconExtensions Class
- ExtractBitmaps Overload
- KGySoft.Drawing Namespace
- IconExtensionsExtractIcons(Icon, PixelFormat)

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](#) page.

Copyright © KGy SOFT. All rights reserved.
Extracts every image of specified size from an `Icon` instance.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

```csharp
public static Bitmap[] ExtractBitmaps(
    this Icon icon,
    Size size,
    bool keepOriginalFormat = false
)
```

#### Parameters

- **icon**  
  Type: `System.Drawing.Icon`  
  The icon that may contain multiple images.

- **size**  
  Type: `System.Drawing.Size`  
  The required icon size to retrieve.

- **keepOriginalFormat** (Optional)  
  Type: `System.Boolean`  
  If , keeps the original image format stored in the `icon`. Possible transparent pixels of non-32 bpp ARGB formats may be black. If , always returns 32 bpp images with transparency. This parameter is optional.  
  Default value: .

#### Return Value

Type: `Bitmap`  
An array of `Bitmap` instances, which were extracted from the `icon`.

### Usage Note

In Visual Basic and C#, you can call this method as an instance method on
any object of type Icon. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference

IconExtensions Class
ExtractBitmaps Overload
KGySoft.Drawing Namespace
IconExtensionsExtractIcons(Icon, Size)

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
**IconExtensions ExtractIcon**

**KGY SOFT Drawing Libraries Help**

## Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExtractIcon(Icon, Size)</td>
<td>Extracts the first icon of specified size from an Icon instance. Unless the Icon constructors, this method works as expected.</td>
</tr>
<tr>
<td>ExtractIcon(Icon, Int32)</td>
<td>Extracts the icon of specified index from an Icon instance.</td>
</tr>
<tr>
<td>ExtractIcon(Icon, Size, PixelFormat)</td>
<td>Extracts the icon of specified size and pixel format from an Icon instance. Unless the Icon constructors, this method works as expected.</td>
</tr>
</tbody>
</table>

**Top**

**See Also**

**Reference**

IconExtensions Class
KGYSoft.Drawing Namespace

Find the complete KGY SOFT Libraries documentation at the [KGY SOFT Docs](#) page.

Copyright © KGY SOFT. All rights reserved.
Extracts the first icon of specified size from an `Icon` instance. Unless the `Icon` constructors, this method works as expected.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

```csharp
public static Icon ExtractIcon(
    this Icon icon,
    Size size
)
```

**Parameters**

- `icon`  
  Type: `System.DrawingIcon`  
  The icon that may contain multiple images.

- `size`  
  Type: `System.DrawingSize`  
  The required icon size to retrieve.

**Return Value**

Type: `Icon`  
An `Icon` instance, which contains only a single image, or if no icon found with the specified size.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type `Icon`. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

### See Also
Reference
IconExtensions Class
ExtractIcon Overload
KGySoft.Drawing Namespace
IconExtensionsExtractBitmap(Icon, Size, Boolean)

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Extracts the icon of specified index from an Icon instance.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

```
public static Icon ExtractIcon(  
    this Icon icon,  
    int index
)
```

### Parameters

- **icon**
  - Type: System.DrawingIcon  
  - The icon that may contain multiple images.

- **index**
  - Type: SystemInt32  
  - The zero-based index of the icon image to retrieve.

### Return Value

Type: Icon  
An Icon instance, which contains only a single image, or if the specified index was too large.

### Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type Icon. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

### See Also

Reference
IconExtensions Class
ExtractIcon Overload
KGySoft.Drawing Namespace
IconExtensionsExtractBitmap(Icon, Int32, Boolean)

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Extracts the icon of specified size and pixel format from an Icon instance. Unless the Icon constructors, this method works as expected.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

```csharp
public static Icon ExtractIcon(
    this Icon icon,
    Size size,
    PixelFormat pixelFormat
)
```

**Parameters**

- `icon`  
  Type: `System.Drawing.Icon`  
  The icon that may contain multiple images.

- `size`  
  Type: `System.Drawing.SizeF`  
  The required icon size to retrieve.

- `PixelFormat`  
  Type: `System.Drawing.Imaging.PixelFormat`  
  The required pixel format to retrieve.

**Return Value**

Type: Icon  
An Icon instance, which contains only a single image, or if no icon found with the specified size and format.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type Icon. When you use instance method syntax to call this
method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference
IconExtensions Class
ExtractIcon Overload
KGySoft.Drawing Namespace
IconExtensionsExtractBitmap(Icon, Size, PixelFormat, Boolean)

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
IconExtensions.ExtractIcons Method

Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExtractIcons(Icon)</td>
<td>Extracts every icon from an Icon instance as separated instances.</td>
</tr>
<tr>
<td>ExtractIcons(Icon, PixelFormat)</td>
<td>Extracts every icon of specified pixel format from an Icon instance as separated instances.</td>
</tr>
<tr>
<td>ExtractIcons(Icon, Size)</td>
<td>Extracts every icon of specified size from an Icon instance as separated instances.</td>
</tr>
</tbody>
</table>

See Also

Reference

IconExtensions Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
IconExtensions.ExtractIcons

Method (Icon)

Extracts every icon from an Icon instance as separated instances.

Namespace: KGySoft.Drawing

Syntax

```csharp
public static Icon[] ExtractIcons(
    this Icon icon
)
```

Parameters

icon
  Type: System.Drawing.Icon
  The icon that may contain multiple images.

Return Value

Type: Icon
An array of Icon instances, which were extracted from the icon.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type Icon. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference

IconExtensions Class
ExtractIcons Overload
KGySoft.Drawing Namespace
System.DrawingIcon
IconExtensionsExtractBitmaps(Icon, Boolean)

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
**IconExtensions**

**ExtractIcons Method** *(Icon, PixelFormat)*

Extracts every icon of specified pixel format from an Icon instance as separated instances.

**Namespace**: KGySoft.Drawing  

**Syntax**

```csharp
public static Icon[] ExtractIcons(
    this Icon icon,
    PixelFormat pixelFormat
)
```

**Parameters**

- **icon**  
  - Type: System.Drawing.Icon  
  - The icon that may contain multiple images.

- **pixelFormat**  
  - Type: System.Drawing.Imaging.PixelFormat  
  - The required pixel format to retrieve

**Return Value**

- Type: Icon  
- An array of Icon instances, which were extracted from the icon.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type Icon. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

**See Also**

Reference
IconExtensions Class
ExtractIcons Overload
KGySoft.Drawing Namespace
System.DrawingIcon
IconExtensionsExtractBitmaps(Icon, PixelFormat, Boolean)

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
IconExtensions ExtractIcons Method (Icon, Size)

Extracts every icon of specified size from an Icon instance as separated instances.

Namespace: KGySoft.Drawing  

Syntax

```csharp
c# public static Icon[] ExtractIcons(
this Icon icon,
Size size
)
```

Parameters

- **icon**
  - Type: System.DrawingIcon
  - The icon that may contain multiple images.

- **size**
  - Type: System.DrawingSize
  - The required icon size to retrieve.

Return Value

Type: Icon
An array of Icon instances, which were extracted from the icon.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type Icon. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference
IconExtensions Class
ExtractIcons Overload
KGySoft.Drawing Namespace
System.DrawingIcon
IconExtensionsExtractBitmaps(Icon, Size, Boolean)

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
**IconExtensions**

**ExtractNearestBitmap**

**Method**

Extracts the nearest image of specified size and pixel format from an `Icon` instance.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

```csharp
public static Bitmap ExtractNearestBitmap(
    this Icon icon,
    Size size,
    PixelFormat pixelFormat,
    bool keepOriginalFormat = false
)
```

### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>icon</code></td>
<td><code>System.Drawing.Icon</code></td>
<td>The icon that may contain multiple images.</td>
</tr>
<tr>
<td><code>size</code></td>
<td><code>System.Drawing.Size</code></td>
<td>The required icon size to retrieve.</td>
</tr>
<tr>
<td><code>pixelFormat</code></td>
<td><code>System.Drawing.Imaging.PixelFormat</code></td>
<td>The required pixel format to retrieve</td>
</tr>
<tr>
<td><code>keepOriginalFormat</code> (Optional)</td>
<td><code>System.Boolean</code></td>
<td>If <code>true</code>, keeps the original image format stored in the <code>icon</code>. Possible transparent pixels of non-32 bpp ARGB formats may be black. If <code>false</code>, always returns 32 bpp images with transparency. This parameter is optional. Default value: <code>false</code>.</td>
</tr>
</tbody>
</table>
**Return Value**

Type: **Bitmap**

An **Bitmap** instance, which was extracted from the *icon*. If no icon was found with the specified size and format the nearest image (*pixelFormat* matches first, then *size*) is returned.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type **Icon**. When you use instance method syntax to call this method, omit the first parameter. For more information, see **Extension Methods (Visual Basic)** or **Extension Methods (C# Programming Guide)**.

**See Also**

**Reference**

**IconExtensions Class**

**KGySoft.Drawing Namespace**

**IconExtensionsExtractNearestIcon(Icon, Size, PixelFormat)**

Find the complete KGy SOFT Libraries documentation at the **KGy SOFT Docs** page.

Copyright © KGy SOFT. All rights reserved.
IconExtensions\ExtractNearestIcon Method

Extracts the nearest icon of specified size and pixel format from an Icon instance. Unless the Icon constructors, this method works as expected.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

## Syntax

```
public static Icon ExtractNearestIcon(
    this Icon icon,
    Size size,
    PixelFormat pixelFormat
)
```

### Parameters

**icon**

Type: System.DrawingIcon  
The icon that may contain multiple images.

**size**

Type: System.Drawing.Size  
The required icon size to retrieve.

**pixelFormat**

Type: System.Drawing.Imaging.PixelFormat  
The required pixel format to retrieve

### Return Value

Type: Icon  
An Icon instance, which contains only a single image. If no icon was found with the specified size and format the nearest icon (pixelFormat matches first, then size) is returned.

### Usage Note

In Visual Basic and C#, you can call this method as an instance method on...
any object of type Icon. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference

IconExtensions Class

KGySoft.Drawing Namespace

IconExtensionsExtractNearestBitmap(Icon, Size, PixelFormat, Boolean)

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
IconExtensions

GetImagesCount Method

Gets the number of images in the icon.

**Namespace:** KGySoft.Drawing

**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public static int GetImagesCount(
    this Icon icon
)
```

**Parameters**

*icon*

- **Type:** System.Drawing.Icon
  - The icon to check.

**Return Value**

- **Type:** Int32
  - The number of images in the icon.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type Icon. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

**See Also**

**Reference**

- IconExtensions Class
- KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Saves the *icon* into the specified *stream*. Unlike *Icon.Save*, this method can save every icon with high quality, even *SystemIcons* members, and icons created by the *Icon.FromHandle* method.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

```csharp
public static void SaveHighQuality(  
    this Icon icon,  
    Stream stream
)
```

#### Parameters

**icon**

Type: *System.DrawingIcon*

The icon to save

**stream**

Type: *System.IOStream*

A stream into which the icon has to be saved.

#### Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type *Icon*. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

### See Also

- **Reference**
  - IconExtensions Class
  - KGySoft.Drawing Namespace
Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
IconExtensions.ToAlphaBitmap

Converts the specified `icon` to a `Bitmap`. While `Icon.ToBitmap` may return a wrong result when icon contains semi-transparent pixels, this method returns an image, in which alpha channel is always correctly applied for the image.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll)  
Version: 4.6.1

### Syntax

```csharp
public static Bitmap ToAlphaBitmap(
    this Icon icon
)
```

**Parameters**

- **icon**
  - Type: `System.Drawing.Icon`
  - The icon optionally with transparency.

**Return Value**

- **Type:** `Bitmap`
  - A `Bitmap` that represents the converted `Icon`.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type `Icon`. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

### Remarks

`ExtractBitmap` and `ToAlphaBitmap(Icon)` methods may return a different result even if the `icon` contains a single image only. The `ExtractBitmap` overloads works from the saved icon stream in the first place, which is slower than this method.
If the *icon* contains multiple images consider to use either the `ExtractBitmap` overloads to specify the exact image to return, or the `ToMultiResBitmap` method, which returns every images in a single combined `Bitmap`.

### See Also

**Reference**

IconExtensions Class

KGySoft.Drawing Namespace

IconExtensionsExtractBitmap(Icon, Boolean)

IconExtensionsExtractBitmap(Icon, Size, PixelFormat, Boolean)

IconExtensionsExtractBitmap(Icon, Int32, Boolean)

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](https://www.kgysoft.com/) page.

Copyright © KGy SOFT. All rights reserved.
IconExtensions::ToCursorHandle Method

Converts the provided icon to a CursorHandle, which can be passed to the System.Windows.Forms.Cursors constructor to create a new cursor.

Namespace: KGySoft.Drawing

Syntax

```csharp
public static CursorHandle ToCursorHandle(
    this Icon icon,
    Point cursorHotspot = null
)
```

Parameters

- **icon**
  - Type: System.Drawing.Icon
  - The Icon, which should be converted to a cursor.

- **cursorHotspot** (Optional)
  - Type: System.Drawing.Point
  - The hotspot coordinates of the cursor. This parameter is optional.
  - Default value: 0; 0 (top-left corner)

Return Value

- Type: CursorHandle
- A CursorHandle instance that can be used to create a System.Windows.Forms.Cursor instance.

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type Icon. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).
See Also
Reference
IconExtensions Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
IconExtensionsToMultiResBitmap Method

Converts the icon to a Bitmap instance, which contains every image of the icon. When the returned Bitmap is used to create another Bitmap or is drawn into a Graphics, the best-fitting image is automatically applied.

Namespace: KGySoft.Drawing

Syntax

```csharp
public static Bitmap ToMultiResBitmap(
    this Icon icon
)
```

Parameters

icon
Type: System.DrawingIcon
The icon to convert to a multi-resolution Bitmap.

Return Value
Type: Bitmap
A Bitmap instance, which contains every image of the icon.

Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type Icon. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

See Also

Reference
IconExtensions Class
KGySoft.Drawing Namespace
Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Provides some icon-related methods as well as properties returning general icons in multi resolution. Unlike SystemIcons, these icons should be disposed when not used any more.

### Inheritance Hierarchy

**SystemObject**  \> **KGySoft.DrawingIcons**

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll)  
**Version:** 4.6.1

### Syntax

#### C#  
```csharp
public static class Icons
```

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

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>CombineTransform(Bitmap)</code></td>
<td>Combines the provided <em>images</em> into a multi-resolution <em>Icon</em> instance.</td>
</tr>
<tr>
<td><code>CombineTransform(Icon)</code></td>
<td>Combines the provided <em>icons</em> into a multi-resolution <em>Icon</em> instance.</td>
</tr>
<tr>
<td><code>CombineTransform(Bitmap, Color)</code></td>
<td>Combines the provided <em>images</em> into a multi-resolution <em>Icon</em> instance.</td>
</tr>
<tr>
<td><code>FromExtension</code></td>
<td>Gets the system-associated icon of a file extension.</td>
</tr>
<tr>
<td><code>FromFile(String)</code></td>
<td>Extracts dual-resolution icons from a file and returns them as separated <em>Icon</em> instances.</td>
</tr>
<tr>
<td><code>FromFile(String, SystemIconSize)</code></td>
<td>Extracts icons of the specified <em>size</em> from a file and returns them as separated <em>Icon</em> instances.</td>
</tr>
<tr>
<td><code>GetStockIcon</code></td>
<td>Tries to get a system stock icon. When there is no icon defined for provided <em>id</em>, or Windows version is below Vista, this method returns . In Windows XP use the predefined property members to retrieve system icon instances.</td>
</tr>
</tbody>
</table>
## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>Gets the Application icon displaying a window (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)</td>
</tr>
<tr>
<td>Error</td>
<td>Gets the Error icon displaying a white &quot;X&quot; in a red circle (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)</td>
</tr>
<tr>
<td>Information</td>
<td>Gets the Information icon displaying a white &quot;i&quot; in a blue circle (Sizes: 256x256, 48x48, 32x32, 16x16)</td>
</tr>
<tr>
<td>Question</td>
<td>Gets the Question icon displaying a white &quot;?&quot; in a blue circle (Sizes: 256x256, 64x64, 48x48, 32x32, 16x16)</td>
</tr>
<tr>
<td>SecurityError</td>
<td>Gets the Security Error icon displaying a red shield with a white &quot;X&quot; (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)</td>
</tr>
<tr>
<td>SecurityQuestion</td>
<td>Gets the Security Question icon displaying a blue shield with a white &quot;?&quot; (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)</td>
</tr>
<tr>
<td>SecurityShield</td>
<td>Gets the Security Shield icon displaying a blue-yellow shield (Sizes: 256x256, 128x128, 48x48, 32x32, 24x24, 16x16, 8x8)</td>
</tr>
<tr>
<td>SecuritySuccess</td>
<td>Gets the Security Success icon displaying a green shield with a white check (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)</td>
</tr>
<tr>
<td>SecurityWarning</td>
<td>Gets the Security Warning icon displaying a yellow shield with a black &quot;!&quot; (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)</td>
</tr>
<tr>
<td>Shield</td>
<td>Gets the Windows Shield icon displaying a red-green-blue-yellow shield (Sizes: 256x256, 128x128, 48x48, 32x32, 24x24, 16x16, 8x8)</td>
</tr>
<tr>
<td>SystemApplication</td>
<td>Gets an Icon instance that contains a large and a small Application icon as it is stored in the current operating system. In Windows Vista and above sizes are depending on current DPI settings, in Windows XP they have always 32x32 and 16x16 sizes.</td>
</tr>
<tr>
<td>SystemError</td>
<td>Gets an Icon instance that contains a large and a small Error icon as it is stored in the current operating system. In Windows Vista and above sizes are depending on current DPI settings, in Windows XP they have always 32x32 and 16x16 sizes.</td>
</tr>
<tr>
<td>SystemInformation</td>
<td>Gets an Icon instance that contains a large and a small Information icons.</td>
</tr>
</tbody>
</table>
icon as it is stored in the current operating system. In Windows Vista and above sizes are depending on current DPI settings, in Windows XP they have always 32x32 and 16x16 sizes.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Icon" /> SystemQuestion</td>
<td>Gets an Icon instance that contains a large and a small Question icon as it is stored in the current operating system. In Windows Vista and above sizes are depending on current DPI settings, in Windows XP they have always 32x32 and 16x16 sizes.</td>
</tr>
<tr>
<td><img src="image" alt="Icon" /> SystemShield</td>
<td>Gets an Icon instance that contains the Shield icon as it is stored in the current operating system. In Windows Vista and above icon contains two sizes, which are depending on current DPI settings, in Windows XP the icon contains multiple resolution and color depths.</td>
</tr>
<tr>
<td><img src="image" alt="Icon" /> SystemWarning</td>
<td>Gets an Icon instance that contains a large and a small Warning icon as it is stored in the current operating system. In Windows Vista and above sizes are depending on current DPI settings, in Windows XP they have always 32x32 and 16x16 sizes.</td>
</tr>
<tr>
<td><img src="image" alt="Icon" /> Warning</td>
<td>Gets the Warning icon displaying a black &quot;!&quot; in a yellow triangle (Sizes: 256x256, 48x48, 32x32, 16x16)</td>
</tr>
</tbody>
</table>

Top

See Also

Reference

KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
The **Icons** type exposes the following members.

## Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Combine(Bitmap)</td>
<td>Combines the provided <em>images</em> into a multi-resolution <em>Icon</em> instance.</td>
</tr>
<tr>
<td>Combine(Icon)</td>
<td>Combines the provided <em>icons</em> into a multi-resolution <em>Icon</em> instance.</td>
</tr>
<tr>
<td>Combine(Bitmap, Color)</td>
<td>Combines the provided <em>images</em> into a multi-resolution <em>Icon</em> instance.</td>
</tr>
<tr>
<td>FromExtension</td>
<td>Gets the system-associated icon of a file extension.</td>
</tr>
<tr>
<td>FromFile(String)</td>
<td>Extracts dual-resolution icons from a file and returns them as separated <em>Icon</em> instances.</td>
</tr>
<tr>
<td>FromFile(String, SystemIconSize)</td>
<td>Extracts icons of the specified <em>size</em> from a file and returns them as separated <em>Icon</em> instances.</td>
</tr>
<tr>
<td>GetStockIcon</td>
<td>Tries to get a system stock icon. When there is no icon defined for provided <em>id</em>, or Windows version is below Vista, this method returns . In Windows XP use the predefined property members to retrieve system icons.</td>
</tr>
</tbody>
</table>

**See Also**

**Reference**

**Icons Class**

**KGySoft.Drawing Namespace**

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](#) page.

Copyright © KGy SOFT. All rights reserved.
## IconsCombine Method

### Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>★ Combine(Bitmap)</td>
<td>Combines the provided images into a multi-resolution Icon instance.</td>
</tr>
<tr>
<td>★ Combine(Icon)</td>
<td>Combines the provided icons into a multi-resolution Icon instance.</td>
</tr>
<tr>
<td>★ Combine(Bitmap, Color)</td>
<td>Combines the provided images into a multi-resolution Icon instance.</td>
</tr>
</tbody>
</table>

### See Also

- **Reference**
  - Icons Class
  - KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](page).  
Copyright © KGy SOFT. All rights reserved.
Combines the provided images into a multi-resolution Icon instance.

**Namespace**: KGySoft.Drawing  

**Syntax**

```csharp
public static Icon Combine(
        params Bitmap[] images
    )
```

**Parameters**

*images*
- **Type**: System.Drawing.Bitmap
- The images to be added to the result icon. Images can be non-squared ones. Transparency is determined automatically by image format.

**Return Value**
- **Type**: Icon
- An Icon instance that contains every image of the source images.

**See Also**

- **Reference**
  - Icons Class
  - Combine Overload
  - KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
Combine Method (Icon)

Combines the provided icons into a multi-resolution Icon instance.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll)  
**Version:** 4.6.1

### Syntax

**C#**

```csharp
public static Icon Combine(
    params Icon[] icons
)
```

**Parameters**

*icons*

Type: System.Drawing.Icon  
The icons to be combined.

**Return Value**

Type: Icon  
An Icon instance that contains every image of the source icons.

### Remarks

The elements of icons may contain multiple icons.

### See Also

**Reference**

Icons Class  
Combine Overload  
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.  
Copyright © KGy SOFT. All rights reserved.
**Icons**

**Combine Method (Bitmap, Color)**

Combines the provided `images` into a multi-resolution `Icon` instance.

**Namespace:** KGYSoft.Drawing  
**Assembly:** KGYSoft.Drawing (in KGYSoft.Drawing.dll)  
**Version:** 4.6.1

## Syntax

**C#**

```csharp
public static Icon Combine(
    Bitmap[] images,
    Color[] transparentColors
)
```

### Parameters

- **images**
  - **Type:** System.Drawing.Bitmap
  - The images to be added to the icon. Images can be non-squares ones.

- **transparentColors**
  - **Type:** System.Drawing.Color
  - An array of transparent colors of the images. The array must have as many elements as `images`.

### Return Value

- **Type:** Icon  
  - An `Icon` instance that contains every image of the source `images`.

## See Also

**Reference**

- Icons Class
- Combine Overload
- KGYSoft.Drawing Namespace

Find the complete KGY SOFT Libraries documentation at the KGY SOFT
Docs page.
Copyright © K Gy SOFT. All rights reserved.
**IconsFromExtension Method**

Gets the system-associated icon of a file extension.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public static Icon FromExtension(
    string extension,
    SystemIconSize size
)
```

**Parameters**

- **extension**  
  Type: SystemString  
  A file name (can be a non-existing one) or an extension for which the associated icon is about to be retrieved.

- **size**  
  Type: KGySoft.DrawingSystemIconSize  
  The size of the icon to be retrieved.

**Return Value**

Type: Icon  
The icon of the specified extension.

**See Also**

**Reference**

- Icons Class  
- KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FromFile(String)</td>
<td>Extracts dual-resolution icons from a file and returns them as separated Icon instances.</td>
</tr>
<tr>
<td>FromFile(String, SystemIconSize)</td>
<td>Extracts icons of the specified size from a file and returns them as separated Icon instances.</td>
</tr>
</tbody>
</table>

See Also

Reference

Icons Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
IconsFromFile Method (String)

Extracts dual-resolution icons from a file and returns them as separated Icon instances.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

```csharp
public static Icon[] FromFile(
    string fileName
)
```

### Parameters

`fileName`

Type: `System.String`

The name of the file. Can be an executable file, a .dll or icon file.

### Return Value

Type: `Icon`

The icons of the specified file, or an empty array if the file does not exist or does not contain any icons.

### Remarks

If `fileName` refers to an icon file use the `Icon(String)` constructor instead.

The images of an `Icon` can be extracted by the `IconExtensions.ExtractBitmaps` methods.

### See Also

**Reference**

Icons Class
FromFile Overload
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Extracts icons of the specified size from a file and returns them as separated Icon instances.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

```csharp
public static Icon[] FromFile(
    string fileName,
    SystemIconSize size
)
```

### Parameters

- **fileName**
  - Type: `string`  
  - The name of the file. Can be an executable file, a .dll or icon file.

- **size**
  - Type: `KGySoft.DrawingSystemIconSize`  
  - The size of the icons to be extracted.

### Return Value

- Type: `Icon`  
  - The icons of the specified file, or an empty array if the file does not exist or does not contain any icons.

### Remarks

If `fileName` refers to an icon file use the `Icon(String)` constructor instead.

The images of an `Icon` can be extracted by the `IconExtensions.ExtractBitmaps` methods.
See Also

Reference
Icons Class
FromFile Overload
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
GetStockIcon Method

Tries to get a system stock icon. When there is no icon defined for provided id, or Windows version is below Vista, this method returns . In Windows XP use the predefined property members to retrieve system icons.

Namespace: KGySoft.Drawing

Syntax

```csharp
public static Icon GetStockIcon(
    StockIcon id
)
```

Parameters

id
Type: KGySoft.DrawingStockIcon
Id of the icon to retrieve. For future compatibility reasons non-defined StockIcon values are also allowed.

Return Value
Type: Icon
An Icon instance containing a small and large icon when an icon belongs to id, or , when no icon found or Windows version is below Vista.

See Also

Reference
Icons Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
The **Icons** type exposes the following members.

## Properties

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>Gets the Application icon displaying a window (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)</td>
</tr>
<tr>
<td>Error</td>
<td>Gets the Error icon displaying a white &quot;X&quot; in a red circle (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)</td>
</tr>
<tr>
<td>Information</td>
<td>Gets the Information icon displaying a white &quot;i&quot; in a blue circle (Sizes: 256x256, 48x48, 32x32, 16x16)</td>
</tr>
<tr>
<td>Question</td>
<td>Gets the Question icon displaying a white &quot;?&quot; in a blue circle (Sizes: 256x256, 64x64, 48x48, 32x32, 16x16)</td>
</tr>
<tr>
<td>SecurityError</td>
<td>Gets the Security Error icon displaying a red shield with a white &quot;X&quot; (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)</td>
</tr>
<tr>
<td>SecurityQuestion</td>
<td>Gets the Security Question icon displaying a blue shield with a white &quot;?&quot;, (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)</td>
</tr>
<tr>
<td>SecurityShield</td>
<td>Gets the Security Shield icon displaying a blue-yellow shield (Sizes: 256x256, 128x128, 48x48, 32x32, 24x24, 16x16, 8x8)</td>
</tr>
<tr>
<td>SecuritySuccess</td>
<td>Gets the Security Success icon displaying a green shield with a white check (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)</td>
</tr>
<tr>
<td>SecurityWarning</td>
<td>Gets the Security Warning icon displaying a yellow shield with a black &quot;!&quot;, (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)</td>
</tr>
<tr>
<td>Shield</td>
<td>Gets the Windows Shield icon displaying a red-green-blue-yellow shield (Sizes: 256x256, 128x128, 48x48, 32x32, 24x24, 16x16, 8x8)</td>
</tr>
<tr>
<td>SystemApplication</td>
<td>Gets an <strong>Icon</strong> instance that contains a large and a small Application icon as it is stored in the current operating system. In Windows Vista and above sizes are depending on current DPI settings, in Windows XP they have always 32x32 and 16x16 sizes.</td>
</tr>
<tr>
<td>Icon</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SystemError</td>
<td>Gets an Icon instance that contains a large and a small Error icon as it is stored in the current operating system. In Windows Vista and above sizes are depending on current DPI settings, in Windows XP they have always 32x32 and 16x16 sizes.</td>
</tr>
<tr>
<td>SystemInformation</td>
<td>Gets an Icon instance that contains a large and a small Information icon as it is stored in the current operating system. In Windows Vista and above sizes are depending on current DPI settings, in Windows XP they have always 32x32 and 16x16 sizes.</td>
</tr>
<tr>
<td>SystemQuestion</td>
<td>Gets an Icon instance that contains a large and a small Question icon as it is stored in the current operating system. In Windows Vista and above sizes are depending on current DPI settings, in Windows XP they have always 32x32 and 16x16 sizes.</td>
</tr>
<tr>
<td>SystemShield</td>
<td>Gets an Icon instance that contains the Shield icon as it is stored in the current operating system. In Windows Vista and above icon contains two sizes, which are depending on current DPI settings, in Windows XP the icon contains multiple resolution and color depths.</td>
</tr>
<tr>
<td>SystemWarning</td>
<td>Gets an Icon instance that contains a large and a small Warning icon as it is stored in the current operating system. In Windows Vista and above sizes are depending on current DPI settings, in Windows XP they have always 32x32 and 16x16 sizes.</td>
</tr>
<tr>
<td>Warning</td>
<td>Gets the Warning icon displaying a black &quot;!&quot; in a yellow triangle (Sizes: 256x256, 48x48, 32x32, 16x16)</td>
</tr>
</tbody>
</table>

**See Also**

Reference

Icons Class

KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
**IconsApplication Property**

Gets the Application icon displaying a window (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)

**Namespace:** KGySoft.Drawing
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public static Icon Application { get; }
```

**Remarks**

The icon contains the following images:

**See Also**

Reference
Icons Class
KGySoft.Drawing Namespace
Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Icons Error Property

Gets the Error icon displaying a white "X" in a red circle (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)

Namespace: KGySoft.Drawing

Syntax

```csharp
public static Icon Error { get; }
```

Remarks

The icon contains the following images:

See Also

Reference
Icons Class
KGySoft.Drawing Namespace
Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
**IconsInformation Property**  
KGY SOFT Drawing Libraries Help

Gets the Information icon displaying a white "i" a blue circle (Sizes: 256x256, 48x48, 32x32, 16x16)

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public static Icon Information { get; }
```

**Property Value**

Type: Icon

**Remarks**

The icon contains the following images:

![Information Icon](image)

**See Also**

Reference  
Icons Class  
KGySoft.Drawing Namespace
Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Imports KGySoft.Drawing

Namespace: KGySoft.Drawing

Syntax

```csharp
public static Icon Question { get; }
```

Remark

The icon contains the following images:
Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Syntax

```csharp
public static Icon SecurityError { get; }
```

Remarks

The icon contains the following images:

See Also

Reference

Icons Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Gets the Security Question icon displaying a blue shield with a white "?" (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)

Namespace: KGySoft.Drawing

Syntax

```csharp
public static Icon SecurityQuestion { get; }
```

Property Value
Type: Icon

Remarks

The icon contains the following images:
Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
** KGy SOFT **

**Icons**

**SecurityShield Property**

Gets the Security Shield icon displaying a blue-yellow shield (Sizes: 256x256, 128x128, 48x48, 32x32, 24x24, 16x16, 8x8)

**Namespace:** KGySoft.Drawing

**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```
public static Icon SecurityShield { get; }
```

**Property Value**

*Type: Icon*

**Remarks**

The icon contains the following images:

![Security Shield Icon](image)

**See Also**

Reference

Icons Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Gets the Security Success icon displaying a green shield with a white check (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)

Namespace: KGySoft.Drawing

Syntax

```csharp
public static Icon SecuritySuccess { get; }
```

Remarks

The icon contains the following images:

See Also

Reference

Icons Class
KGYSoft.Drawing Namespace

Find the complete KGY SOFT Libraries documentation at the KGY SOFT Docs page.
Copyright © KGY SOFT. All rights reserved.
**KGy SOFT .net**

**Icons**

**Security Warning Property**

- **Get**s the Security Warning icon displaying a yellow shield with a black "!" (Sizes: 256x256, 48x48, 32x32, 24x24, 16x16)

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public static Icon SecurityWarning { get; }
```

- **Property Value**
  - **Type:** Icon

**Remarks**

The icon contains the following images:

![Security Warning Icon](image)

**See Also**

- Reference
- Icons Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
**Icons Shield Property**

Gets the Windows Shield icon displaying a red-green-blue-yellow shield (Sizes: 256x256, 128x128, 48x48, 32x32, 24x24, 16x16, 8x8)

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

<table>
<thead>
<tr>
<th>C#</th>
<th>VB</th>
<th>C++</th>
<th>F#</th>
</tr>
</thead>
</table>

```csharp
public static Icon Shield { get; }
```

**Property Value**  
**Type:** Icon

### Remarks

The icon contains the following images:

![Shield Icon](image)

### See Also

- Reference
- Icons Class
- KGySoft.Drawing Namespace
**Icons SystemApplication Property**

- [x] Gets an [Icon](#) instance that contains a large and a small Application icon as it is stored in the current operating system. In Windows Vista and above sizes are depending on current DPI settings, in Windows XP they have always 32x32 and 16x16 sizes.

**Namespace**: KGySoft.Drawing  
**Assembly**: KGySoft.Drawing (in KGySoft.Drawing.dll)  
**Version**: 4.6.1

### Syntax

```csharp
public static Icon SystemApplication { get; }
```

**Property Value**  
**Type**: Icon

### Remarks

In Windows Vista and Windows 7, with default DPI settings the icon contains the following images:

![Icon](#)

In Windows XP the icon contains the following images:

![Icon](#)

### See Also

- Reference  
  - Icons Class  
  - KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](#) page.
Icons System Error Property

⚠️ Gets an Icon instance that contains a large and a small Error icon as it is stored in the current operating system. In Windows Vista and above sizes are depending on current DPI settings, in Windows XP they have always 32x32 and 16x16 sizes.

Namespace: KGySoft.Drawing

Syntax

```c# public static Icon SystemError { get; }
```

Remarks

In Windows Vista and Windows 7, with default DPI settings the icon contains the following images:

![Error Icon](image)

In Windows XP the icon contains the following images:

![Error Icon](image)

See Also

Reference
- Icons Class
- KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
**Property**

Gets an `Icon` instance that contains a large and a small Information icon as it is stored in the current operating system. In Windows Vista and above sizes are depending on current DPI settings, in Windows XP they have always 32x32 and 16x16 sizes.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public static Icon SystemInformation { get; }
```

**Remarks**

In Windows Vista and Windows 7, with default DPI settings the icon contains the following images:

In Windows XP the icon contains the following images:

**See Also**

- **Reference**  
  - Icons Class  
  - KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
**Icons**

**SystemQuestion Property**

GETS an `Icon` instance that contains a large and a small Question icon as it is stored in the current operating system. In Windows Vista and above, sizes are depending on current DPI settings, in Windows XP they have always 32x32 and 16x16 sizes.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll)  
**Version:** 4.6.1

**Syntax**

```
public static Icon SystemQuestion { get; }
```

**Remarks**

In Windows Vista and Windows 7, with default DPI settings the icon contains the following images:

!?

In Windows XP the icon contains the following images:

!?

**See Also**

- **Reference**
  - Icons Class
  - KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
**KGy SOFT**

**Icons**

**SystemShield**

**Property**

![Image](image)

Gets an `Icon` instance that contains the Shield icon as it is stored in the current operating system. In Windows Vista and above icon contains two sizes, which are depending on current DPI settings, in Windows XP the icon contains multiple resolution and color depths.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll)  
**Version:** 4.6.1

**Syntax**

```csharp
public static Icon SystemShield { get; }
```

**Remarks**

In Windows 7, with default DPI settings the icon contains the following images:

![Images](image)

In Windows Vista, with default DPI settings the icon contains the following images:

![Images](image)

In Windows XP the icon contains different color depth version of the following images:

![Images](image)

**See Also**
Reference

Icons Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
Icons System Warning Property

⚠ Gets an Icon instance that contains a large and a small Warning icon as it is stored in the current operating system. In Windows Vista and above sizes are depending on current DPI settings, in Windows XP they have always 32x32 and 16x16 sizes.

Namespace: KGySoft.Drawing

Syntax

```csharp
public static Icon SystemWarning { get; }
```

Remarks

In Windows Vista and Windows 7, with default DPI settings the icon contains the following images:

⚠⚠

In Windows XP the icon contains the following images:

⚠⚠

See Also

Reference
Icons Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
**IconsWarning Property**

⚠ Gets the Warning icon displaying a black "!" in a yellow triangle (Sizes: 256x256, 48x48, 32x32, 16x16)

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public static Icon Warning { get; }
```

**Remarks**

The icon contains the following images:

![Warning Icon](image)

**See Also**

- Reference
- Icons Class
- KGySoft.Drawing Namespace
Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
ImageExtensions Class

Contains extension methods for the Image type.

Inheritance Hierarchy

System\Object  KGySoft.Drawing\ImageExtensions

Namespace: KGySoft.Drawing

Syntax

```csharp
public static class ImageExtensions
```

The ImageExtensions type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConvertPixelFormat</td>
<td>Converts the image to another one with the desiredPixelFormat. See the Remarks section for details.</td>
</tr>
<tr>
<td>EqualsByContent</td>
<td>Compares an image to another one by content and returns whether they are equal. Images of different size or pixel format are considered as difference.</td>
</tr>
<tr>
<td>GetBitsPerPixel</td>
<td>Gets the bits per pixel (bpp) value of the image.</td>
</tr>
<tr>
<td>SaveAsGif(Image, Stream, Boolean)</td>
<td>Saves the specified image as a GIF image. See the Remarks section for the differences compared to the Image.Save(Stream,ImageFormat) method.</td>
</tr>
<tr>
<td>SaveAsGif(Image, Stream, Color)</td>
<td>Saves the specified image as a GIF image. See the Remarks section for the differences compared to the Image.Save(Stream,ImageFormat) method.</td>
</tr>
<tr>
<td>SaveAsMultipageTiff</td>
<td>Saves the provided images as a multi-page TIFF into the specified Stream. When Image instances in images contain already multiple pages, only the current page is taken.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>ToGrayscale</td>
<td>Converts an image to a grayscale one.</td>
</tr>
<tr>
<td>ToIcon</td>
<td>Creates an Icon from an Image.</td>
</tr>
</tbody>
</table>

**See Also**

**Reference**

KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.

Copyright © KGy SOFT. All rights reserved.
The ImageExtensions type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConvertPixelFormat</td>
<td>Converts the image to another one with the desiredPixelFormat. See the Remarks section for details.</td>
</tr>
<tr>
<td>EqualsByContent</td>
<td>Compares an image to another one by content and returns whether they are equal. Images of different size or pixel format are considered as difference.</td>
</tr>
<tr>
<td>GetBitsPerPixel</td>
<td>Gets the bits per pixel (bpp) value of the image.</td>
</tr>
<tr>
<td>SaveAsGif(Image, Stream, Boolean)</td>
<td>Saves the specified image as a GIF image. See the Remarks section for the differences compared to the Image.Save(Stream,ImageFormat) method.</td>
</tr>
<tr>
<td>SaveAsGif(Image, Stream, Color)</td>
<td>Saves the specified image as a GIF image. See the Remarks section for the differences compared to the Image.Save(Stream,ImageFormat) method.</td>
</tr>
<tr>
<td>SaveAsMultipageTiff</td>
<td>Saves the provided images as a multi-page TIFF into the specified Stream. When Image instances in images contain already multiple pages, only the current page is taken.</td>
</tr>
<tr>
<td>ToGrayscale</td>
<td>Converts an image to a grayscale one.</td>
</tr>
<tr>
<td>ToIcon</td>
<td>Creates an Icon from an Image.</td>
</tr>
</tbody>
</table>

See Also

Reference

ImageExtensions Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
ImageExtensions\ConvertPixelFormat

Method

Converts the image to another one with the desired \texttt{PixelFormat}.
See the \textbf{Remarks} section for details.

\textbf{Namespace:} KGySoft.Drawing

\textbf{Syntax}

\begin{verbatim}
public static Image ConvertPixelFormat(
    this Image image,
    PixelFormat newPixelFormat,
    Color[] palette
)
\end{verbatim}

\textbf{Parameters}

\textit{image}
Type: \texttt{System.DrawingImage}
The original image to convert.

\textit{newPixelFormat}
Type: \texttt{System.Drawing.ImagingPixelFormat}
The desired new pixel format. If the requested format is an indexed one, built-in strategies are used for the generated palette.

\textit{palette}
Type: \texttt{System.DrawingColor}
The required palette for the result image. If, the palette will be taken from source or will be generated on demand. If palette contains transparent color, it might be considered. If it contains too few elements black entries will be added.

\textbf{Return Value}
Type: \texttt{Image}
A new \texttt{Image} instance with the desired pixel format.
Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type `Image`. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

Remarks

If `newPixelFormat` is `Format8bppIndexed`, `image` has no palette and `palette` is , a standard palette will be used. Transparency will be preserved.

If `newPixelFormat` is `Format4bppIndexed`, `image` has no palette and `palette` is , the standard 16 color palette will be used.

If `newPixelFormat` is `Format1bppIndexed`, `image` has no palette and `palette` is , black and white colors will be used.

If the target pixel format is indexed, `palette` contains the transparent color (`Color.Transparent`), and the source has transparency, then the result will have transparency for fully transparent pixels.

See Also

Reference

ImageExtensions Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
ImageExtensionsEqualsByContent Method

Compares an image to another one by content and returns whether they are equal. Images of different size or pixel format are considered as difference.

Namespace: KGySoft.Drawing

Syntax

```csharp
public static bool EqualsByContent(
    this Image image1,
    Image image2
)
```

Parameters

- **image1**
  - Type: System.Drawing.Image
  - First image instance.

- **image2**
  - Type: System.Drawing.Image
  - Second image instance.

Return Value

Type: Boolean
if both images have the same content; otherwise, .

Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type Image. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

Remarks

If an image is not a Bitmap instance, a temporary Bitmap is created for the
check.

<table>
<thead>
<tr>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>This method compares images by raw content. If the images have padding in each stride (content row), padding content is considered as well.</td>
</tr>
</tbody>
</table>

**See Also**

**Reference**

- **ImageExtensions Class**
- **KGySoft.Drawing Namespace**

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs page](https://kgysoft.com/docs).

Copyright © KGy SOFT. All rights reserved.
**ImageExtensions**

**GetBitsPerPixel Method**

Gets the bits per pixel (bpp) value of the image.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public static int GetBitsPerPixel(
    this Image image
)
```

**Parameters**

- `image`:  
  Type: System.Drawing.Image  
  The image to obtain the bits-per-pixel value from.

**Return Value**

- Type: Int32  
  The bits per pixel (bpp) value of the image.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type Image. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

**See Also**

- Reference: ImageExtensions Class  
  KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
## ImageExtensions\_SaveAsGif Method

### Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image.png" alt="Image" /> _SaveAsGif(Image, Stream, Boolean)</td>
<td>Saves the specified <em>image</em> as a GIF image. See the <strong>Remarks</strong> section for the differences compared to the <em>Image.Save(Stream, ImageFormat)</em> method.</td>
</tr>
<tr>
<td><img src="image.png" alt="Image" /> _SaveAsGif(Image, Stream, Color)</td>
<td>Saves the specified <em>image</em> as a GIF image. See the <strong>Remarks</strong> section for the differences compared to the <em>Image.Save(Stream, ImageFormat)</em> method.</td>
</tr>
</tbody>
</table>

### Top

### See Also

**Reference**

- ImageExtensions Class
- KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](http://www.kgysoft.com/docs) page.

Copyright © KGy SOFT. All rights reserved.
ImageExtensions

SaveAsGif Method (Image, Stream, Boolean)

Saves the specified image as a GIF image.
See the Remarks section for the differences compared to the Image.Save(Stream,ImageFormat) method.

Namespace: KGySoft.Drawing

Syntax

```csharp
public static void SaveAsGif(
    this Image image,
    Stream stream,
    bool allowDithering = false
)
```

Parameters

image
Type: System.DrawingImage
The image to save. If image contains multiple images other than animated GIF frames, then only the current image will be saved.

stream
Type: System.IOStream
The stream to save the image into.

allowDithering (Optional)
Type: System.Boolean
to allow dithering high color images using a fix palette; otherwise, . This parameter is optional.
Default value: .

Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type Image. When you use instance method syntax to call this
method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

**Remarks**

When an image is saved by the `Image.Save(Stream, ImageFormat)` method using the GIF image format, then the original palette of an indexed source image and transparency can be lost in many cases. Unless the source image is already a 8 bpp one, the built-in encoder will use a fixed palette and dithers the image, while transparency will be lost.

This method preserves transparency of fully transparent pixels even if `allowDithering` is.

**See Also**

Reference

ImageExtensions Class
SaveAsGif Overload
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
ImageExtensions

SaveAsGif Method (Image, Stream, Color)

Saves the specified *image* as a GIF image. See the **Remarks** section for the differences compared to the *Image.Save(Stream,ImageFormat)* method.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

**C#**

```csharp
public static void SaveAsGif(
    this Image image,
    Stream stream,
    Color[] palette
)
```

**Parameters**

- **image**
  - Type: *System.DrawingImage*
  - The image to save. If image contains multiple images or frames, then the current image will be saved. Animated GIF can be saved only if *palette* is.

- **stream**
  - Type: *System.IOStream*
  - The stream to save the image into.

- **palette**
  - Type: *System.DrawingColor*
  - The desired custom palette to use. If , and a palette cannot be taken from the source image, then a default palette will be used.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type *Image*. When you use instance method syntax to call this method, omit the first parameter. For more information, see **Extension**
When an image is saved by the `Image.Save(Stream,ImageFormat)` method using the GIF image format, then the original palette of an indexed source image and transparency can be lost in many cases. Unless the source image is already a 8 bpp one, the built-in encoder will use a fixed palette and dithers the image, while transparency will be lost.

This method preserves transparency of fully transparent pixels unless `palette` is specified and does not contain the transparent color.

**See Also**

Reference

ImageExtensions Class

SaveAsGif Overload

KGYSoft.Drawing Namespace

Find the complete KGY SOFT Libraries documentation at the KGY SOFT Docs page.

Copyright © KGY SOFT. All rights reserved.
ImageExtensions

SaveAsMultipageTiff

Method

Saves the provided images as a multi-page TIFF into the specified Stream. When Image instances in images contain already multiple pages, only the current page is taken.

Namespace: KGySoft.Drawing

Syntax

```csharp
public static void SaveAsMultipageTiff(
    this IEnumerable<Image> images,
    Stream stream
)
```

Parameters

images
  Type: System.Collections.Generic.IEnumerable<Image>
  The images to save into the TIFF data stream.

stream
  Type: System.IOStream
  The stream into the TIFF data is to be saved.

Usage Note
In Visual Basic and C#, you can call this method as an instance method on any object of type IEnumerable<Image>. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

Remarks
When *images* contain multi-page instances, this method takes only the current page. You can extract images by `ExtractBitmaps` extension method.

Compression mode and bit depth is chosen for each page based on pixel format.

**See Also**

**Reference**

ImageExtensions Class  
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](#) page.  
Copyright © KGy SOFT. All rights reserved.
ImageExtensionsToGrayscale Method

Converts an image to a grayscale one.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public static Image ToGrayscale(
    this Image image
)
```

**Parameters**

*image*
- Type: System.DrawingImage  
The image to convert to grayscale.

**Return Value**
- Type: Image  
The grayscale version of the original *image*.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type Image. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

**See Also**

Reference
- ImageExtensions Class  
- KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
ImageExtensions.ToIcon Method

Creates an Icon from an Image.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll)  
**Version:** 4.6.1

**Syntax**

```csharp
public static Icon ToIcon(
    this Image image,
    int size,
    bool keepAspectRatio
)
```

**Parameters**

- **image**
  - Type: System.Drawing.Image
  - The image to be converted to an icon.

- **size**
  - Type: System.Int32
  - The required size of the icon.

- **keepAspectRatio**
  - Type: System.Boolean
  - When source image is not square sized, determines whether the image should keep aspect ratio.

**Return Value**

Type: Icon  
An Icon instance created from the image.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type Image. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension.
Methods (Visual Basic) or Extension Methods (C# Programming Guide).

Remarks

The result icon will be always square sized. To create a non-squared icon, use the Icons.Combine method instead.

See Also

Reference

ImageExtensions Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
MetafileExtensions Class

Contains extension methods for the Metafile type.

Inheritance Hierarchy

SystemObject  KGySoft.Drawing.MetafileExtensions

Namespace: KGySoft.Drawing

Syntax

```csharp
public static class MetafileExtensions
```

The MetafileExtensions type exposes the following members.

Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save(Metafile, Stream)</td>
<td>Saves a Metafile instance into a Stream. Actual format is selected by the raw format of the metafile.</td>
</tr>
<tr>
<td>Save(Metafile, Stream, Boolean)</td>
<td>Saves a Metafile instance into a Stream using the required format.</td>
</tr>
<tr>
<td>ToBitmap</td>
<td>Creates a Bitmap of a Metafile instance provided in the metafile parameter.</td>
</tr>
</tbody>
</table>

See Also

Reference

KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
The `MetafileExtensions` type exposes the following members.

### Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Save(Metafile, Stream)</code></td>
<td>Saves a <code>Metafile</code> instance into a <code>Stream</code>. Actual format is selected by the raw format of the metafile.</td>
</tr>
<tr>
<td><code>Save(Metafile, Stream, Boolean)</code></td>
<td>Saves a <code>Metafile</code> instance into a <code>Stream</code> using the required format.</td>
</tr>
<tr>
<td><code>ToBitmap</code></td>
<td>Creates a <code>Bitmap</code> of a <code>Metafile</code> instance provided in the <code>metafile</code> parameter.</td>
</tr>
</tbody>
</table>

---

### See Also

**Reference**

- `MetafileExtensions Class`
- `KGySoft.Drawing Namespace`

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](#) page.

Copyright © KGy SOFT. All rights reserved.
Overload List

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save(Metafile, Stream)</td>
<td>Saves a Metafile instance into a Stream. Actual format is selected by the raw format of the metafile.</td>
</tr>
<tr>
<td>Save(Metafile, Stream, Boolean)</td>
<td>Saves a Metafile instance into a Stream using the required format.</td>
</tr>
</tbody>
</table>

See Also

Reference

MetafileExtensions Class
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
MetafileExtensionsSave Method (Metafile, Stream)

Saves a Metafile instance into a Stream. Actual format is selected by the raw format of the metafile.

**Namespace:** KGySoft.Drawing  
**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

### Syntax

```csharp
public static void Save(
    this Metafile metafile,
    Stream stream
)
```

### Parameters

- **metafile**
  - Type: System.Drawing.Imaging.Metafile
  - The Metafile instance to save.

- **stream**
  - Type: System.IO.Stream
  - The Stream into the metafile should be saved.

### Usage Note

In Visual Basic and C#, you can call this method as an instance method on any object of type Metafile. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).

### See Also

- MetafileExtensions Class
- Save Overload
KGYSoft.Drawing Namespace

Find the complete KGY SOFT Libraries documentation at the KGY SOFT Docs page.
Copyright © KGY SOFT. All rights reserved.
MetafileExtensions\n
Save Method
(Metafile, Stream, Boolean)

Saves a Metafile instance into a Stream using the required format.

**Namespace:** KGySoft.Drawing

**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public static void Save(
    this Metafile metafile,
    Stream stream,
    bool forceWmfFormat
)
```

**Parameters**

- **metafile**
  - Type: System.Drawing.ImagingMetafile
  - The Metafile instance to save.

- **stream**
  - Type: System.IOStream
  - The Stream into the metafile should be saved.

- **forceWmfFormat**
  - Type: SystemBoolean
  - When , forces to use the Windows Metafile Format (WMF), even if the metafile itself is encoded by Enhanced Metafile Format (EMF). When , uses the appropriate format automatically.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type Metafile. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension Methods (Visual Basic) or Extension Methods (C# Programming Guide).
See Also

Reference

MetafileExtensions Class
Save Overload
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.
MetafileExtensionsToBitmap Method

Creates a Bitmap of a Metafile instance provided in the metafile parameter.

**Namespace:** KGySoft.Drawing

**Assembly:** KGySoft.Drawing (in KGySoft.Drawing.dll) Version: 4.6.1

**Syntax**

```csharp
public static Bitmap ToBitmap(
    this Metafile metafile,
    Size requestedSize,
    bool antiAliased = false
)
```

**Parameters**

- **metafile**
  - Type: System.Drawing.Imaging.Metafile
  - The Metafile to convert.

- **requestedSize**
  - Type: System.Drawing.SizeF
  - The requested size of the result Bitmap.

- **antiAliased** (Optional)
  - Type: System.Boolean
  - to create an anti-aliased result; otherwise, .
  - Default value: .

**Return Value**

- Type: Bitmap
  - A Bitmap instance of the requested size.

**Usage Note**

In Visual Basic and C#, you can call this method as an instance method on any object of type Metafile. When you use instance method syntax to call this method, omit the first parameter. For more information, see Extension
Methods (Visual Basic) or Extension Methods (C# Programming Guide).

**See Also**

Reference

MetafileExtensions Class

KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](#) page.

Copyright © KGy SOFT. All rights reserved.
StockIcon Enumeration

Represents the Windows stock icons can be retrieved by the Icons.GetStockIcon method in Windows Vista and above. See also the SHSTOCKICONID Enumeration at the Microsoft Docs site.

Namespace: KGySoft.Drawing

Syntax

```csharp
public enum StockIcon
```

Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DocNoAssoc</td>
<td>0</td>
<td>Document of a type with no associated application.</td>
</tr>
<tr>
<td>DocAssoc</td>
<td>1</td>
<td>Document of a type with an associated application.</td>
</tr>
<tr>
<td>Application</td>
<td>2</td>
<td>Generic application with no custom icon.</td>
</tr>
<tr>
<td>Folder</td>
<td>3</td>
<td>Folder (generic, unspecified state).</td>
</tr>
<tr>
<td>FolderOpen</td>
<td>4</td>
<td>Folder (open).</td>
</tr>
<tr>
<td>Drive525</td>
<td>5</td>
<td>5.25-inch disk drive.</td>
</tr>
<tr>
<td>Drive35</td>
<td>6</td>
<td>3.5-inch disk drive.</td>
</tr>
<tr>
<td>DriveRemove</td>
<td>7</td>
<td>Removable drive.</td>
</tr>
<tr>
<td>DriveFixed</td>
<td>8</td>
<td>Fixed drive (hard disk).</td>
</tr>
<tr>
<td>DriveNet</td>
<td>9</td>
<td>Network drive (connected).</td>
</tr>
<tr>
<td>DriveNetDisabled</td>
<td>10</td>
<td>Network drive (disconnected).</td>
</tr>
<tr>
<td>DriveCD</td>
<td>11</td>
<td>CD drive.</td>
</tr>
<tr>
<td>DriveRam</td>
<td>12</td>
<td>RAM disk drive.</td>
</tr>
<tr>
<td>World</td>
<td>13</td>
<td>The entire network.</td>
</tr>
<tr>
<td><strong>Server</strong></td>
<td>15</td>
<td>A computer on the network.</td>
</tr>
<tr>
<td>---------------</td>
<td>----</td>
<td>----------------------------</td>
</tr>
<tr>
<td><strong>Printer</strong></td>
<td>16</td>
<td>A local printer or print destination.</td>
</tr>
<tr>
<td><strong>MyNetwork</strong></td>
<td>17</td>
<td>The Network virtual folder</td>
</tr>
<tr>
<td><strong>Find</strong></td>
<td>22</td>
<td>The Search feature.</td>
</tr>
<tr>
<td><strong>Help</strong></td>
<td>23</td>
<td>The Help and Support feature.</td>
</tr>
<tr>
<td><strong>Share</strong></td>
<td>28</td>
<td>Overlay for a shared item.</td>
</tr>
<tr>
<td><strong>Link</strong></td>
<td>29</td>
<td>Overlay for a shortcut.</td>
</tr>
<tr>
<td><strong>SlowFile</strong></td>
<td>30</td>
<td>Overlay for items that are expected to be slow to access.</td>
</tr>
<tr>
<td><strong>Recycler</strong></td>
<td>31</td>
<td>The Recycle Bin (empty).</td>
</tr>
<tr>
<td><strong>RecyclerFull</strong></td>
<td>32</td>
<td>The Recycle Bin (not empty).</td>
</tr>
<tr>
<td><strong>MediaCDAudio</strong></td>
<td>40</td>
<td>Audio CD media.</td>
</tr>
<tr>
<td><strong>Lock</strong></td>
<td>47</td>
<td>Security lock.</td>
</tr>
<tr>
<td><strong>AutoList</strong></td>
<td>49</td>
<td>A virtual folder that contains the results of a search.</td>
</tr>
<tr>
<td><strong>PrinterNet</strong></td>
<td>50</td>
<td>A network printer.</td>
</tr>
<tr>
<td><strong>ServerShare</strong></td>
<td>51</td>
<td>A server shared on a network.</td>
</tr>
<tr>
<td><strong>PrinterFax</strong></td>
<td>52</td>
<td>A local fax printer.</td>
</tr>
<tr>
<td><strong>PrinterFaxNet</strong></td>
<td>53</td>
<td>A network fax printer.</td>
</tr>
<tr>
<td><strong>PrinterFile</strong></td>
<td>54</td>
<td>A file that receives the output of a Print to file operation.</td>
</tr>
<tr>
<td><strong>Stack</strong></td>
<td>55</td>
<td>A category that results from a Stack by command to organize the contents of a folder.</td>
</tr>
<tr>
<td><strong>MediaSvcd</strong></td>
<td>56</td>
<td>Super Video CD (SVCD) media.</td>
</tr>
<tr>
<td><strong>StuffedFolder</strong></td>
<td>57</td>
<td>A folder that contains only subfolders as child items.</td>
</tr>
<tr>
<td><strong>DriveUnknown</strong></td>
<td>58</td>
<td>Unknown drive type.</td>
</tr>
<tr>
<td><strong>DriveDvd</strong></td>
<td>59</td>
<td>DVD drive.</td>
</tr>
<tr>
<td><strong>MediaDvd</strong></td>
<td>60</td>
<td>DVD media.</td>
</tr>
<tr>
<td><strong>MediaDvdRam</strong></td>
<td>61</td>
<td>DVD-RAM media.</td>
</tr>
<tr>
<td><strong>MediaDvdRW</strong></td>
<td>62</td>
<td>DVD-RW media.</td>
</tr>
<tr>
<td><strong>MediaDvdR</strong></td>
<td>63</td>
<td>DVD-R media.</td>
</tr>
<tr>
<td><strong>MediaDvdRom</strong></td>
<td>64</td>
<td>DVD-ROM media.</td>
</tr>
<tr>
<td><strong>MediaCDAudioPlus</strong></td>
<td>65</td>
<td>CD+ (enhanced audio CD) media.</td>
</tr>
<tr>
<td><strong>MediaCDRW</strong></td>
<td>66</td>
<td>CD-RW media.</td>
</tr>
<tr>
<td>Term</td>
<td>Page</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MediaCDR</td>
<td>67</td>
<td>CD-R media.</td>
</tr>
<tr>
<td>MediaCDBurn</td>
<td>68</td>
<td>A writeable CD in the process of being burned.</td>
</tr>
<tr>
<td>MediaBlankCD</td>
<td>69</td>
<td>Blank writable CD media.</td>
</tr>
<tr>
<td>MediaCDRom</td>
<td>70</td>
<td>CD-ROM media.</td>
</tr>
<tr>
<td>AudioFiles</td>
<td>71</td>
<td>An audio file.</td>
</tr>
<tr>
<td>ImageFiles</td>
<td>72</td>
<td>An image file.</td>
</tr>
<tr>
<td>VideoFiles</td>
<td>73</td>
<td>A video file.</td>
</tr>
<tr>
<td>MixedFiles</td>
<td>74</td>
<td>A mixed file.</td>
</tr>
<tr>
<td>FolderBack</td>
<td>75</td>
<td>Folder back.</td>
</tr>
<tr>
<td>FolderFront</td>
<td>76</td>
<td>Folder front.</td>
</tr>
<tr>
<td>Shield</td>
<td>77</td>
<td>Security shield. Use for UAC prompts only.</td>
</tr>
<tr>
<td>Warning</td>
<td>78</td>
<td>The warning icon.</td>
</tr>
<tr>
<td>Information</td>
<td>79</td>
<td>The information icon.</td>
</tr>
<tr>
<td>Error</td>
<td>80</td>
<td>The error icon.</td>
</tr>
<tr>
<td>Key</td>
<td>81</td>
<td>Key.</td>
</tr>
<tr>
<td>Software</td>
<td>82</td>
<td>Software.</td>
</tr>
<tr>
<td>Rename</td>
<td>83</td>
<td>A UI item, such as a button, that issues a rename command.</td>
</tr>
<tr>
<td>Delete</td>
<td>84</td>
<td>A UI item, such as a button, that issues a delete command.</td>
</tr>
<tr>
<td>MediaAudioDvd</td>
<td>85</td>
<td>Audio DVD media.</td>
</tr>
<tr>
<td>MediamMoieDvd</td>
<td>86</td>
<td>Movie DVD media.</td>
</tr>
<tr>
<td>MediaEnhancedCD</td>
<td>87</td>
<td>Enhanced CD media.</td>
</tr>
<tr>
<td>MediaEnhancedDvd</td>
<td>88</td>
<td>Enhanced DVD media.</td>
</tr>
<tr>
<td>MediaHDDvd</td>
<td>89</td>
<td>High definition DVD media in the HD DVD format.</td>
</tr>
<tr>
<td>MediaBluray</td>
<td>90</td>
<td>High definition DVD media in the Blu-ray Disc™ format.</td>
</tr>
<tr>
<td>MediaVcd</td>
<td>91</td>
<td>Video CD (VCD) media.</td>
</tr>
<tr>
<td>MediaDvdPlusR</td>
<td>92</td>
<td>DVD+R media.</td>
</tr>
<tr>
<td>MediaDvdPlusRW</td>
<td>93</td>
<td>DVD+RW media.</td>
</tr>
<tr>
<td>DesktopPC</td>
<td>94</td>
<td>A desktop computer.</td>
</tr>
<tr>
<td>MobilePC</td>
<td>95</td>
<td>A mobile computer (laptop).</td>
</tr>
<tr>
<td>Users</td>
<td>96</td>
<td>The User Accounts Control Panel item.</td>
</tr>
<tr>
<td>MediaSmartMedia</td>
<td>97</td>
<td>Smart media.</td>
</tr>
<tr>
<td>MediaCompactFlash</td>
<td>98</td>
<td>CompactFlash media.</td>
</tr>
<tr>
<td>DeviceCellphone</td>
<td>99</td>
<td>A cell phone.</td>
</tr>
<tr>
<td>DeviceCamera</td>
<td>100</td>
<td>A digital camera.</td>
</tr>
<tr>
<td>DeviceVideoCamera</td>
<td>101</td>
<td>A digital video camera.</td>
</tr>
<tr>
<td>DeviceAudioPlayer</td>
<td>102</td>
<td>An audio player.</td>
</tr>
<tr>
<td>NetworkConnect</td>
<td>103</td>
<td>Connect to network.</td>
</tr>
<tr>
<td>Internet</td>
<td>104</td>
<td>The Network and Internet Control Panel item.</td>
</tr>
<tr>
<td>ZipFile</td>
<td>105</td>
<td>A compressed file with a .zip file name extension.</td>
</tr>
<tr>
<td>Settings</td>
<td>106</td>
<td>The Additional Options Control Panel item.</td>
</tr>
<tr>
<td>DriveHDDvd</td>
<td>132</td>
<td>Windows Vista with Service Pack 1 (SP1) and later. High definition DVD drive (any type - HD DVD-ROM, HD DVD-R, HD-DVD-RAM) that uses the HD DVD format.</td>
</tr>
<tr>
<td>DriveBD</td>
<td>133</td>
<td>Windows Vista with SP1 and later. High definition DVD drive (any type - BD-ROM, BD-R, BD-RE) that uses the Blu-ray Disc format.</td>
</tr>
<tr>
<td>MediaHDDvdRom</td>
<td>134</td>
<td>Windows Vista with SP1 and later. High definition DVD-ROM media in the HD DVD-ROM format.</td>
</tr>
<tr>
<td>MediaHDDvdR</td>
<td>135</td>
<td>Windows Vista with SP1 and later. High definition DVD-R media in the HD DVD-R format.</td>
</tr>
<tr>
<td>MediaHDDvdRam</td>
<td>136</td>
<td>Windows Vista with SP1 and later. High definition DVD-RAM media in the HD DVD-RAM format.</td>
</tr>
<tr>
<td>MediaBDRom</td>
<td>137</td>
<td>Windows Vista with SP1 and later. High definition DVD-ROM media in the Blu-ray Disc BD-ROM format.</td>
</tr>
<tr>
<td>MediaBDR</td>
<td>138</td>
<td>Windows Vista with SP1 and later. High definition write-once media in the Blu-ray Disc BD-R format.</td>
</tr>
<tr>
<td>MediaBDRE</td>
<td>139</td>
<td>Windows Vista with SP1 and later. High definition read/write media in the Blu-ray Disc BD-RE format.</td>
</tr>
<tr>
<td>ClusteredDrive</td>
<td>140</td>
<td>Windows Vista with SP1 and later. A cluster disk array.</td>
</tr>
</tbody>
</table>

### See Also

Reference

KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the [KGy SOFT Docs](#) page.
SystemIconSize Enumeration

Represents the predefined system icon sizes.

Namespace: KGySoft.Drawing

Syntax

C# | VB | C++ | F#
---|----|-----|-----
public enum SystemIconSize

Members

<table>
<thead>
<tr>
<th>Member name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Large</td>
<td>0</td>
<td>Represents the larger version of an icon. With 100% scale settings (96 DPI) this usually means the 32x32 system icon size.</td>
</tr>
<tr>
<td>Small</td>
<td>1</td>
<td>Represents the smaller version of an icon. With 100% scale settings (96 DPI) this usually means the 16x16 system icon size.</td>
</tr>
</tbody>
</table>

See Also

Reference
KGySoft.Drawing Namespace

Find the complete KGy SOFT Libraries documentation at the KGy SOFT Docs page.
Copyright © KGy SOFT. All rights reserved.