# HOWTO: Print Multiple Images

The following is a simple example how to print multiple images to a single page. The key concepts to takeaway is the setting of the GetImage event. Images would print one per page if we used the ImagePrintDocument's SetImagesFromImageCollection() method. By overriding the GetImage event it allows us to control how the printing occurs. Another important method is the AtalaImage.Draw(). It is overloaded so it has a variety of parameter options. There is an important parameter in the method called renderBottomTop. Please be advised this will default to false when using one of the overloaded methods that does not pass this value. In order to print color images a value of true must be used.

```
sing System; using System.Text; using System.Drawing; using System.Drawing.Printing; using
Atalasoft.Imaging; using Atalasoft.Imaging.WinControls; namespace ConsolePrintTest { class
Program { static void Main(string[] args) { ImageCollection coll = new ImageCollection();
coll.Add(new AtalaImage("Test1.bmp")); coll.Add(new AtalaImage("Test2.bmp")); PrintImageTest
pt = new PrintImageTest(coll); pt.PrintImages(); } } public class PrintImageTest { private
ImagePrintDocument _printObj; private ImageCollection _images; public
PrintImageTest(ImageCollection images) { _printObj = new ImagePrintDocument(); _images =
images; } public void PrintImages() { _printObj.GetImage += new
PrintImageEventHandler(AtalaPrint_GetImage); _printObj.Print(); } void
AtalaPrint_GetImage(object sender, PrintImageEventArgs e) { int x = 0; int y = 0; for (int i
= 0; i < _images.Count; i++) { AtalaImage img = _images[i]; img.Draw(e.Graphics, new
System.Drawing.Rectangle(x, y, Convert.ToInt32(e.Graphics.DpiX),
Convert.ToInt32(e.Graphics.DpiY)), new System.Drawing.Rectangle(Point.Empty, img.Size),
true); y = y + 1000; } e.Graphics.DrawString("Put Some Text there", new
System.Drawing.Font("Arial", 12), Brushes.Black, new PointF(300, 300)); } } }
```

Original Article:

Q10205 - HOWTO: Print multiple images