

برنامج معالجة صور باستخدام لغة الفيجول بيسك

الصورى الرقمية: هي تمثيل للصور الثنائية الأبعاد على الحاسوب بواسطة الصفر والواحد (01). وتتكون كل صورة رقمية على الكمبيوتر من البيكسل وهو أصغر وحدة في الصورة. وكل صورة هي مصفوفة تحتوى على صفوف وأعمدة من البيكسلات وكلما زادت عدد البيكسلات كلما كانت الصورة أوضح

Digital image: function of 2 variables, $f(x,y)$, where x and y are spatial coordinates,

Pixels : Elements of the digital image , each has intensity

Note: images can be: binary, grayscale, color.

The image consists of finite number of pixels ($f(x,y)$)

Every pixel is an intersection تقاطع between a row and a column.

every pixel has intensity كثافة

$f(4,3) = 123$

Refers to a pixel existing on the intersection between row 4 with column 3, and its intensity is 123.

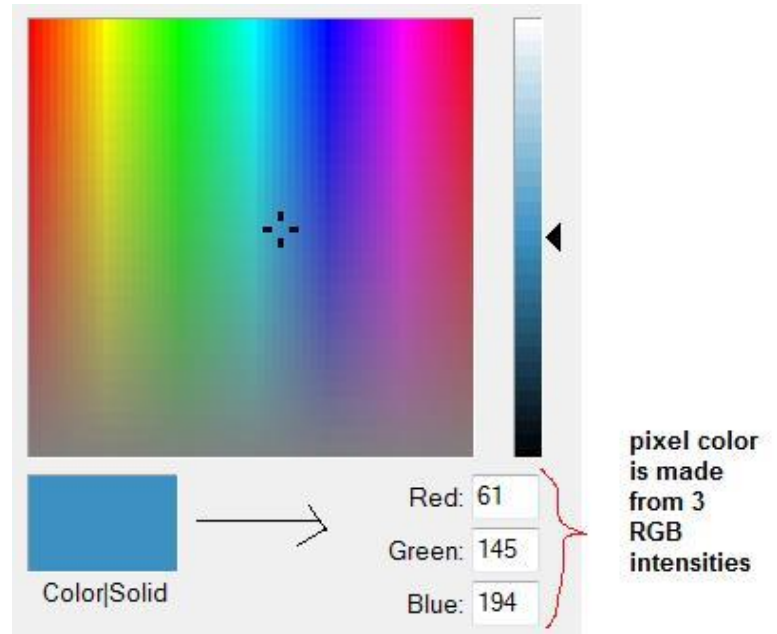
Grayscale Images

A grayscale (or graylevel) image is simply one in which the only colors are shades of gray (255 – 0)

Color Images

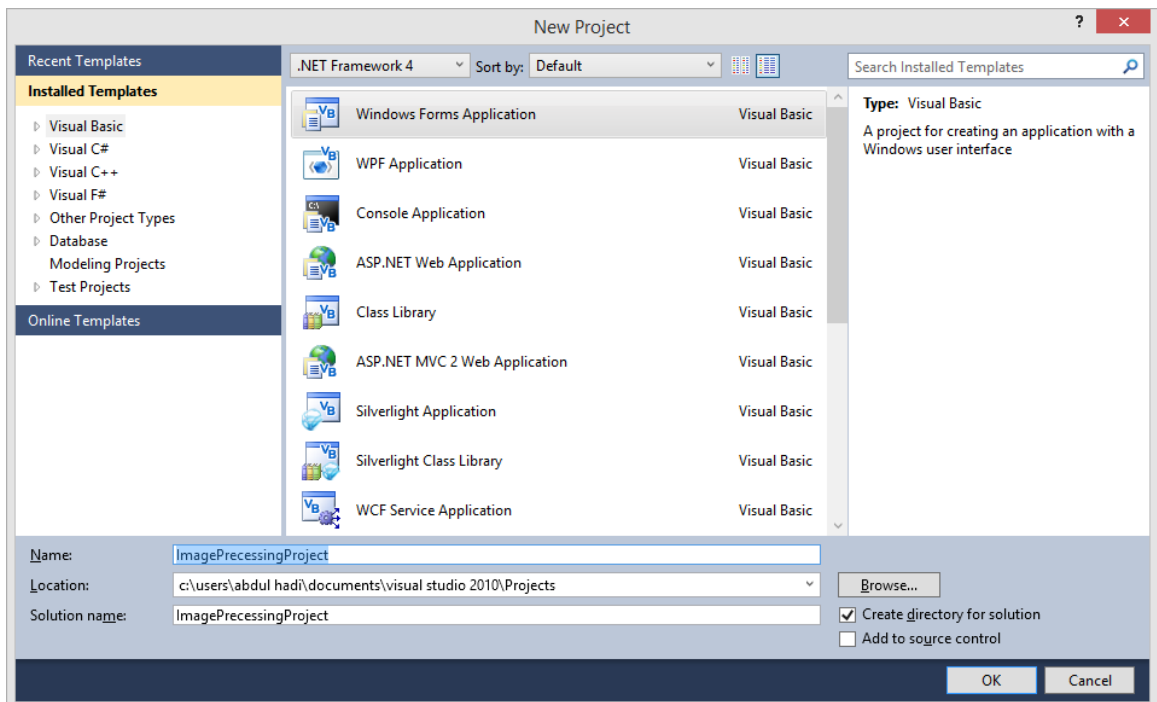
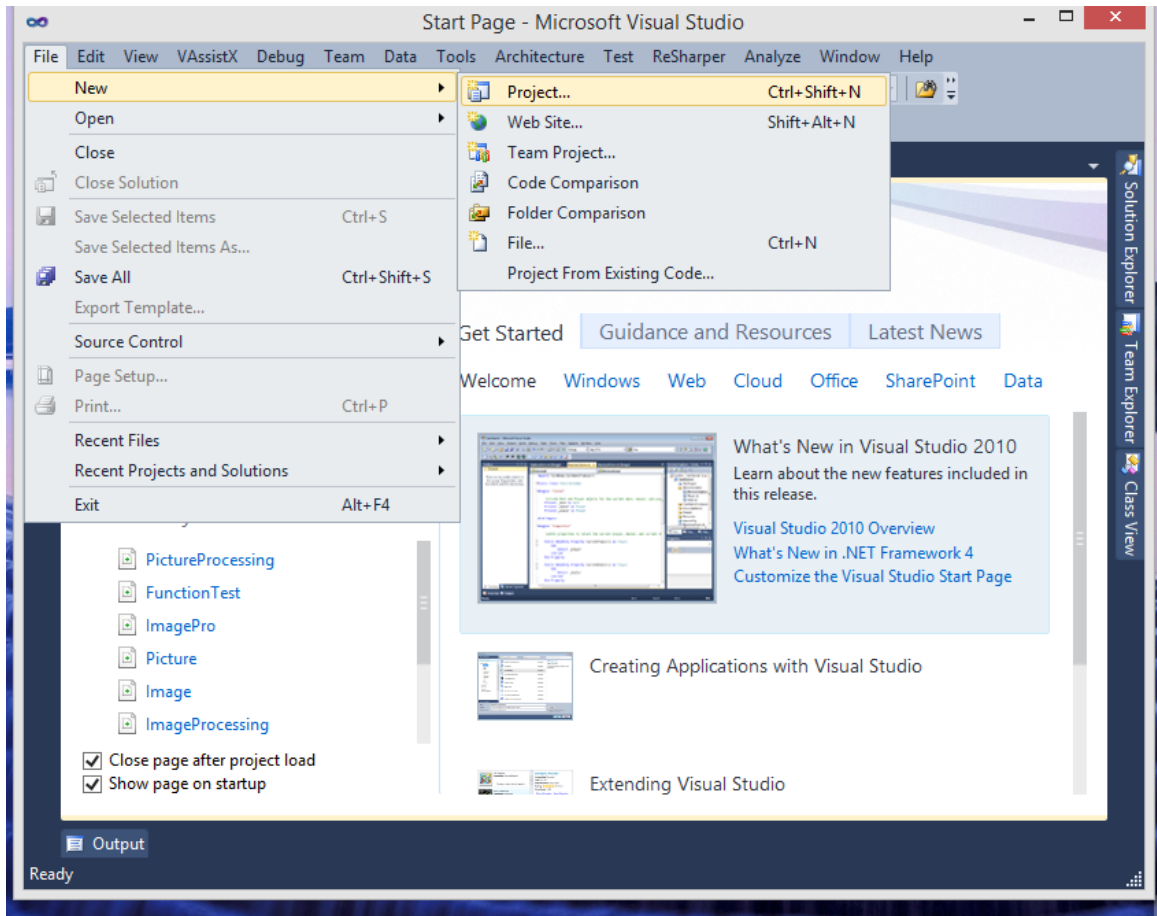


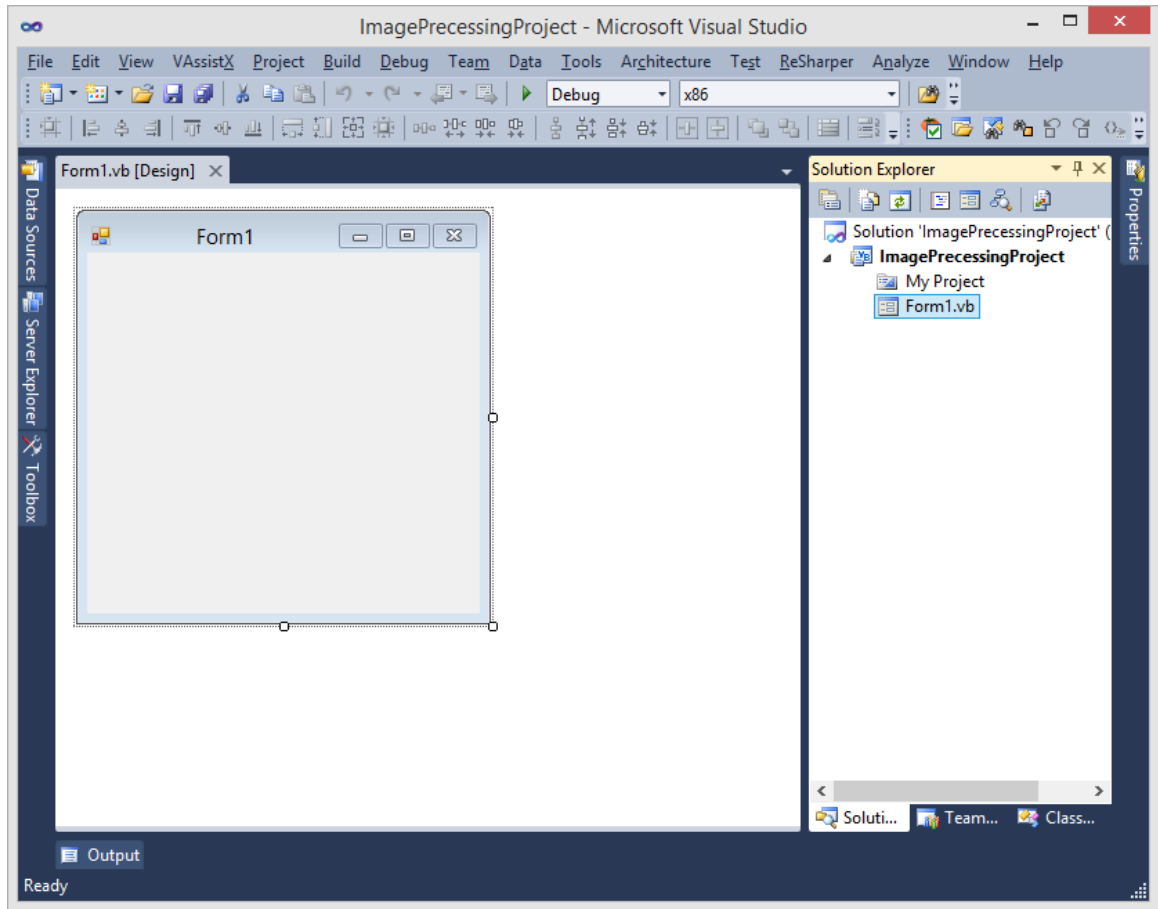
Color image: A color image contains pixels each of which holds three intensity values corresponding to the red, green, and blue or (RGB)



تصميم البرنامج

من قائمة File اضيف مشروعاً جديداً باسم (ImageProcessingProject)





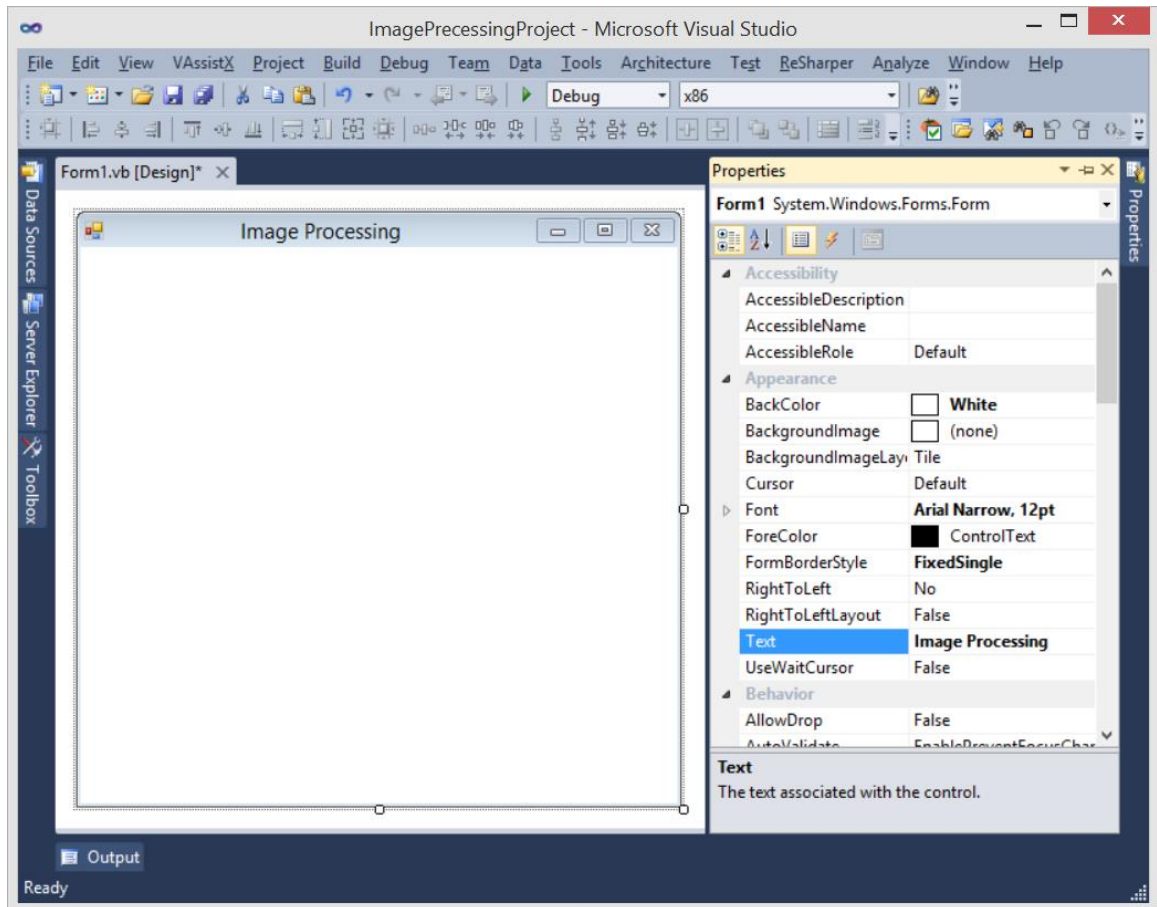
نقوم بتغيير خصائص الفورم الى

BackColor :White

Font: Arial ,12pt للتحكم فى شكل وحجم الخط

FormBorderStyle: fixedSingle للتحكم فى شكل حدود نافذة النموذج

Text: Image Processing



بعد ذلك نضيف MenuStrip الى الفوم لاضافة قائمة ثم نضيف القوائم

&File

&Open

&Save As ...

-

&Exit

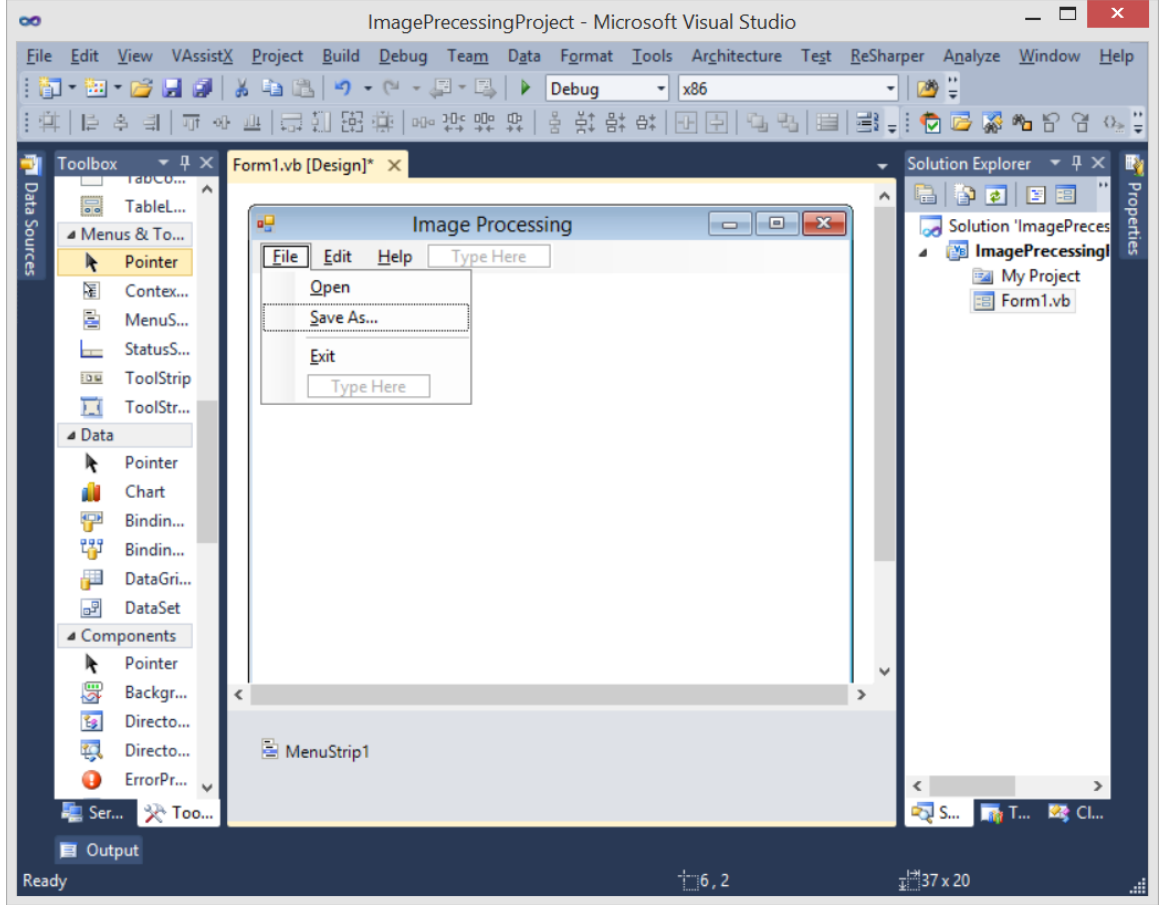
&Edit

&Convert Image To greyscale

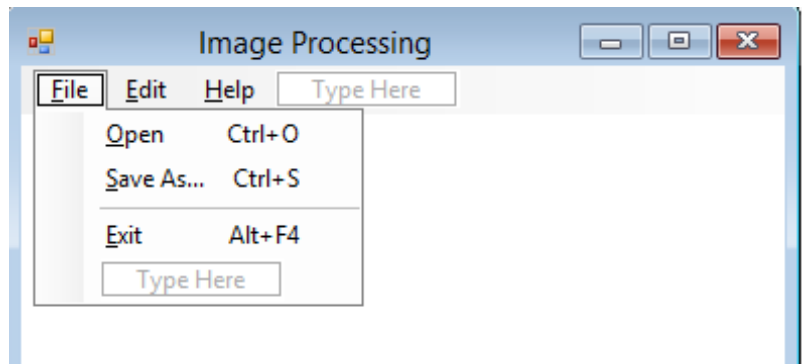
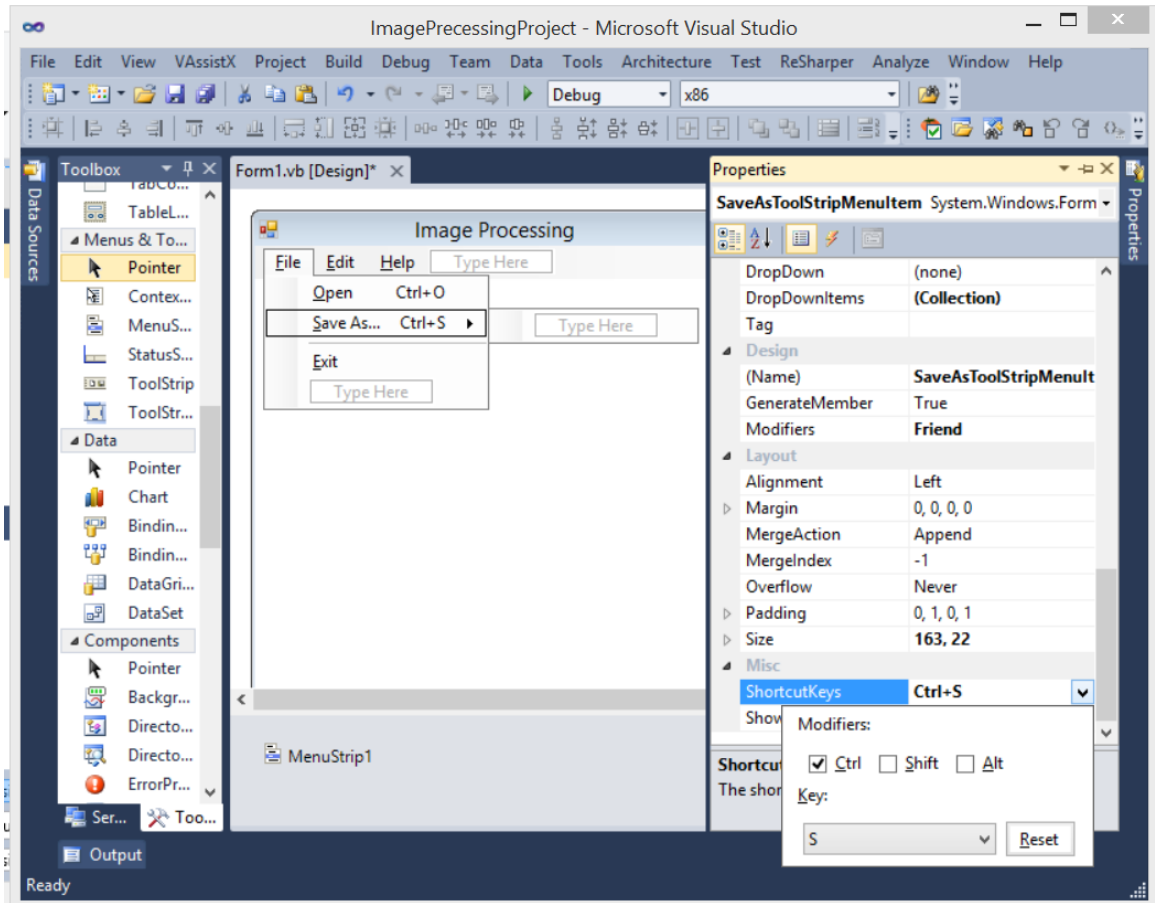
&Convert Image To black && white

&Help

& About Program

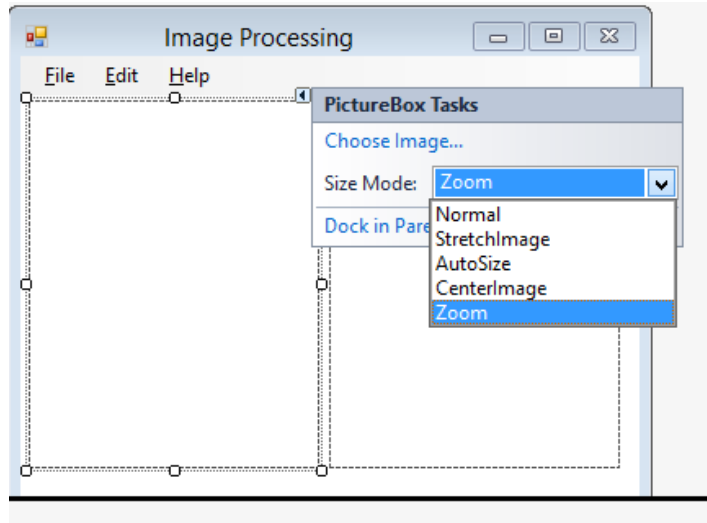


نضيف shortcutkeys للقوائم نختار القائمة ثم نختار ال shortcutkeys من ال
Properties



بعد ذلك نضيف
PictureBox ونغير
خصائصهما الى

- 1-
Name: picOriginal
Size Mode: Zoom
- 2-
Name: picModified
Size Mode: Zoom

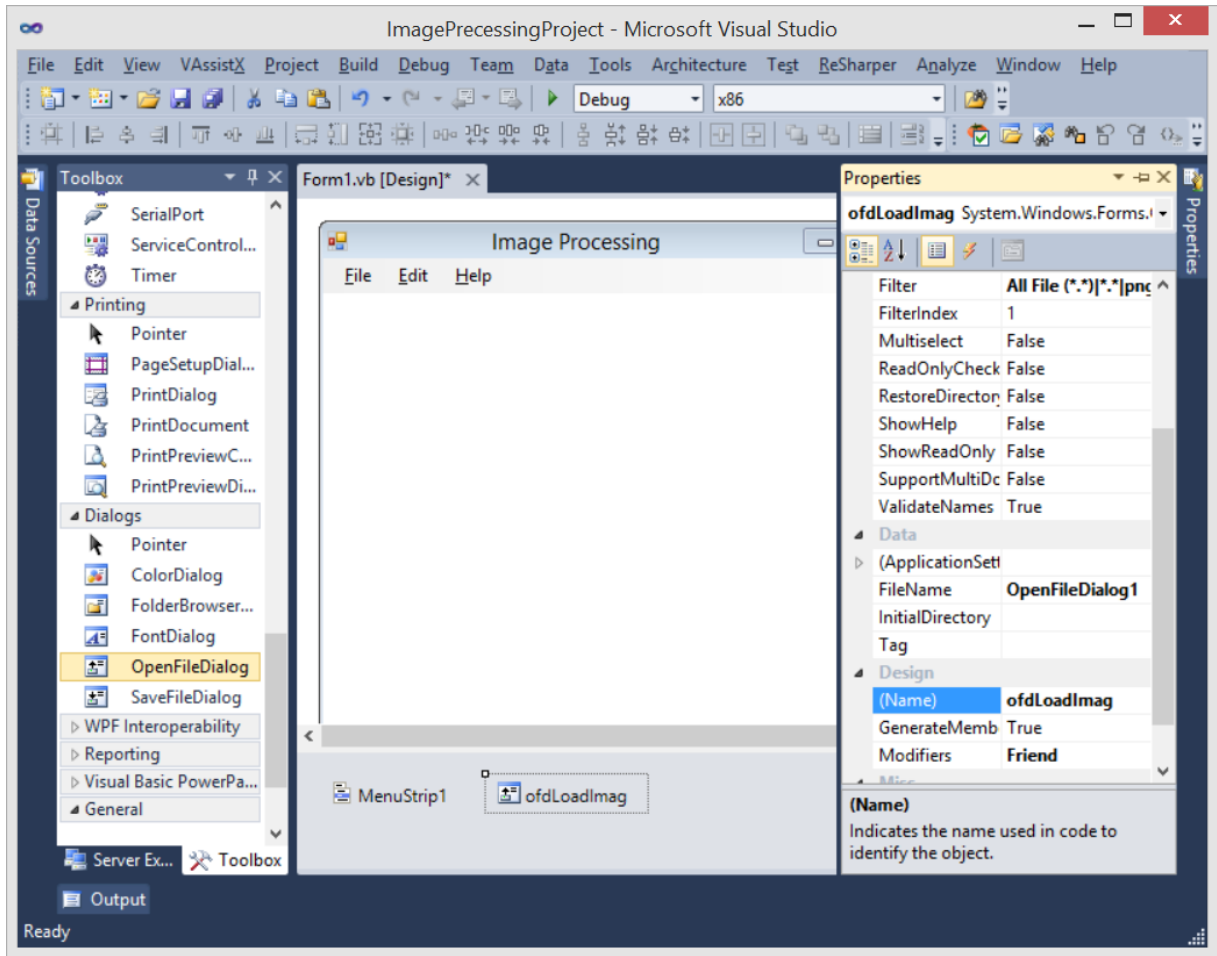


نضيف open file dialog (يستخدم لاضهار نافذة اضافة ملف الى البرنامج كما في برامج
الافيس وغيرها) ونغير الخصائص الى :

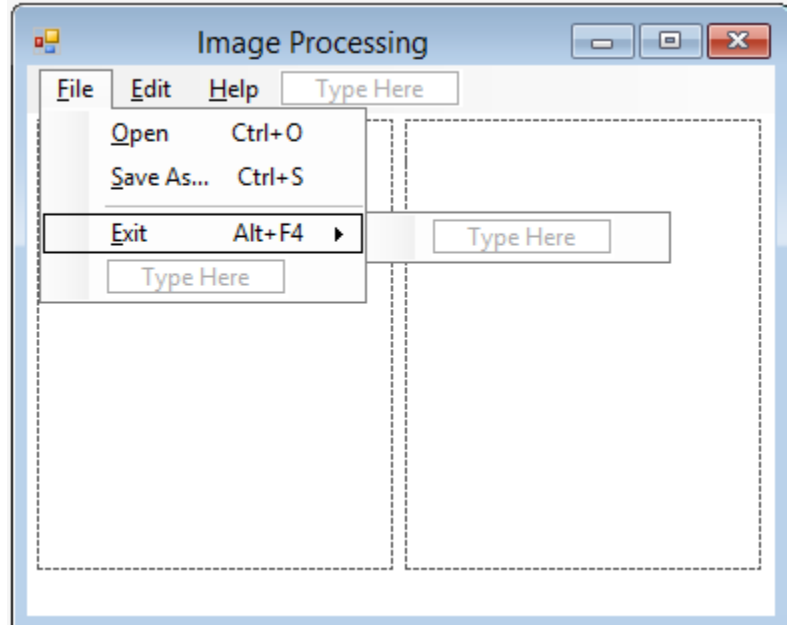
Name: ofdLoadImag

Filter: All File (*.*)| *.* | png Image (*.png) | *.png | jpg Image (*.jpg) | *.jpg

ملاحظة ان استخدام الفلتر لاضهار امتداد الصور فقط او كل الملفات

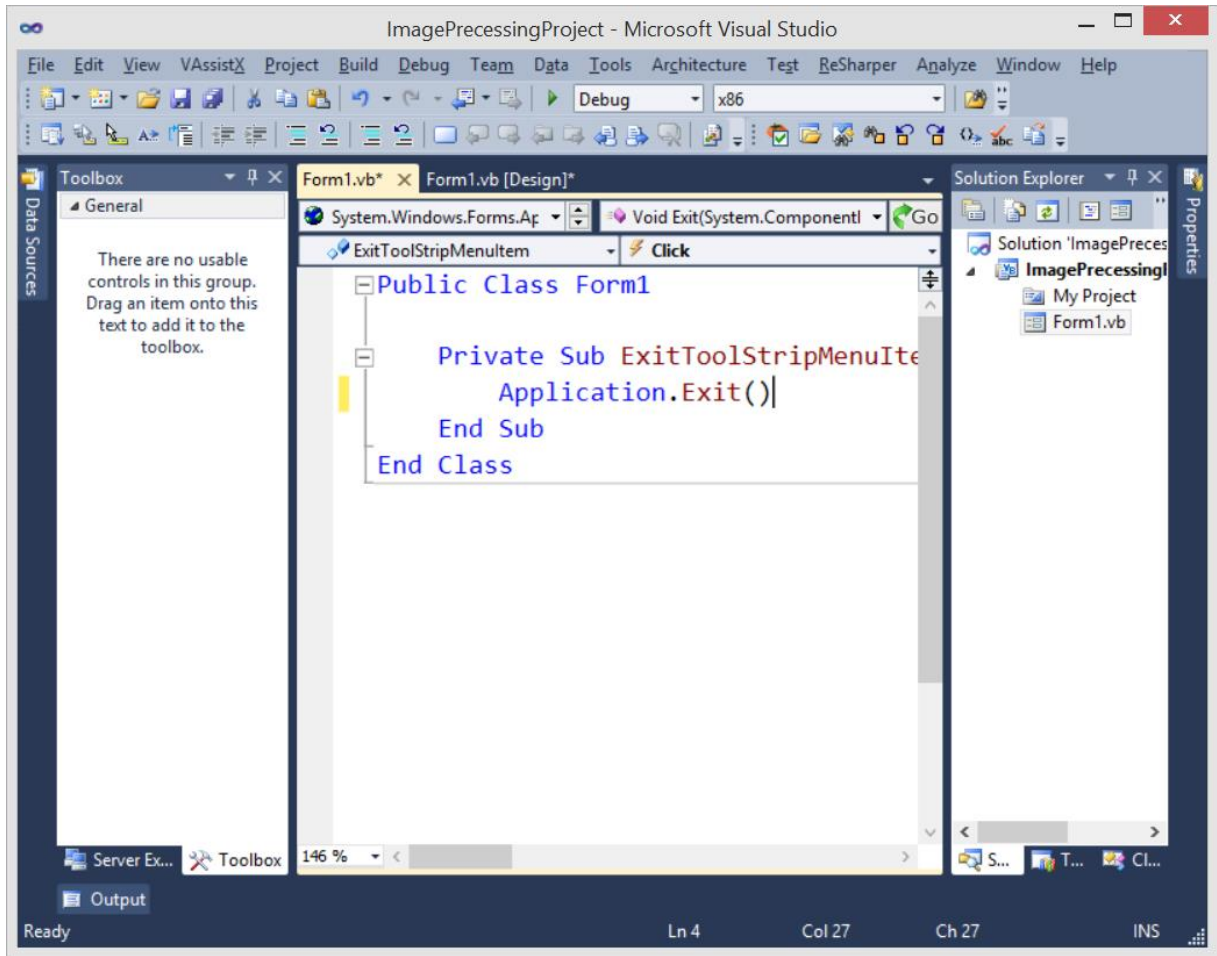


الآن بعد اكمال الواجهة نبدأ بكتابة الكود نضغط دبل كليك على Exit لتفتح نافذة الكود

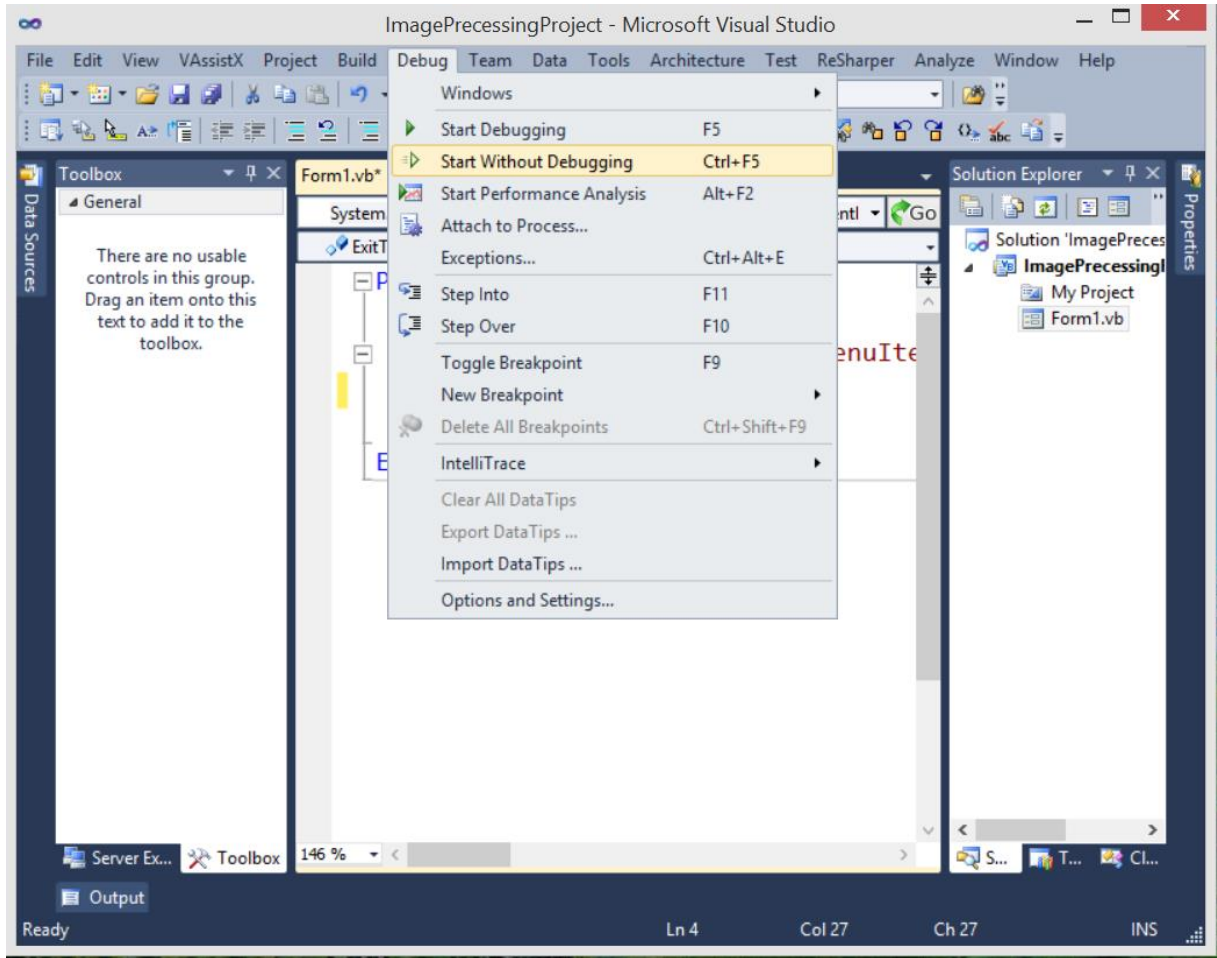


نكتب الكود الذي يقوم باغلاق البرنامج

`Application.Exit()`

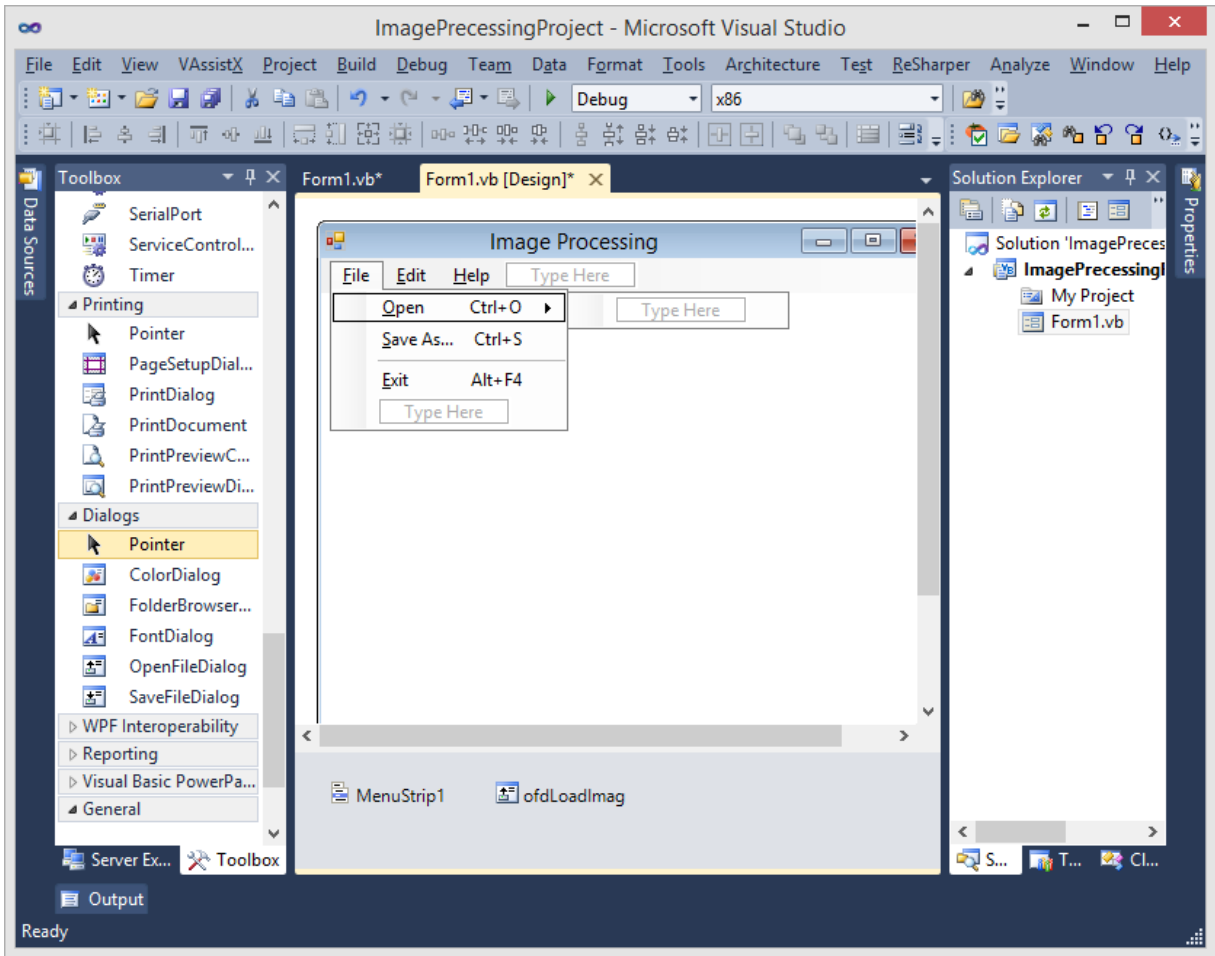


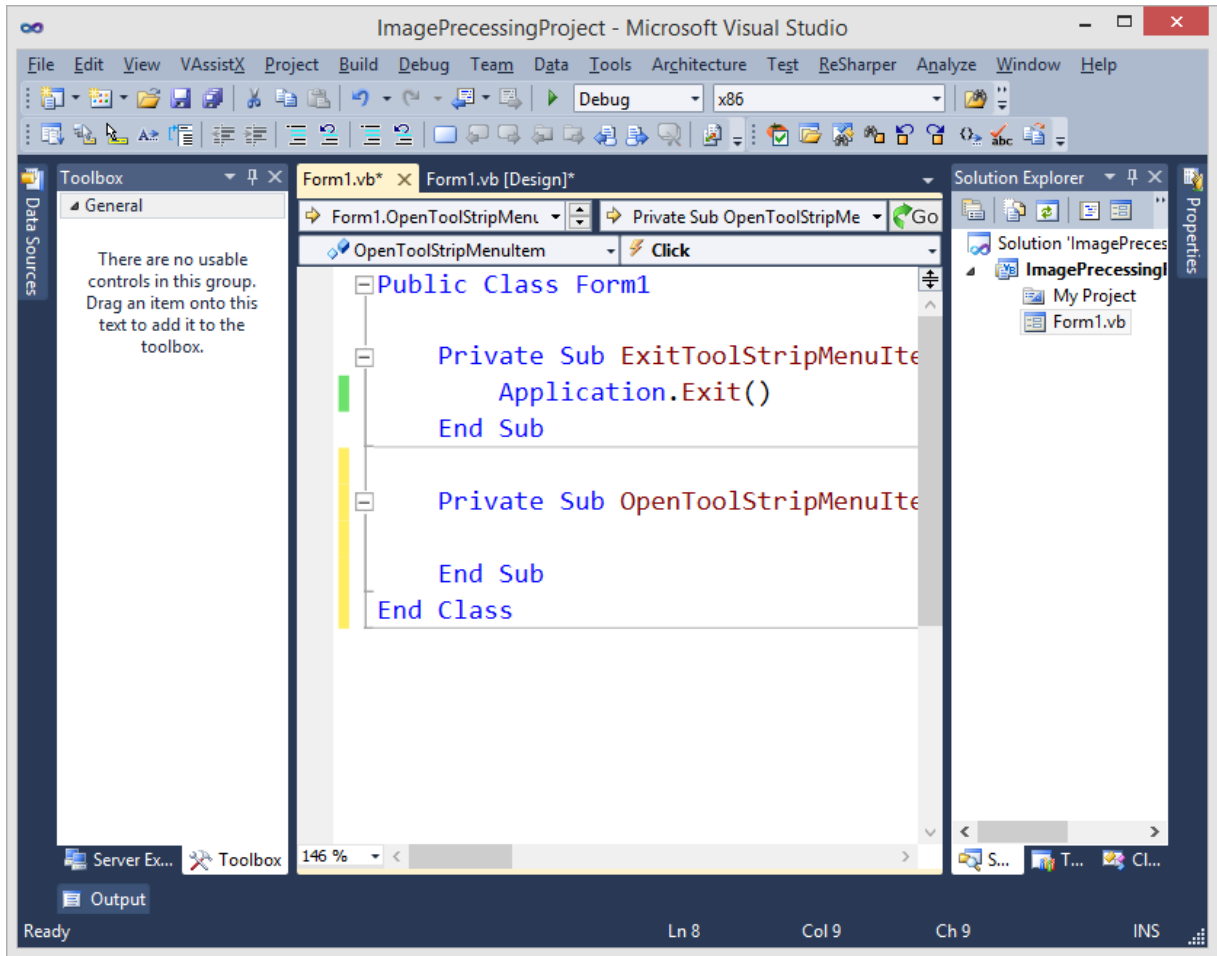
نقوم بتجربة البرنامج وذلك بعمل (Run) للبرنامج من قائمة (Debug) نختار (Start) (without debugging)



بعد ذلك نضيف الكود لـ Open بالرجوع الى ال نافذة Design ونضغط دبلل كليك على

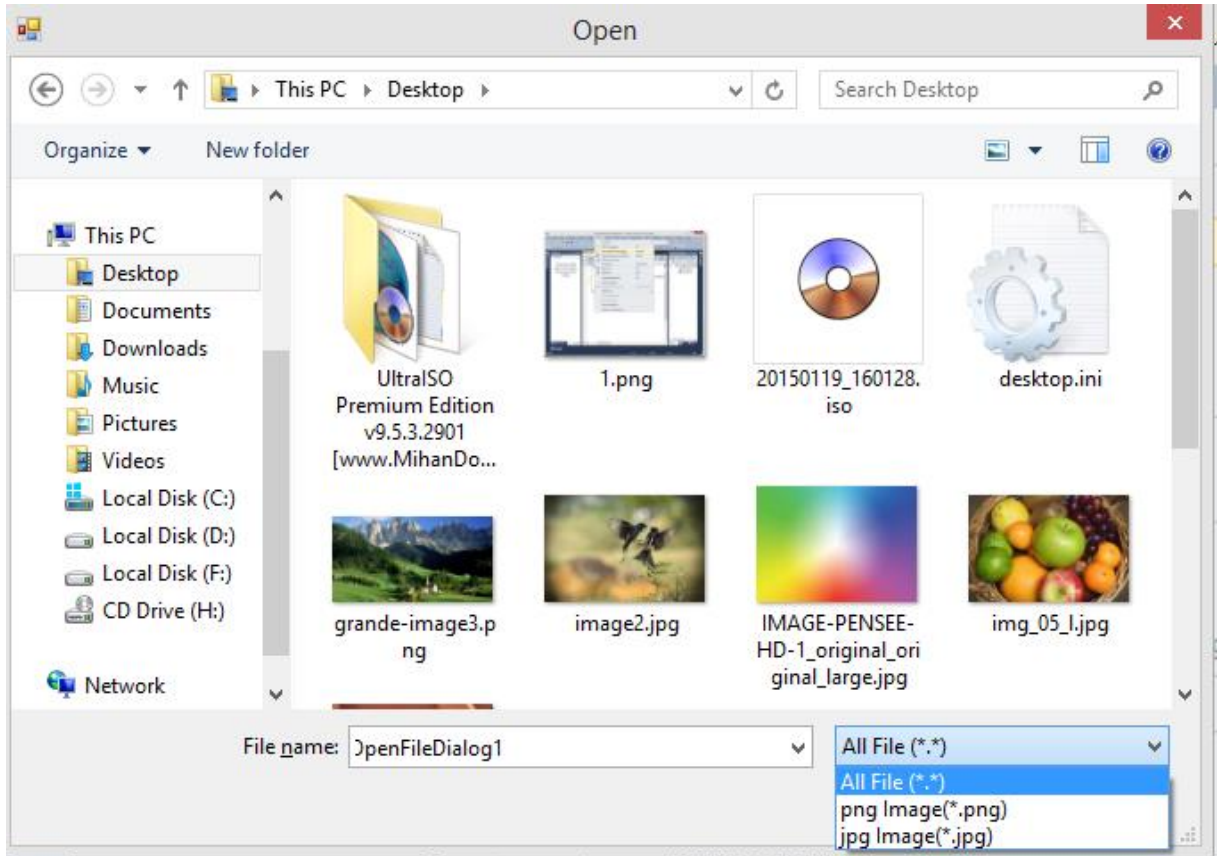
Open





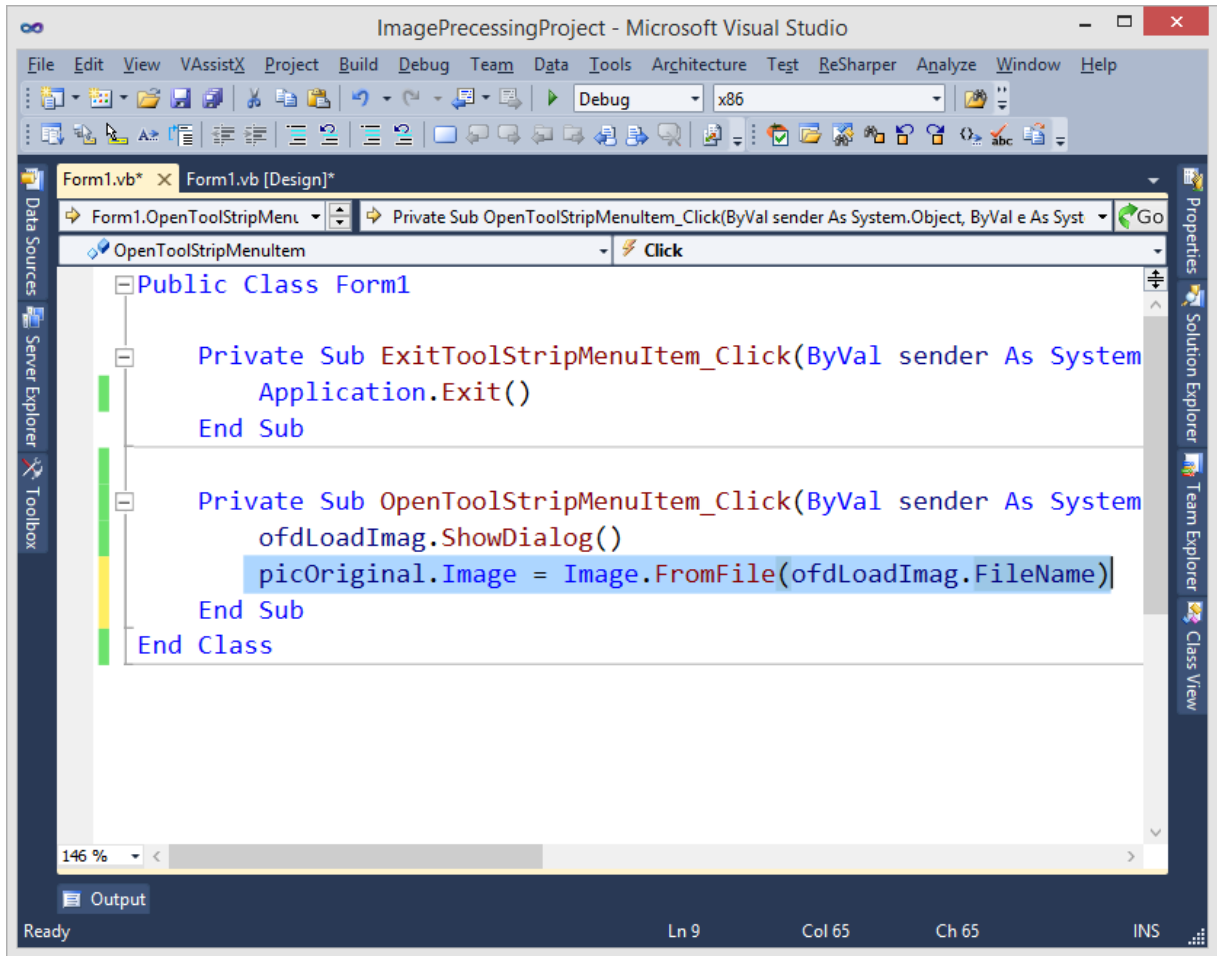
ان عملية فتح نافذة اختيار الصورة تتم حسب الكود التالي

ofdLoadImag.ShowDialog()

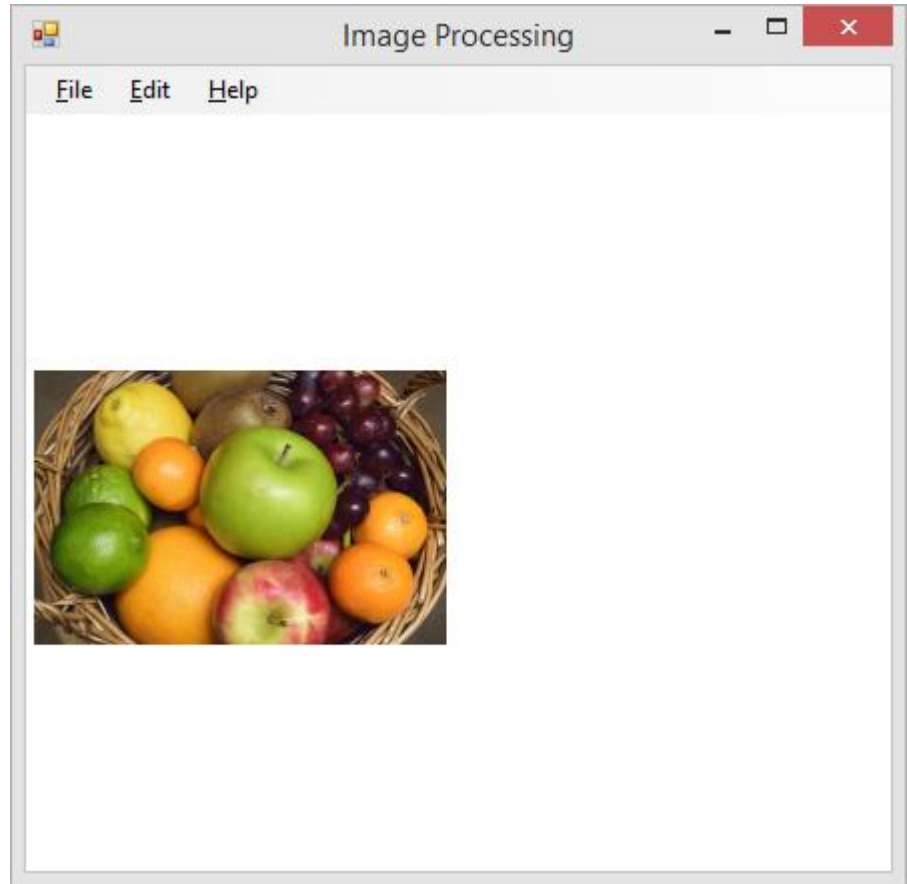


حيث ستظهر نافذة Open وتلاحظ ان الفلتر يحدد نوع الصور

ولكن عن اختيار صورة و Open لم يتم اي شي لان open file dialog تقوم بحفظ ال Path للصورة ولتحميل الصورة داخل ال PictureBox1 نكتب الكود التالي
picOriginal.Image = Image.FromFile(ofdLoadImag.FileName)



الآن عند عمل Run للبرنامج والضغط على Open ثم نختار الصورة سوف تتحمل الصورة إلى PictureBox1 والذي تم تغيير اسمه إلى picOriginal



نبدأ الان بكتابة الكود الخاص بتحويل الصورة الى Grayscale image

ان الصورة تتكون من صفوف واعمدة من البكسيالات وكل بيكسل يتكون من ثلاثة الوان هي الالوان الرئيسية (Red, Green, Blue) ولتحويل الصورة الى Grayscale نقوم بقراءة كل بيكسل ويجاد المعدل للبيكسل ثم تحويل قيم ال (Red, Green, Blue) الى قيمة ال average

مثال

(200,100,30)	(150,16,80)
--------------	-------------

(20,220,80)	(0,20,0)
-------------	----------

نقوم بحساب المعدل لكل خلية

$$\text{Avg}=(200+100+30)/3$$

$$\text{Avg}=110$$

$$\text{Avg}=(150+16+80)$$

$$\text{Avg}=82$$

$$\text{Avg}=(20+220+80)/3$$

$$\text{Avg}=106.66666$$

$$\text{Avg}=(0+20+0)/3$$

$$\text{Avg}=6.666666$$

نقوم بابدال القيم في كل خلية بالمعدل المقابل لها

(110,110,110)	(82,82,82)
(106,106,106)	(6,6,6)

ان الصور عبارة عن مصفوفة ثنائية تبدأ من (0,0) وتنتهي ب $(N - 1 \times M - 1)$ حيث يبدأ الصف الاول ب 0 وينتهي ب $N - 1$ (N هي عرض الصورة) كما يبدأ العمود ب 0 وينتهي ب $M - 1$ (M هو ارتفاع الصورة)

صورة $(N \times M)$

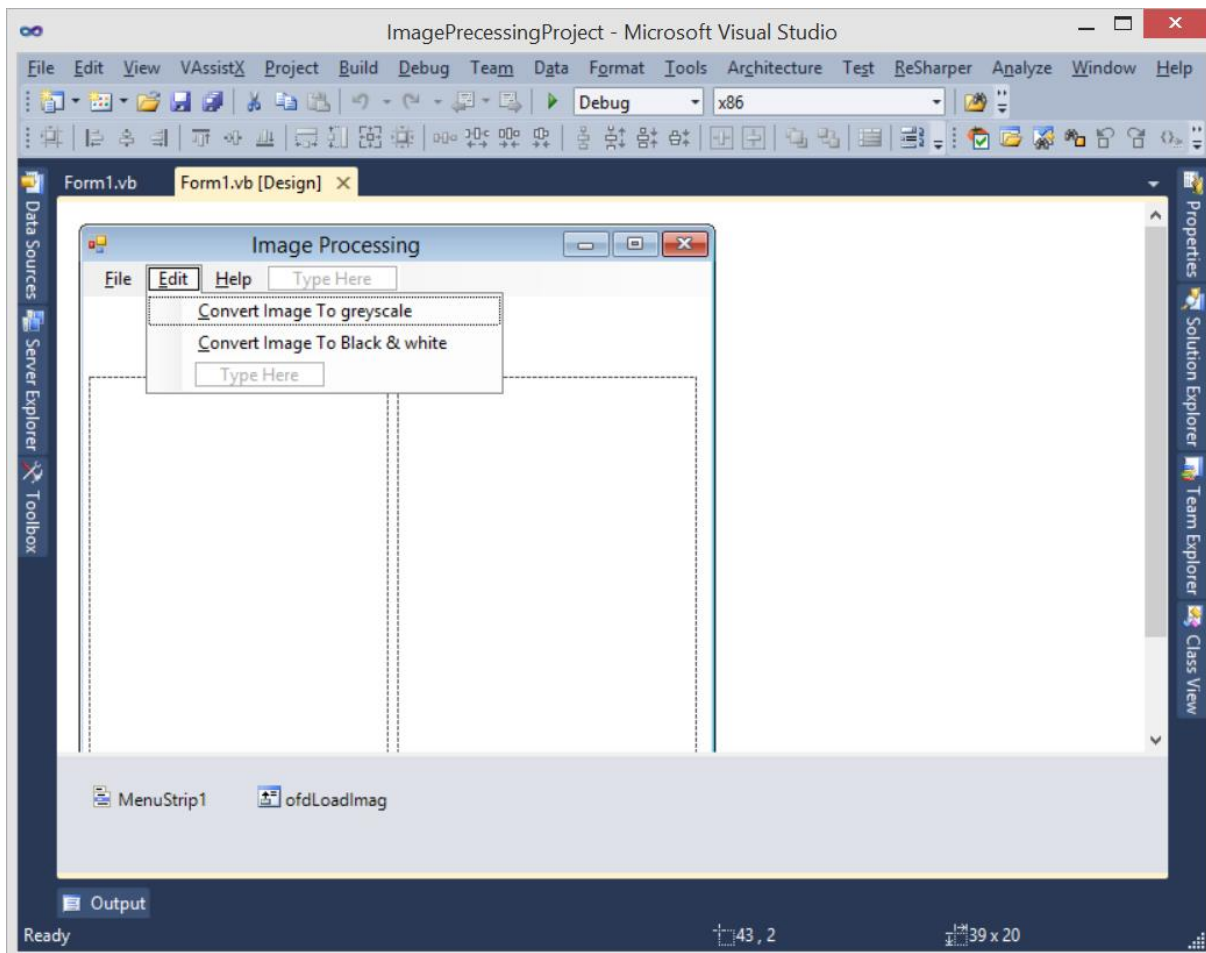
(0,0).....(0,M-1)

(1,0).....(1,M-1)

| |
| |
| |

(1-N,0).....(1-N,M-1)

ولكتابة الكود الخاص بتحويل الصورة الى تدرج رمادي نضغط دبلل كليك على Convert
to Grayscale من قائمة Edit في وضع التصميم



ونكتب الكود التالي

```
Dim width As Integer = picOriginal.Image.Width  
Dim height As Integer = picOriginal.Image.Height  
Dim tempPicture As New Bitmap(picOriginal.Image)
```

```

Dim p As Color
For x = 0 To width - 1
    For y = 0 To height - 1
        p = tempPicture.GetPixel(x, y)
        Dim r As Integer = p.R
        Dim g As Integer = p.G
        Dim b As Integer = p.B
        Dim avg As Integer = (r + g + b) / 3
tempPicture.SetPixel(x, y, Color.FromArgb(p.A, avg, avg, avg))
    Next
Next
picModified.Image = tempPicture

```

شرح الكود

السطر الاول والثاني لتعريف متغير نوع Integer لعرض وارتفاع الصورة

السطر الثالث تعريف متغير من نوع Bitmap ونسخ الصورة الموجودة في picOriginal اليه حيث ستكون هذه الصورة كصورة وقتية.

السطر الرابع تعريف متغير من نوع Color ل تخزين قيم RGB لكل بيكسل

ان الصورة عبارة عن مصفوفة ثنائية مثال صورة مكونة من صفان وثلاثة اعمدة

(200,100,30)	(150,16,80)	(200,100,30)
(20,220,80)	(0,20,0)	(0,40,20)

الان علينا قراءة الصف الاول الخلية الاولى وايجاد المعدل ثم ارجاع القيمة في نفس المكان

او الى مصفوفة جديدة

نفرض ان اسم المصفوفة هو A وان $A(0,0)$ يساوي القيمة داخل البيكسل الاول
(200,100,30), قيمة $A(0,1)$ تساوي البيكسل الثاني (150,16,80) وهكذا

لمعالجة البيكسيالات نقوم بقراءة بيكسل بيكسل بداية من الصفر

$P=A.$ GetPixel (0,0)

الان P تحوي قيمة البيكسل الاول (200,100,30)

$r=P.R$

نسخ قيمة اللون الاحمر من الصف الاول الى متغير اسمة r فتكون قيمة r هي 200

$g=P.G$

نسخ قيمة اللون الاخضر من الصف الاول الى متغير اسمة g فتكون قيمة g هي 100

$b=P.B$

نسخ قيمة اللون الازرق من الصف الاول الى متغير اسمة b فتكون قيمة b هي 30

بعد تحصيل القيم الثلاثة نجد المعدل

$Avg=(r+b+g)/3$

بعد ايجاد المعدل نقوم بتعريف مصفوفة جديدة باسم X وننقل القيم الجديدة اليها

$X(0,0).$ SetPixel (avg,avg,avg)

الان $X(0,0)$ تحتوي على (110,110,110)

بعد اكمال البيكسل الاول نقوم بتغير البيكسل الثاني (الخلية الثانية)

$P=A.$ GetPixel (0,1)

$r=P.R$

$g=P.G$

b=P.B

Avg=(r+b+g)/3

X(0,1). SetPixel (avg,avg,avg)

وهكذا حتى اكمال جميع عناصر الصف الاول بعد ذلك نقوم بمعالجة البيكسل الاول في العمود الثاني

P=A. GetPixel (1,0)

r=P.R

g=P.G

b=P.B

Avg=(r+b+g)/3

X(1,0). SetPixel (avg,avg,avg)

نلاحظ ان الخطوات تتكرر باستثناء رقم الصف والعمود لهذا نستخدم ال For Loop حيث ان ال loop هو تكرار لعملية معينة محصورة بين For و Next

For x = بداية العداد To نهاية العداد

تكرار العمليات

Next

مثال تكرار عملية معينة 100 مرة

For i = 0 To 99

تكرار العمليات

Next

Nested Loop هو عداد داخل عداد

For **x** = بداية العداد To نهاية العداد الاول

For **y** = بداية العداد To نهاية العداد الثاني

تكرار العمليات

Next

Next

الان باستخدام nested loop يصبح الكود كالتالي

For **x** =0 To عرض الصورة

For **y** =0 To ارتفاع الصورة

P=A. GetPixel (**x,y**)

r=P.R

g=P.G

b=P.B

Avg=(r+b+g)/3

X(**x,y**). SetPixel (avg,avg,avg)

Next

Next

وبالرجوع الى كود تعديل الصورة الى GrayScale اخر خطوة هي لنسخ الصورة التي تم تغيير

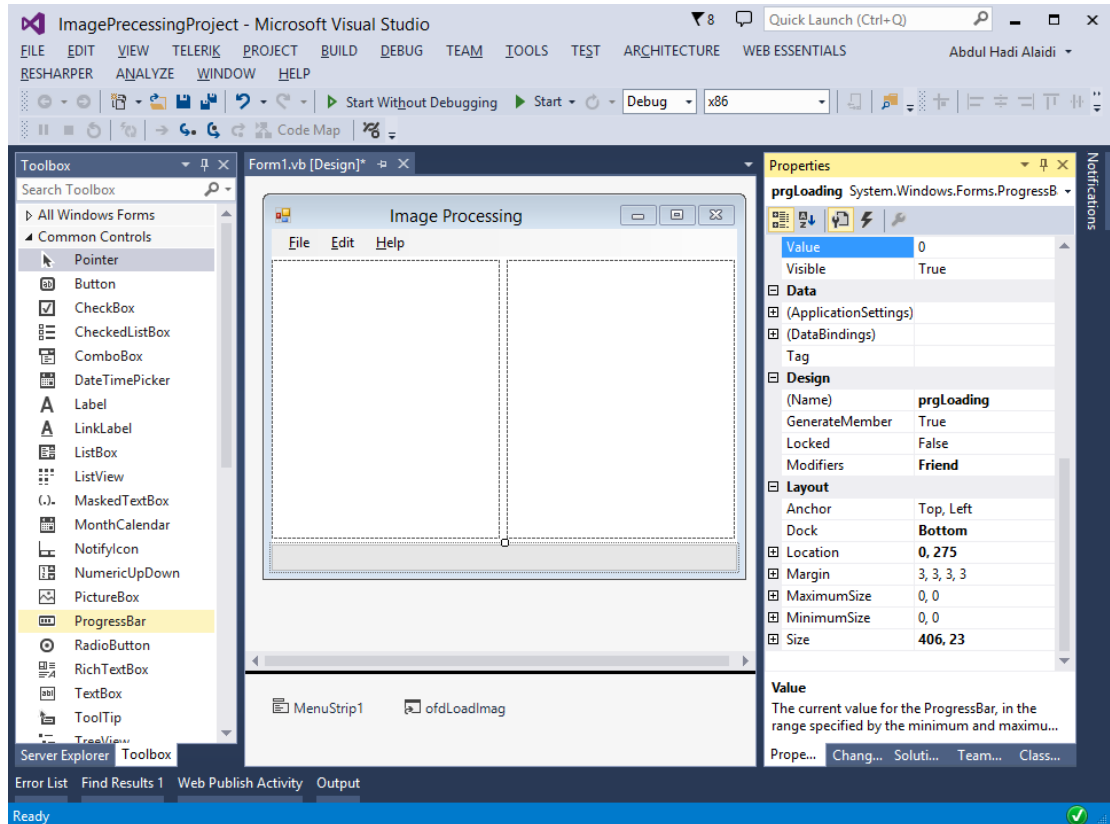
الوانها الى التدرج الرمادي الى pictureBox2 والتي تم تغيير اسمها الى picModified

picModified.Image = tempPicture

نضيف Progressbar الى اسفل الفورم ونغير الخصائص الى

Name: prgLoading

Dock: Bottom



يستخدم ال ProgressBar لظهار مقدار التقدم في معالجة ما نعدل الكود السابق ليصبح كالتالي

```
Dim width As Integer = picOriginal.Image.Width
Dim height As Integer = picOriginal.Image.Height
Dim tempPicture As New Bitmap(picOriginal.Image)
Dim p As Color
```

```
prgLoading.Value = 0
prgLoading.Maximum = width * height
```

```
For x = 0 To width - 1
    For y = 0 To height - 1
        p = tempPicture.GetPixel(x, y)
        Dim r As Integer = p.R
        Dim r As Integer = p.R
        Dim g As Integer = p.G
```



```

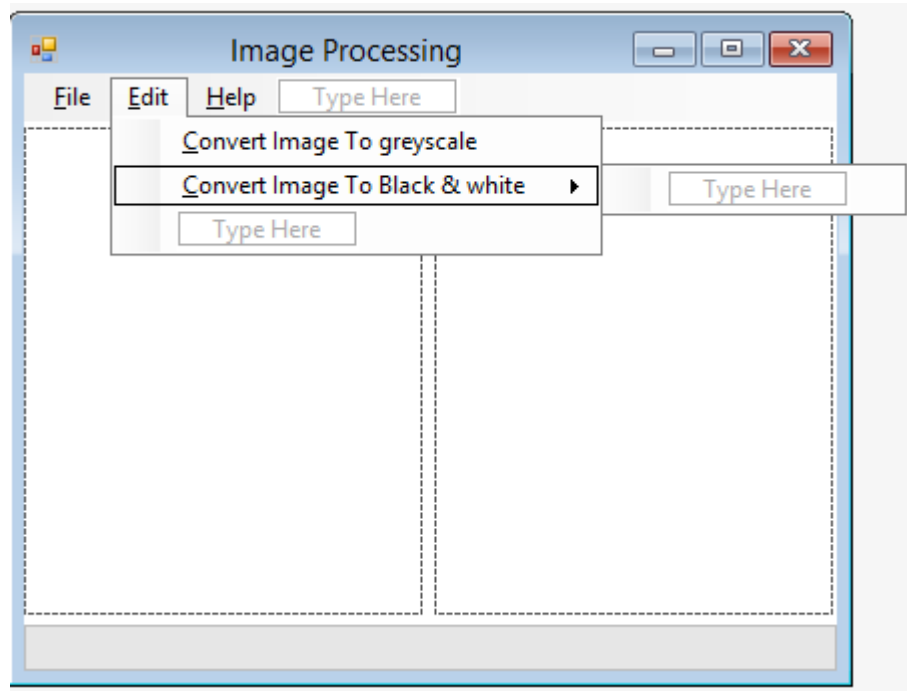
Dim b As Integer = p.B
Dim avg As Integer = (r + g + b) / 3
tempPicture.SetPixel(x, y, Color.FromArgb(p.A, avg, avg, avg))
prgLoading.Value = prgLoading.Value + 1
Next
Next
picModified.Image = tempPicture
prgLoading.Value = 0

```

تحويل الصورة الى اسود وابيض

نرجع الى التصميم ونضغط دبل كليك على

Convert Image To Black & white



نكتب الكود السابق مع تغير بسيط

```

Dim width As Integer = picOriginal.Image.Width
Dim height As Integer = picOriginal.Image.Height
Dim tempPicture As New Bitmap(picOriginal.Image)

```

```

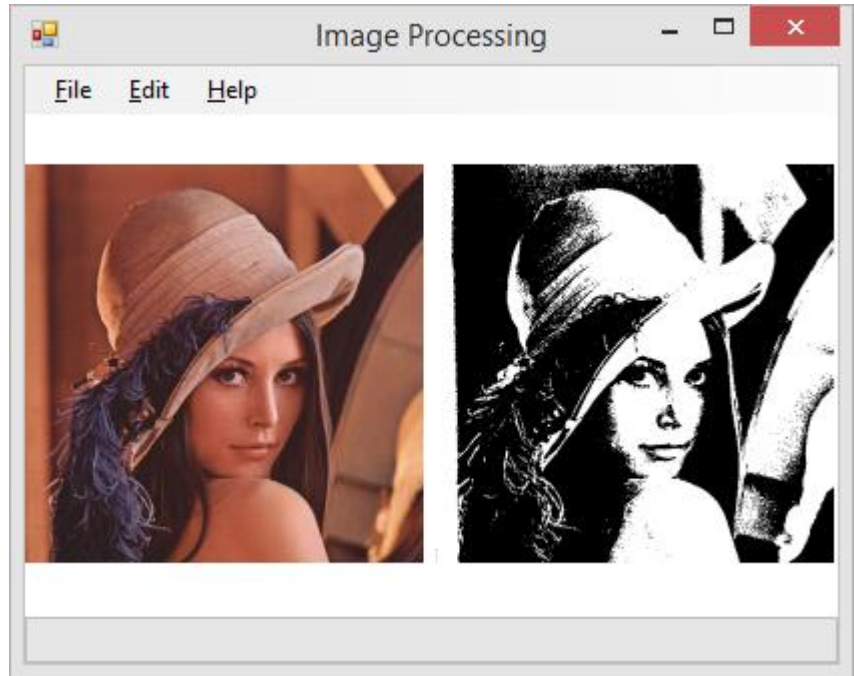
Dim p As Color

prgLoading.Value = 0
prgLoading.Maximum = width * height

For x = 0 To width - 1
    For y = 0 To height - 1
        p = tempPicture.GetPixel(x, y)
        Dim r As Integer = p.R
        Dim r As Integer = p.R
        Dim g As Integer = p.G
        Dim b As Integer = p.B
        Dim avg As Integer = (r + g + b) / 3
        If avg > 100 Then
            tempPicture.SetPixel(x, y, Color.White)
        Else
            tempPicture.SetPixel(x, y, Color.Black)
        End If

        prgLoading.Value = prgLoading.Value + 1
    Next
Next
picModified.Image = tempPicture
prgLoading.Value = 0

```



ان تغير الصورة الى ابيض واسود يعتمد على حد العتبة (في الكود السابق حد العتبة يساوي 100) حيث سيتم تغير الالوان الى اسود اذا كان المعدل اقل من حد العتبة والى ابيض في حالة كان المعدل اكبر من حد العتبة