البدايات المضيئة
في
Visual Basic
2012

اعداد : معاذ الزفتاوي

Email: tashfeen1985@gmail.com

ت : 201120031403+

مقدمة :

بعد الحمد لله والصلاة و السلام على رسول الله وبعد
فنمضي قدما لاستكمال هذه السلسلة المباركة (البدايات المضيئة في الفيجوال بيسك) وبعد الجزء الاول الذي يشمل مبادئ البرمجة و تعلم كيف تطور ادوات تحكم خاصة بك فانى اقدم هذا الكتاب لكي يجعلك تحسن شكل ادوات التحكم و تحسن شكل الواجهة الرسومية فيكون لها شكل جمالي يجذب زبائنك كما انك اذا فهمت محتويات هذا الكتاب سوف تجد نفسك قد وضعت على الطريق الذي يساعدك لفهم كيف تصمم الالعاب كما ستجد المفتاح الاساسي لفهم اهم المكتبات التي تصمم الالعاب ك **DIRECTX**

ما هي مكتبة GDi

هي مكتبة تم تطويرها ضمن فضاءات الاساماء التي في المكتبة العامة NetFramWark فهي تحتوي على العديد من التصنيفات و الادوات التي تساعدنا على الرسم على النموذج و ادوات التحكم كما سنشرح في الهذا الكتاب

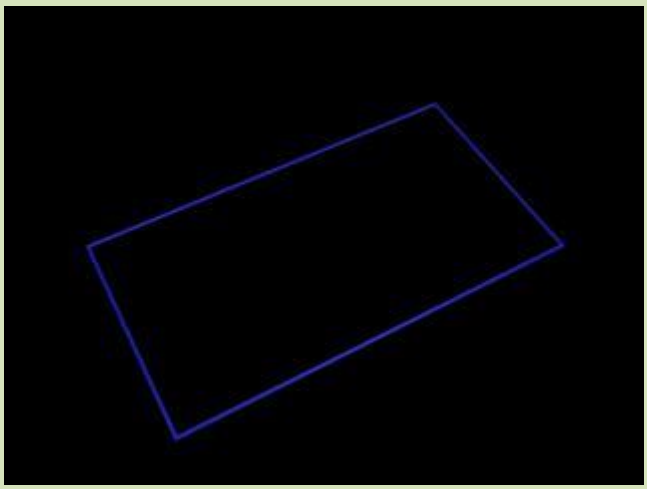
ففي هذا الكتاب سوف نتعمل

- مفاهيم اساسية
- ايجاد الاسطح للرسم
- رسم الاشكال
- انواع الفرشاة و القلم
- ادوات رسم اخرى .. التصنيف matrix و printDocument و تطبيق الرسم على الادوات التحكم
- تطبق عملي الجزء الاول و الذي سنشرح فيه كيف عمل برنامج رسام و تنسيق النصوص وتحريك الصور
- تطبق عملي الجزء الثاني وهو سوف نطبق بشكل متقدم على ادوات التحكم

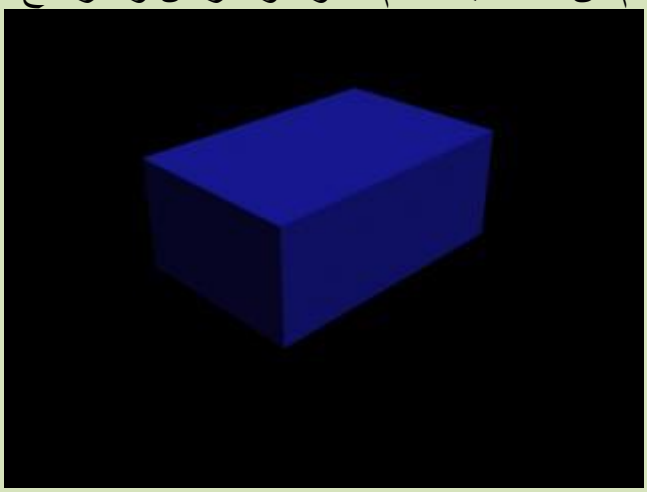
الفصل الاول
مقدمة في الجرافيك

ينقسم الجرافيك في البرمجة الى قسمين :
1- ثنائي لابعاد 2- ثلاثي الابعاد

الثنائي الابعاد هو انك ترسم الصور وتراها من اتجاهين فقط هما الطول و العرض



اما الثلاثي الابعاد هو الرسم من ثلاثة ابعاد هم الطول والعرض والارتفاع



قامت شركة مايكروسوفت ببرمجة مكتابتين في تصميم الجرافيك وهم

Graphics Device Interface GDI-1

وهي محور دراستنا في هذا الكتاب وهي مكتبة مصممة للرسم الثنائي الابعاد وهي مكتبة موجودة داخل الفيجوال استوديو ضمن مكتاب المكتبة الام netFramWork ويمكن استدعائها عن طريق الامر import كما هو موضح في الشكل التالي

```
Imports System.Drawing
Imports System.Drawing.Drawing2D
```

الاسطر السابقة تكتب في اعلى صفحة الكود قبل الاعلان عن فضاء اسماء والذي شرحناه في الجزء الاول من هذه السلسلة

وفي هذا السطر السابق كنت استدعيت فضاء الاسماء الذي يحتوي على اوامر الرسم و اوامر الرسم الثنائي الابعاد

كنا تحتوي المكتبة على تصنيفات اخرى فنعطي نبذة مختصرى عن كل تصنيف في الجدول الاتي

التصنيف	امر الاستدعاء	الاستخدام
Drawing	System.Drawing	هو الفئة الساسية وبها اوامر الرسم
Drawing2D	System.Drawing.Drawing2D	تقدم دوال متقدمة في الرسم
Imaging	System.Drawing.Imaging	تتعامل مع الصور
Printing	System.Drawing.Printing	تحول الرسم الى صور للطباعة
Text	System.Drawing.Text	التعامل بصورة رسومية مع الخطوط

التصنيفات في مكتبة GDI واستخداماتها

DIRECTX-2

هذه المكتبة مسؤولة علن الرسم الثنائي الابعاد و الثلاثي الابعاد لكي تتعامل معها يجب نتثيبها على الجهار كما ان المكتبة بها امكانية القدرة للتعامل مع الاجهزة الموصل للحاسب ككارت الصوت وغيهرا ولكن احب ان اخبرك عزيز القرئ انك لن تستطيع فهم هذة المكتبة الا اذا فهمت مكتبة GDI التي سنتناولها في الفصول القادمة بان الله لذا اطلب منك التركيز في قرائة الكتاب محاولة تطبيق التطبيقات عدة مرات لتتمرن على ممارسة المكتبة وستشعر بعد ذلك بمتعة العمل بهذه المكتبة .

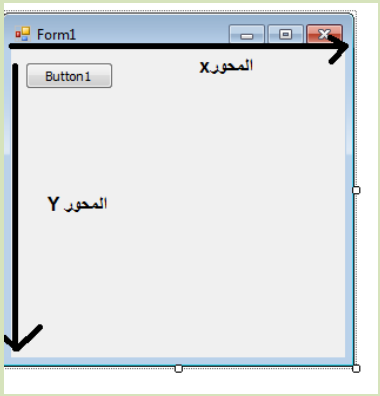
مصطلحات هامة

1- الطول و العرض

في الرسم الثنائي الابعاد انت لا تستخدم الا بعدين اما الطول و العرض او الارتفاع و العرض ففي لغة البرمجة انت تتعامل الكائنات بهذا المفهم فمثلا اذا اردت تغيير اطوال ال Button على الفورم او حتى الفورم نفسه تجد هناك خاصيتين تتعامل مع الطول و العرض وهما Width & Height ويجب ان تنعلم ان وحدة القياس هي البكسل pixl و التي سنتعامل معها في صفحات هذا الكتاب

2- الموقع

اذا اردت تحريك مكان الزر او مكان الفورم فانك تحتاج لتحديد النقطة التي يقع عليها الزر سواء كان داخل كل Formاو
Group Box او panal ولكي يتسطيع الفهم اكثر انظر للمثال الاتي

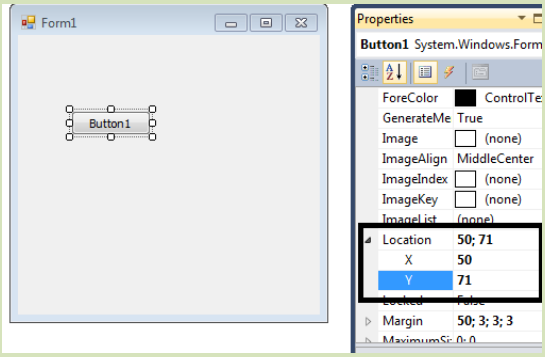


انت تريد تحريك الباتون على الفورم من يمين الفورم الى يساره لذا تحدد النقطة التي تقع على الفورم بحيث يكون النقطة الواقعة الركن اعلى يسار الفورم تكون (0,0) يعني محور السينات والصادات يبدأ من النقطة الموجودة اعلى يسار الفورم
ولكي نحدد مكان الباتون فاننا نحدد مكان النقطة التقع اعلى يسار الباتون فهي النقطة التي نستخدمها في تحديد مكان أي كائن في الفيجوال استوديو فنحن نحدد النقطة الواقعة اعلى يسار الباتون في المحور X يحدد الوضع الافقي للباتون و النقطة Y تحدد الموقع الراسي للباتون مثلا في الشكل السابق مكان الباتون على النقطة (12 , 12) ولكي نجعل مكان حيث اننا نحدد المحور السيني الاول ثم المحلور الصادي فاذا اردنا نغير مكان الباتون على الفروم باستخدام الاحداثيات فاننا نستخدم خاصية location و التي تحدد الموقع كما يمكن تحديد مكان الباتون عن طريق خاصيتي top , left وهي خاصيتي يسار و اعلى الباتون فالخاصية left تحدد مكان الباتون الافقي حيث ان قيمتها تزيد عن تحريك الباتون من اليسار لليمني وتقل عند التحريك من اليمين لليسار الى ان تصل للقيم صفر وهي قيمة يسار الفورم او الحاوية التي تحتوي الباتون فهي تمثل المحور X

كذلك نقول نفس الكلام في الخاصية top فهي تمثل المحور Y و تحدد المكان الراسي للباتون حيث تقل عن تحريك الباتون من اسفل لاعلى و تزيد اذا حراكناة من الاعلى لاسفل

مثال

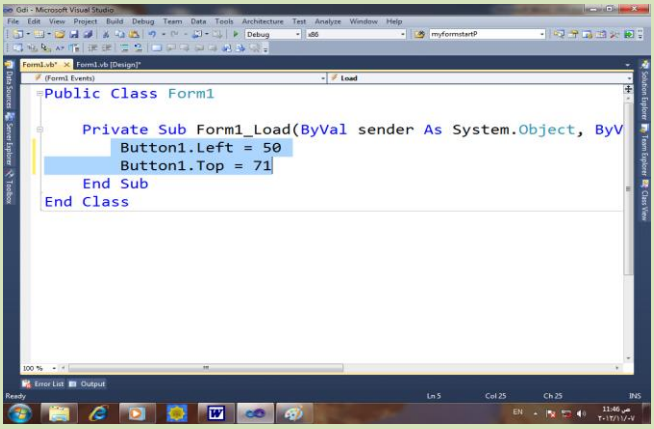
لنفترض اننا نريد تحريك الباتون من النقطة (12 , 12) الموجود بها الى النقطة (50,71) أي يكون قمية يسار الباتون تساوي 50 وقيمة اعلى الباتون تساوي 71 ولكي نقوم بذلك يمكن فعل ذلك اثناء وقت التصميم بتغيير قيمة الخاصية x الى 50 و الخاصية y الى 71 المدرجتان تحت خاصية locationكما هو موضح في الشكل الاتي



ويمكن تغييره عند باستخدام الكود فيتغير مكانه في وقت التشغيل عن طريق الكود الاتي المدرج في الاجراء Form_Load المفغل من حدث load فنكتب الكود

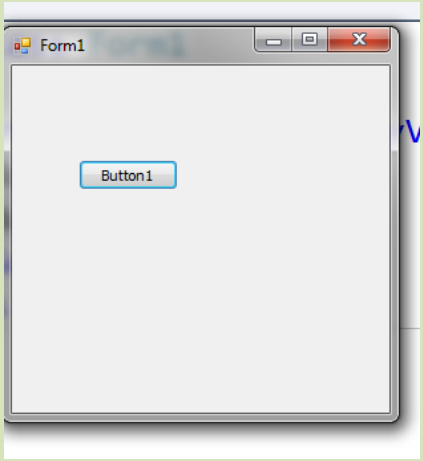
```
Button1.Left = 50
        Button1.Top = 71
```

فيكون شكل الكود كلاتي



في الكود السبق يوضح بيحث اننا عدلنا مكان الباتون عند تحميل الفورم فيتغير يساره للنقطة 50 ميسار الفورم

اعلاه من النقطة 71 من اعلى الفورم فيكون مكان الفورم كلاتي وقت التشغيل



كما يمكنك التاكد من ذلك عند تغير مكان الباتون على أي كان ثم تكتب الكود السابق عند حدث Click الخاص بالباتون

3- الالوان

هي العنصر الذي يعطي الجمال للرسم لذا الاون عنصر اساسي في علمية الرسم وسوف نتناولها بالتفصيل في فصول لاحقة لكي نحدد الالوان نحددها بطريقتين

1- دمج الالون با استخدام اللاوان الساسية GBR و هي تمثل الاوان الاخضر و الحممر و الزرق Green , Blue , Red ,

2- وهي استخدام الالون الموجودة مسبقا وهي تندرج تحت الكائن Color

ملخص الفصل الاول

1- تعرفنا على انواع الجرافيك
2- ماهي المكتبات المستخدمة في الدوت نت
3- كيف نتعامل مع الاطول
4- كيف نحدد موقع أي كائن
5- ما هي الالوان وكيف تحديدها

تمارين للقارئ

عليك تصميم برنامج صغير مهمته عندما تنقر بالماوس على الزر btn1 يتغير طوله الى 200 و عرضه الى 500 و مكان على الفروم في الاحداثيات (500, 400)

سؤال ما هي قيمة left , top

الفصل الثاني

الرسم في الدوت نت

في نهاية الفصل ستكون قادر على

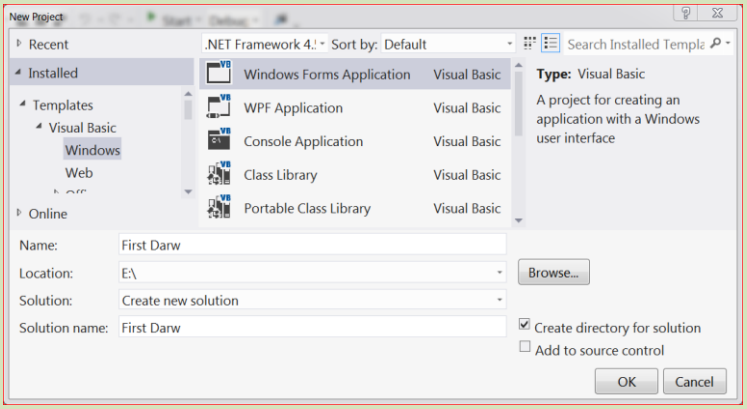
- تحديد الاسطح التي نستخدمها في عملية الرسم
- معرفة مكونات تصنيف Drowing
- بداية الرسم على الاستطح

قبل الشروع في كتابة الاكواد التي تخص الرسم اريدك ان تتخيل معي الفنان او المهندس الذي يريد الرسم ماذا يحتاج عند الرسم بالطبع يحتاج الى وجود ورقة او كشكول ورق ثم يحتاج الى اقلام و فرشاة رسم بمختلف انواعها ثم يحدد ماذا سوف يرسم .
في المثال السابق نحن نقوم كما يقوم الفنان فنحن نحدد ما هي الاسطح التي نريد الرسم بها ثم نجهز الاقلام و الفراشة ثم نبدا بالرسم فلو تعلمنا الرسم بهذه الطريقة فانك سوف تستطيع الرسم على الدوت نت بمكتبتيها ال Dirctxو GDI

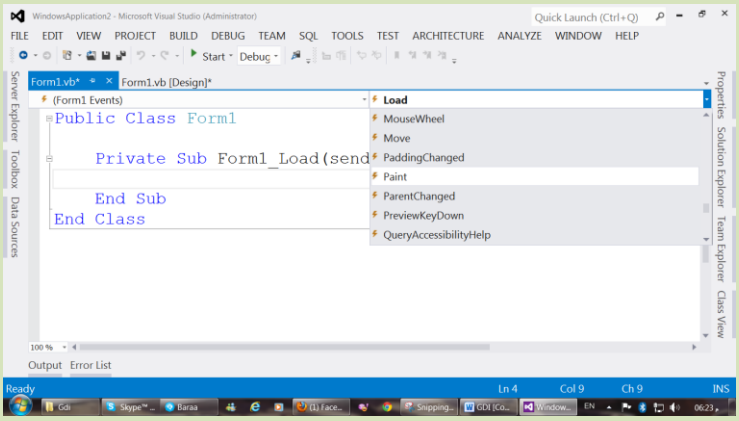
تحديد الاسطح

لنبدأ الان بتحديد السطح التي نريد الرسم عليها فانك اذا اردت الرسم فانك لن تجد معك لا اوراق او لوحات بل ستجد معك اسطح اخرى ك desktop, form , screen , button واخرى هذه الاسطح ,ولكن كيف نستطيع الرسم عليها

- افتح مشروع جديد الفيجوال بيسك وافتح تطبيقات الويندوز وسمي المشروع FirstDraw



- قم باستيراد المكتبة System.Drowing و System.Drawing.Drawing2D
- ثم اضغط على الفورم مرتين و اختر من قائمة الاحداث حدث Paint وهو الحدث المسئول عن تنفيذ اومر الرسم على الفورم



- ثم اكتب الكود الاتي

```
e.Graphics.DrawRectangle(Pens.Beige, 50, 50, 100, 100)
```

لا تحمل هم فهم الكود ولكن اريدك ان تفهم ان e.Graphic هي الاداة المسئولة عن رسم داخل الحدث paint

كما هذا المعامل e الذي يعتبر نوعه من الفئة paintEventArge تحمل خاصية اخرى وهي ClipRectangle وهي اخذ انواع المستطيلات من أي فئة اخرى كما سنوضح في الفصول القادمة

كما اريدك تعرف ان الحدث Paint هو المسئول عن الرسم على الفورم واي اداة اخرى ك button او TextBox وغيرها كنا نريد التنويه ان الحدث Paint يعمل فقط في وقت التحميل فقط أي يعمل مع الحدث load ولا يعمل في أي وقت اخر الا في اذا اعطيته في كل مرة تطلب منه الرسم الامر Refresh لكن تستطيع فهم ذلك اضف زر ثم قم باعلان عن متغير رقمي خاص كي يمكن قرائته من كل عناصر التصنيف

```
private i As Integer
```

داخل الفورم ثم عدل الكود في حدث paint ليكون كلاتي

```
If i = 1 Then

    e.Graphics.FillEllipse(Brushes.Orange, 50, 50, 50, 50)
ElseIf i = 2 Then
    e.Graphics.FillEllipse(Brushes.Red, 50, 50, 50, 50)

ElseIf i = 3 Then
    e.Graphics.FillEllipse(Brushes.Blue, 50, 50, 50, 50)

End If
```

تلاحظ انه نفس الكود السابق ولكن مع التعديل بحيث ان هذا الكود يطلب من الكمبيوتر رسم دائرة تحمل لون برتقالي ويكون مكانها على النقطة , x= 50النقطة y = 50 يكون الطول و العرض50 ايضا ولكن نلاحظ ان اللون يتغير في جملة الشرط بحيث انه اذا كان المتغير I= 1 يكون لون الدائرة برتقالي ويكون عندما يحمل الفورم load واذا كان المتغير يساوي 2 يكون لونه احمر يكون ذلك عن النقر على الزر واذا كان المتغيريساوي 3 يكون لونه ازرق وهو بعد النقر على الفورم.

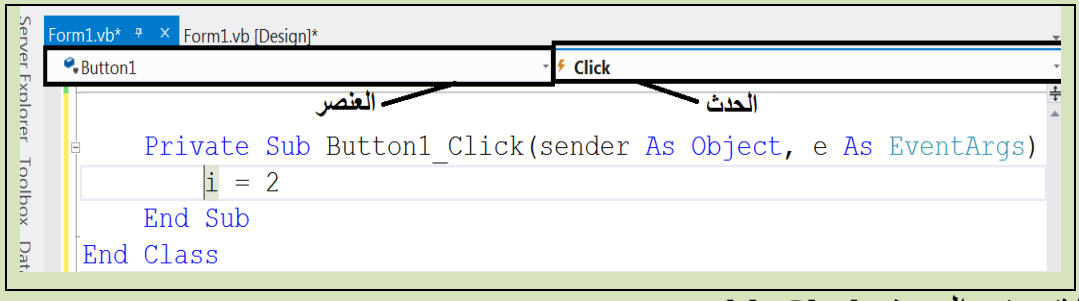
قم بكتابة الكود الاتي في الحدث load للفورم

```
Private Sub Form1_Load(sender As Object, e As EventArgs)
    i = 1
End Sub
```

واكتب الكود الاتي في الحدث Button1_Click

ولذلك باختيار العنصر button من القائمة اليسرى

واختيار الحدث من القائمة اليمنى كما هو موضح في الشكل الاتي



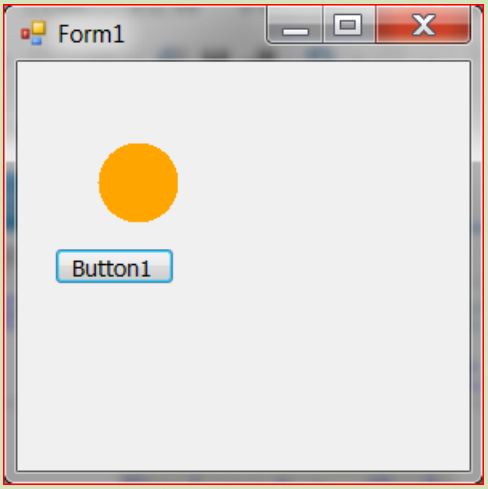
ثم اكتب الكود الاتي في الحدث Form1_DoubleClick

```
Private Sub Form1_Click(sender As Object, e As EventArgs)
    i = 3
End Sub
```

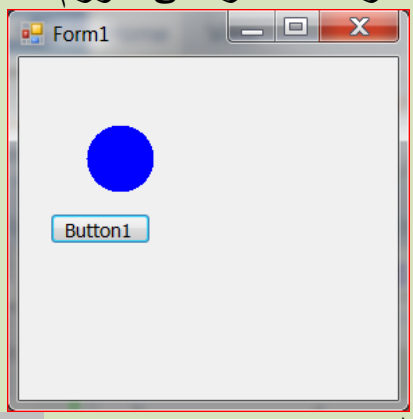
واذا قمت تشغيل البرنامج فانك تلاحظ انه لا يتغير لون الدائرة عند الضغط على الزر ولكي نجعل لون الدائرة يتغير يجب اضافة الكود الاتي داخل الحدث الذي يخث الضغط على الزر button1 ليكون الكود الخاص الضغط على الزر كلاتي

```
i = 2
Me.Refresh()
```

وبذلك يتم تغيير لون الدائرة اذا قمت بتشغيل البرنامج كما هو موضح في الشكل الاتي



و الشكل الاتي يوضح تغير لون الدائرة عند النقر على الفورم



يتغير لون النموذج ذلك اذا عدلت الكود بحيث اضفت الاجراء Refresh بعد الكود الخاص بتغيير المتغير وهو i=3

اريدك لا تحمل هم فهم اساليب الرسم لاننا سنشرح ذلك بالتفصيل في الفصول القادمة اريدك فقط معرفة كيفية عمل الاسطح حتى يسهل عليك الرسم

ولكن ليس سهل كل مرة تقوم بعمل refresh في كل مرة تريد تغيير الرسم لكي نجد حلا لهذه المشكلة نستخدم الخاصية handle الموجودة في في كل السطح كالفروم و ال button و غيرها ولكي نستطيع فهم ذلك بمسح كل الكود في المشروع ثم عند الحدث button_Click اكتب الكود الاتي

```
Dim gra As System.Drawing.Graphics = Graphics.FromHwnd(Me.Handle)
```

الكود السابق يقوم بانشاء متغير من نوع Graphic وهويقوم بمهام الرسم كما يحمل اجرائات ساكنة مهمتها تحديد الاسطح التي نريد الرسم عليها كالاجراء

```
Graphics.FromHwnd
```

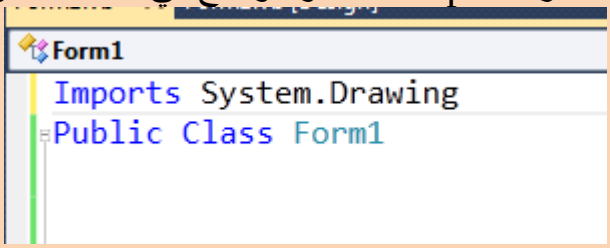
هذا الاجراء مهمته تحديد الاسطح التي نريد الرسم فيها مثل الفورم ولكن نضيف اليها الخاصية handle ويذلك يسهل علينا الرسم ثم نكتب الكود الاتي اسفل السطر السابق

```
gra.DrawRectangle(Pens.Azure, 1, 1, 20, 20)
```

حيث ان المتغير gra الذي اعلنا عنه سيقوم هو بمهام الرسم بدلا من e الموجودة في الحدث paint وبذلك نكون قد حددنا سطح للرسم عليه وتستطيع الرسم عليه من بعد بدون استخدام الحدث paint الخاص بكل اداه

ملحوظة

بدلا من عمل متغير كل مرة نستخدم فضائات الاسماء System.Drawing فاننا نقوم بعمل استيراد لها في اعلى كل تصنيف نستخدم فيه ادوات الجرافيك عن طريق الامر import كما هو موضع في الشكل الاتي



وبذلك يسهل علينا استخدام كل عناصر المكتبة

وبذكل قمنا بتحديد الاسطح التي نريد الرسم عليها ننتقل على الجزء الثاني من الفصل وهو مكونا التصنيف Drowing

فضاء الاسماء Drowing

سنقوم في بشكل سريع باستعراض التصنيفات الموجودة في فضاء الاسماء Drowing حيث انه هو التصنيف الاساسي المسئول عن كل التصنيفات الموجودة في المكتبة وسنقوم بشرح بعض التصنيفات منها في الفصول القادمة ولكن تتعرف بشكل مختصر على التصنيفات تابع مع الجدول الاتي

التصنيف	استخدامه
Bitmap	للتعامل مع الصور
Brush	فرشاة تلوين ولها عدة انواع سنتناولها في فصول قادمة
Color	مسؤول عن الالوان
Font	التصنيف مسؤول تحديد الخط الذي يقوم بكتابة النص

	المرسوم
Graphics	هو عصب المكتبة وهو المسئول عن الرسم وسنتعرف عليه في الفصل القادم
Icon	للتعامل مع الايقونات
Image	للتعامل مع الصور
ImageAnimator	للتعامل مع الصور المتحركة
Pen	قلم الرسم وله عدة انواع وسنتناوله في الفصول القادمة
Point	نقطة الرسم و التي تحدد الاحداثيات
Rectangle	مستطيل وسوف نتناوله هو النقطة في الفصل القادم
Size	مسؤول عن الاحجام
StringFormat	مسؤول عن كل مايتعلق برسم النصوص
Region	هو مسئول عن التعامل مع الاشكال وهو يعتبر كمقص يقص الكائنات على اساس الاشكال كما سنوضح لاحقا

وبعد ما تناولنا شكل التصنيف بشكل مختصر ننتقل للفصل القادم للبدئ بشكل عملي عملية الرسم

ملخص الفصل الثاني

1. تعرفنا على كيفيا تحددي الاسطح و رسمها وما هي الاحداث المناسبة لرسمها كما تعلمنا كيف نرسم من بعد
2. تغرفنا على تصنيف Drawing وما اهم تصنيفاته

الفصل الثالث

الرسم

في نهاية الفصل ستكون قادر على

- تحديد ادوات الرسم كالنقط و المستطيل و الاحجام
- الرسم باستخدام التصنيف Graphic

تصنيفات هامة في عملية الرسم

في بادئ الامر حتي تستطيع عمل تصميمات سريعة ومرنة باستخدام لغة الفيجوال بيسك او السي شارب المطورتان من مايكروسوفت يجب تتعلم بعد الفئات المدرجة تحت فضاء الاسماء Drowing وهي تسحتوي على عدة خصائص واساليب تساعد باقي العناصر و الفئات في عملية الرسم .

في الفقرة الاولي سنتعمل ثلاثة تصنيفات وهي

Point : تصنيف النقطة التي تحدد موقع الشكل الذي نرسمه و يتم تحديد النقطة بتحديد الاحداثيات سيني و صادي او X , Y

Rectangle : تصنيف المستطيل وهي مسئولة عن عمل مستطيل خاص من خلاله نحدد الطول و العر ض و الموقع من خلال الاحداثي السيني و الصادي

Size: تحدد من خلالها خاصيتي الطول و العرض

ولتوضيح ذلك نبدا من الاول تصنيف النقطة

Point

تصنيف النقطة هي تصنيف مسئولة عن تحديد احداثيات عل المحيط الذي تقع فيه سواء كان على فورم او جروب بوكس او panal او حتى على الشاشة Screen ويتم تحديد النقطة كما وضحنا سابقا بحيث يكون الركن العلوي الايسر يحدد نقطة البداية (0,0) بحيث يكون المحور xثم المحور y هكذا (x,y) ولتوضيح اكثر لنتعرف على كيفية انشاء هذة الفئة

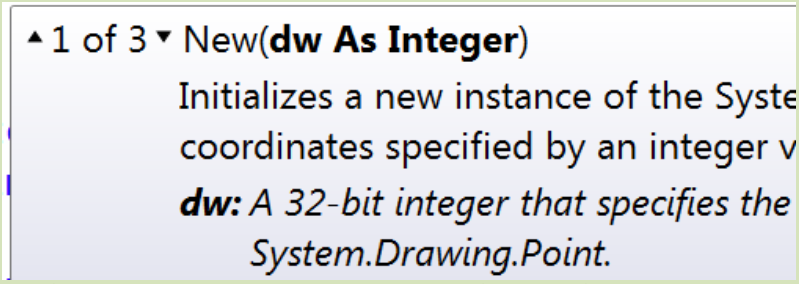
افتح مشروع جديد ثم استورد فضاء الاسماء drowing كما تعلمنا باستخدام الامر

```
Import system.drowing
```

من اعلى الكلاس ثم نعلن علن منتغير ينشء الكائن point

```
Dim p as new point()
```

سنجد اننا يجب ان نحدد معاملات عند الانشاء وعن فتح الاجراء new نجد انه يعطيك ثلاثة خيارات overload كما هو موضح في الشكل الاتي



فسوف نختار الختيار الذي يحدد x, y وهو الاجراء الذي سنتعامل معه بحيث يكون شكل الكود كالاتي

```
Dim p as new point(55,100)
```

بحيث ان المعامل الاول يحدد النقطة , Xو المعامل الثاني يحدد النقطة Y

واذا تعرفنا على الخصائص نجد ان الفئة تحتوي على خاصيتي مهمتين وهما خاصية X و خاصية Y والتان تحددان الاحداثيات

ويمكن توضيح ذلك في الجدول الاتي

الخاصية	الاستخدام
X	تحديد الاحداثي السيني
Y	تحديد الاحداثي الصادي
يتم الانشاء عند الاعلان عن متغير يحمل نوع نقطة كما في الكود الاتي Dim p as new point(55,100) حيث ان X تحدد الاحداثي السيني و Y تحدد الاحداثي الصادي نسبة الى اعلى يسار الحاوية التي تقع فيها النقطة بمعنى ان المحور X في النقطة الواقعة اعلى يسار الحاوية والتكن فورم تساوي 0 وكذلك المحور y يساوي 0	

سنشرح في ذلك امثلة كثير ولكن بعد الانتهاء من التصنيفين الاخرين

Size

هذا التصنيف يحتوي على خاصينين فقط وهما height و with لتحديد طول وعرض كائن ما و عند انشاء الكائن Size نستخدم في عملية الانشاء معاملين هما Widht و height بالترتيب كما هو موضح في الكود الاتي

```
Dim s As New Size(55, 12)
```

بحيث ان العرض يساوي 55 و الطول يساوي 12 وسوف نوضح ذلك في شرح الرسم

Rectangle

هذا التصنيف مهم جدا لانه يحتوي على خاصية نوعها Point و خاصية اخرى نوعها Size بحيث ان point او اسم الخاصية Location تحتوي على موضع النقطة التي تقع اعلى يسار المستطيل , والخاصية Size تحدد حجم المسطيل من العرض Width و hight

ويمكن تلخيص خصائص المستطيل على الحدول الاتي

الخاصية	الاستخدام
X	خاصية رقمية تحديد موضع النقطة على المحور X
Y	خاصية رقمية تحدد النقطة على المحور y
Location	خاصية من نوع Point تقوم بتحديد موضع النقطة من خلال المحورين الy, X
Size	خاصية من نوع Size و اللتان تحدد الطول وعرض المستطيل
Left	خاصية رقمية للقراءة فقط تحدد يسار المستطيل وتساوي الخاصية X
Top	خاصية رقمية للقراءة فقط تحدد اعلى المستطيل وتساوي الخاصية y
Right	خاصية رقمية للقراءة فقط تحدد يمين المستطيل وتساوي الخاصية Xمضاف اليه الخاصية Widthالتي تحدد عرض المستطيل
Bottom	خاصية رقمية للقراءة فقط تحدد اسفل المستطيل وتساوي الخاصية Yمضاف اليه الخاصية height التي تحدد طول المستطيل
Height	خاصية رقمية تحدد طول المستطيل
Width	خاصية رقمية تحدد عرض المستطيل

الخاصية المستخدمة للقراءة فقط هي لا نستطيع تغيير قيمتها

عملية الرسم

بعد ان قمنا بتعلم ما يساعدنا في علمية الرسم سوف نتعلم الان كيف نرسم وسوف نستخدم التصنيف Graphic الذي يعتبر هو عصب مكتبة ال GDI .

التصنيف Graphic

في هذا التصنيف سوف نتعلم اهم ما يستخدم في عملية الرسم اما باقي العناصر سوف نتعلمها في فصول قادمة باذن الله ان هم ما يميز هذا التصنيف وهي دوال الرسم كما سنرى

1- رسم الخط line

في رسم الخط المستقيم نطلب الدالة DrawLine ولتي بها اربعة خيارات بدوال OverLoad حيث يتاح لك استخدام اربع طرق في رسم الخط سنوضح اهمهما وهما

```
DrowLine(pen as pen , pt1 as point, pt2 as point  )
```

pen :

يطلب منك المعامل ان حدد لون القلم الذي تريد ان ترسم به الخط

Pt1:

يطلب ادخال نقطة البداية

Pt2

يطلب ادخال نقطة النهاية

مثال

في المثال يطلب من ان ترسم خط مستقيم تكون نقطة البداية اعلى يسار الفورم و ينتهي في منتصف الفورم

الحل

نفتح مشروع جديد ونكتب على الحدث Form_Click الكود الاتي

```
Dim p1 As New Point(0, 0)
Dim p2 As New Point(Me.Width / 2, Me.Height / 2)
Dim g As Graphics
g.DrawLine(Pens.Black, p1, p2)
```

شرح الكود

1- في السطر الاول قمنا بالاعلان عن متغير يقوم بانشاء كائن من نوع point هذا الكائن عند انشاءه نحدد المحورين X , y بحيث جعلنا x= 0 و y= 0

2- في السطر الثاني قمنا بالاعلان عن متغير يقوم بانشاء كائن من نوع point هذا الكائن عند انشاءه نحدد المحورين X , y بحيث جعلنا x= نصف عرض الفورم و y =نصف طول الفورم

3- اما في السطر الثالث قمنا بالاعلان عن متغير من نوع Graphic مع العلم ان التصنيف Graphic تصنيف مجرد أي لا يمكن انشاءه بالدالة new

4- في السطر الرابع قمنا باستدعاء الدالة Drawline من المتغير g الذي هو من نوع Graphic لرسم الخط فنمرر اليها المعاملات التي شرحناها من قبل

عند تشغيل البرنامج نجد ان البرنامج لم ينفذ الامر لاننا اعطينها امر الرسم بدون ان نعطيه السطح الذي يجب ان نرسم عليه لذا امامنا حلين اما ان نكتب الكود داخل الحدث paint او نكتب الكود الذي يجعلك ترسم من بعد و هو

```
g = Graphics.FromHwnd(Me.Handle)
```

ملحوظة هاااااااام

لا يمكن كتابة الكود الذي يحدد مكان سطح االرسم على الحدث Form_load لانه لن يسمح بالرسم و سيعطي استثناء

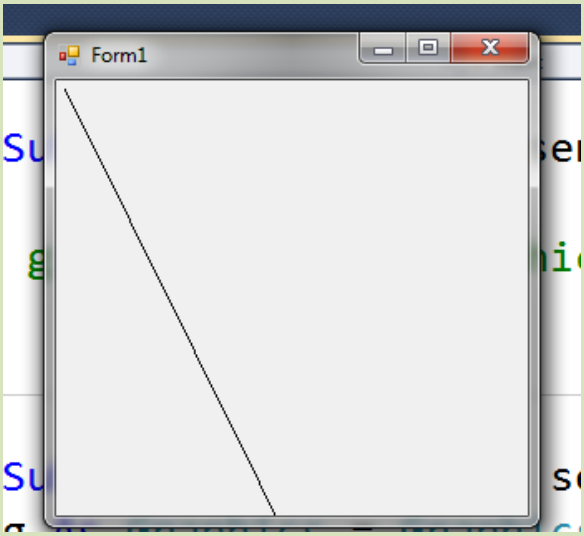
كما نسطيع رسم الخط عن بدون الاعلان عن المتغيرات التي تنشئ الكائن point وانما نستخدم الخيار الثاني و هو

```
Dim g As Graphics = Graphics.FromHwnd(Me.Handle)

g.DrawLine(Pens.Black, 5, 5, 10, 50)
```

بحيث اننا نعلن في السطر الاول عن المتغير الذي يفيد مكان سطح الرسم

اما السطر الثاني فنرسم عن طريق دالة الرسم ولكن عن طريق تمرير المحاور السيني والصادي لنقطة البداية و النهاية

وعند تشغيل البرنامج نجد ان الكمبيوتر رسم الخط كما هو موضح في الشكل الاتي



احب ان اوضح انك عندما تكتب الدالة قد تصادفك دالة رسم الخط تحدد لك معاملات من نوع pointF او معاملات تطلب منك تمرير المحور السني والصادي لكن المعاملات من نوع Single كما هو موضح في الشكل الاتي

```
2 of 2 DrawLine(pen As System.Drawing.Pen,
                x1 As Single,
                y1 As Single,
                x2 As Single,
                y2 As Single)
    Draws a line connecting the two points specified
```

او من نوع pointF

```
2 of 4 DrawLine(pen As System.Drawing.Pen,
                pt1 As System.Drawing.PointF,
                pt2 As System.Drawing.PointF)
    Draws a line connecting two System.Drawing.PointF
    pt1: System.Drawing.PointF structure that represents
```

هذان الخياران يوضحان انك تحدد المحاور السيني و الصادي ولكن مع وجود كسور وارقام عشرية وذلك عندما تريد ان تستخدم نظام قياس غير نظام الpixl البكسل قد يكون بالنظام المللي متر

وتستطيع تغيير نظام القياس من البكسل للمللي متر عن طريق الخاصية pageUnit المنبثقة من التصنيف Graphic كما هو موضح في الكود الاتي

```
Dim gr As Graphics
gr.PageUnit = GraphicsUnit.Millimeter
```

ستجد نفسك عند اعطاء قيم للخاصية تجده يعطيك اختيرات التي تعرف باسم Enum فيظهر لك عده خيارات فسوف استعرض لك بعضها

inch وحدة القياس بوصة
pixl وحدة القياس بكسل
*millimelt*وحدة القياس بالمللي
inch وحدة القياس بالنقطة
كما يوجد وحدات قياس اخرا

2- رسم المستطيلrectangle

ما رايك الان ؟؟؟ هل بدات الان تتعرف على طريقة الرسم عن طريق الفيجوال بيسك فهي طريقة سهلة و بسيطة لكنه تريد منك بعض التركيز
اذا كنت تريد رسم المستطيل سوف نكتب الدالة DrawRectangle ونجد انا تعطيك ثلاثة دوال overload واليد الصيغ الاتية لرسم المستطيل

الصيغة	المعاملات
DrawRectangle(*pen* as pen ,*Rec* as rectangle)	Pen تعني انت تستخدم قلم للرسم وفي هذه الحالة تظهر لك قائمة لاختيار الالوان الختيار اللون الذي نريد رسم به المستطيل Rec هو معامل نوعه مستطيل لاخذ اطوال المستطيل و موقعه
DrawRectangle(*pen* as pen ,*x* as integer ,*y* as integer , *with* as integer , *height* as integer	Pen شرحناه سابقا X لتحديد مكان المستطيل على المحور x Y لتحديد مكان المستطيل على المحور y Width تحديد عرض المستطيل Height تحديد طول المستطيل
DrawRectangle(*pen* as pen ,*x* as single ,*y* as single, *with* as single, *height* as single	نفس فكرة الدالة السابقة و لكن تعتمد على ان يكون وحدة القياس باللملي متر

مثال لنفترض اننا نريد رسم مستطيل على الفورم فقوم بكتابة الكود الاتي على الحدث Paing في الفروم

```
Dim size As New Size(200, 200)
Dim point As New Point(5, 5)
Dim rec As New Rectangle(point, size)
e.Graphics.DrawRectangle(Pens.Blue, rec)
```

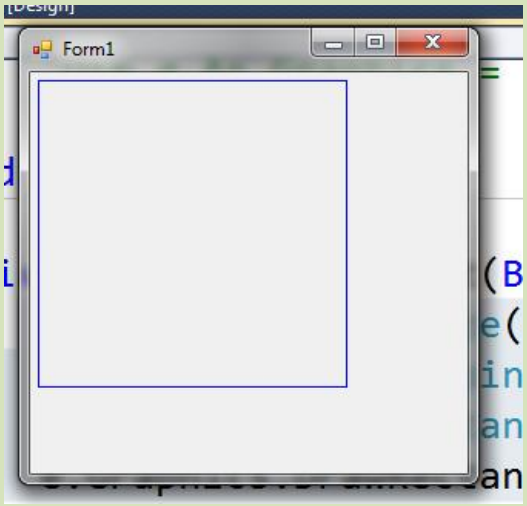
في الكود السابق قمنا بالاعلان عن متغير لانشاء كائن من التصنيف Size لتحديد حجم المستطيل و مررنا معاملات للعرض يساوي 200 و الطول يساوي 200

في السطر الثاني قمنا بالعلان عن متغير لانئاء كائن من نوع point لتحيد موقع المستطيل على الفورم

قي السطر الثالث قمنا بالاعلان عن متغير لانشاء مسطيل لرسمه على الفروم ثم اخذنا الموقع و الاحجام من المتغيرين الاخرين

في السطر الرابع قمنا بكتابة الدالة المسئولة عن رسم الستطيل فاستخدمنا الصيغة الاولى ومررنا اليها المتغير Rec

وعند تشغيل البرنامج تظهر لنا الفورم على الشكل الاتي

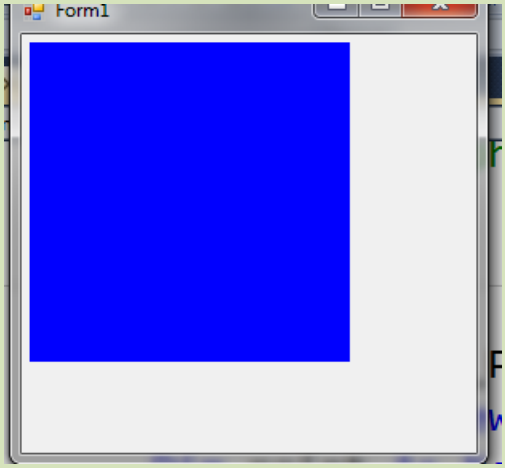


كما يوجد دالة FillRectangle تقوم برسم المستطيل مظلل و هي مثل دالة DrawRectangle ولكن مع اختلاف في معامل اختيار الالوان فهي من نوع brush يعني فرشاه وسنتعرف عليها و الفرق بينها وبين pen في الفصل القادم

ولان نستبدل كود الرسم بالدالة الاتية

```
e.Graphics.FillRectangle(Brushes.Blue, rec)
```

فتظهر النافذة على الشكل الاتي



3- رسم شكل بيضاوي Ellipse

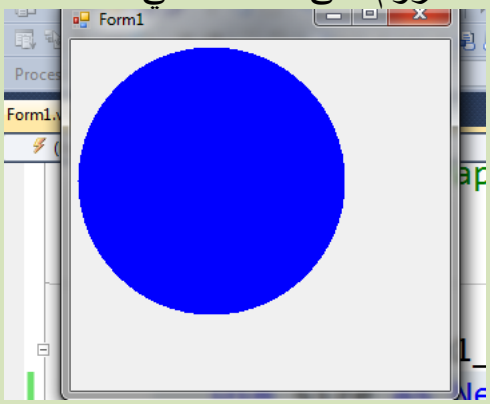
في عملية رسم الدائرة او الشكل البيضاوي فاننا نرسم الدالة DrawEllipse عند رسم بيضاوي مفرغ او FillEllipe في حالة رسم بيضاوي مظلل وعند الرسم يكون لنا ثلاث طرق لرسم الشكل البيضاوي ويمكن تلخصها كلاتي

الصيغة	المعاملات
DrawEllipse(*pen* as pen ,*Rec* as rectangle)	Pen تعني انت تستخدم قلم للرسم وفي هذه الحالة تظهر لك قائمة لاختيار الالوان الختيار اللون الذي نريد رسم به المستطيل Rec هو معامل نوعه مستطيل لاخذ اطوال المستطيل و موقعه
DrawEllipse (*pen* as pen ,*x* as integer ,*y* as integer , *with* as integer , *height* as integer	Pen شرحناه سابقا X لتحديد مكان المستطيل على المحور x Y لتحديد مكان المستطيل على المحور y Width تحديد عرض المستطيل Height تحديد طول المستطيل
DrawEllipse (*pen* as pen ,*x* as single ,*y* as single, *with* as single, *height* as single	نفس فكرة الدالة السابقة و لكن تعتمد على ان يكون وحدة القياس باللملي متر
DrawEllipse(*pen* as pen ,*Recs* as rectangleF)	نفس فكرة الدالة الاولي في ادخل المسطيل و لكن تعتمد على ان يكون وحدة القياس باللملي متر

مثال لنفترض اننا نريد رسم شكل بيضاوي على الفورم فقوم بكتابة الكود الاتي على الحدث Paing في الفروم

```
Dim size As New Size(200, 200)
Dim point As New Point(5, 5)
Dim rec As New Rectangle(point, size)
e.Graphics.FillEllipse(Brushes.Blue, rec)
```

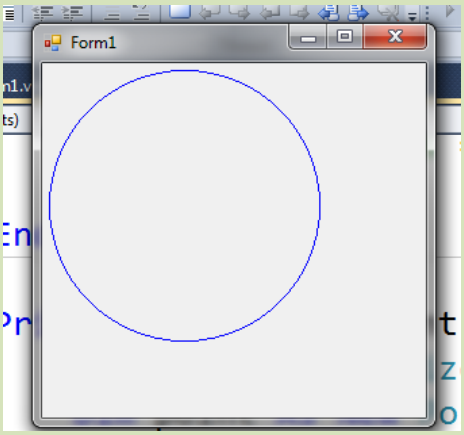
وعند تشغيل البرنامج تظهر لنا الفروم على الشكل الاتي



وعندما نريد رسم شكل بيضاوي مفرغ نستدل دالة الرسم بدالة DrawEllipse فتكون كلاتي

```
e.Graphics.DrawEllipse(Brushes.Blue, rec)
```

فيكون الشكل كلاتي



4- رسم القوس Arc

عند رسم قوس فانك ستحتاج الى ستخدام دالة واحدة فقط وهي DrawArc وهي دالة مسئول عن رسم القوس ولها عدة اختيارات تتشابه مع الدوال السابقة في اختيار اللاوان و تحديد المستطيل ولكن تختلفي معها في معاملين فقط وليك التوضيح

صيغة احدى دوال رسم القوس هي

DrawEllipse(*pen* as pen ,*Rec* as rectangle, *StartAngle* as single, *SweepAngle* as single)

نجد انها تتشابه مع دالة رسم المستطيل و دالة رسم بيضاوي و لكن تختلف معها في معاملين هما

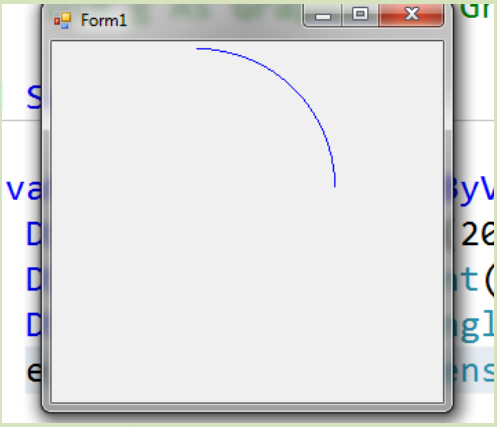
StartAngle : هي تعني الزاوية التي يبدا منها الرسم وعند حساب قياس الزاوية يبدا قياس الزاوية بحيث يكون الزاوية التي درجتها تساوي صفر تكون عبارة عن خط يوازي عرض الفروم او يكون عرض الستطيل الذي اخلناه في معامل Rec وعند قياس الزاية عليها نقيس من عرض المستقيم

SweepAngle هي تعني الزاوية التي ينتهي عندها الرسم ونبدا قياس الزاوية من بحين ان تكون قياس الزاوية من الزاية التي ادخلناها في المعامل *StartAngle*

_ **مثال** لنفترض اننا نريد رسم قوس على الفورم فقوم بكتابة الكود الاتي على الحدث Paing في الفروم

```
Dim size As New Size(200, 200)
Dim point As New Point(5, 5)
Dim rec As New Rectangle(point, size)
e.Graphics.DrawArc(Pens.Blue, rec, 0, -90)
```

عند التشغيل نجد الشكل الاتي



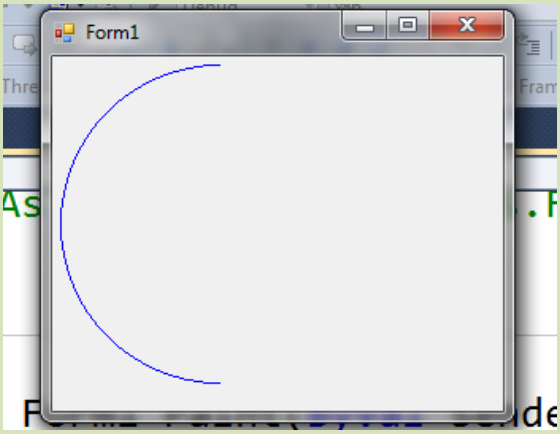
تلاحظ معي نقطة البداية هي درجتها صفر يعني عند نقطة التي تقع على اسفل يمين المستطيل الذي ادخلنا على المتغير Rec وعند قياس الزاية في المعامل *SweepAngle* قسنا الزاية على اساس الزاية من نقطة البداية *StartAngle*

للتوضيح اعدل المعالملات التي تخص الزايا الى يكون

StartAngle = -90

SweepAngle=-180

ثم جرب البرنامج فتلاحظ الاتي



ملحوظة: عند حساب الزاوية من حيث الموجب و السالب فان الزاية الموجبة تزيد عند ابتعاد النقطة عن اعلى او يسار فنجد وتقل وتصبح سالبة عند اقترابها من اعلى الو يسار الفورم

5- رسم منحنى Carve

عند رسم المنحنى فانك ستكتب الدالة DrawCarve وهي تعطيك سبعة خيارات للرسم وسوف نتناول منها واحدة فقط وهي تعطيك الصيغة الاتية

```
DrawCarve(pen as pen ,Point() as point)
```

كلمة pen نعلم انها القلم الذي يرسم المنحنى ولكن كلمة point() هي عبارة عن مصوفة تحمل جميع النقاط التي يمر عليها المنحنى

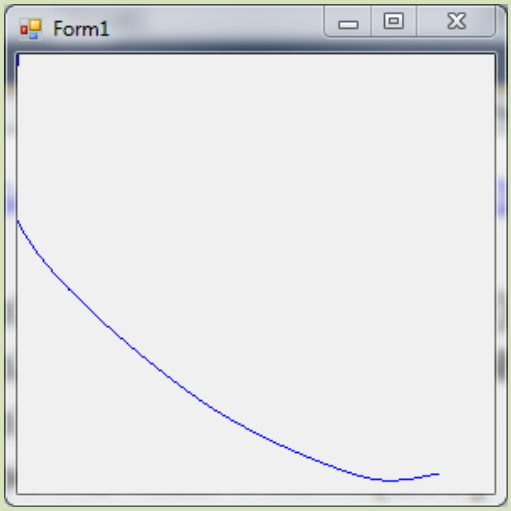
مثال لنفترض اننا نريد رسم منحنى على الفورم فقوم بكتابة الكود الاتي على الحدث Paint في الفروم

```
Dim p1 As New Point(0, 0)
Dim p2 As New Point(0, 100)
Dim p3 As New Point(100, 200)
Dim p4 As New Point(200, 250)
Dim p5 As New Point(250, 250)
Dim po() As Point = {p1, p2, p3, p4, p5}

e.Graphics.DrawCurve(Pens.Black, po)
```

في الكود السابق اعلنا عن 5 نقاط ثم اخلناها في مصفوفة من نوع Point نقطة بحيث يتم بذلك تحديد النقاط التي نريد ان نرسم عليها المنحنى فنحن عندما نريد رسم أي منحنا كنا نحدد المحور السيني و المحور الصادي عندما كنا في المدرسة فهي نفس الفكرة نحدد النقاط ثم نمرر عليها المنجنى

ثم نقوم بعمل دالة الرسم و التي نمرر اليها النقاط التي نريد الرسم عليها

وعند تشغيل البرنامج تظهر لنا الصورة الاتية



6- رسم شكل مضلع Polygon

عند رسم مضلع فانك ستكتب الدالة DrawCarve وهي تعطيك خيارين للرسم وسوف نتناول منها واحدة فقط وهي تعطيك الصيغة الاتية

```
DrawCarve(pen as pen ,Point() as point)
```

طبعا كلمة pen نعلم انها القلم الذي يرسم المضلع ولكن كلمة point() هي عبارة عن مصوفة تحمل جميع النقاط التي يمر عليها المضلع

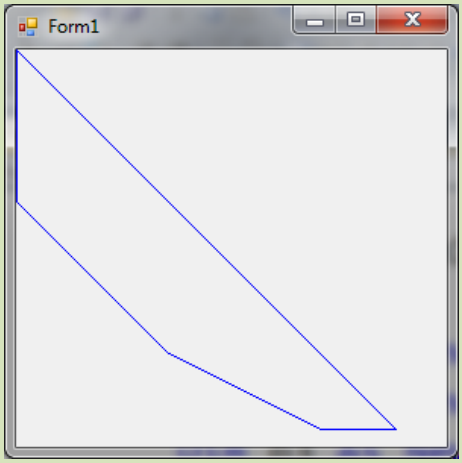
مثال لنفترض اننا نريد رسم مضلع على الفورم فقوم بكتابة الكود الاتي على الحدث Paing في الفروم

```
Dim p1 As New Point(0, 0)
Dim p2 As New Point(0, 100)
Dim p3 As New Point(100, 200)
Dim p4 As New Point(200, 250)
Dim p5 As New Point(250, 250)
Dim po() As Point = {p1, p2, p3, p4, p5}

e.Graphics.DrawPolygon(Pens.Black, po)
```

في الكود السابق اعلنا عن 5 نقاط ثم اخلناها في مصفوفة من نوع Point نقطة بحيث يتم بذلك تحديد النقاط التي نريد ان نرسم عليها المضلع فنحن عندما نريد رسم أي مضلع كنا نحدد المحور السيني و المحور الصادي عندما كنا في المدرسة فهي نفس الفكرة نحدد النقاط ثم نمرر عليها المضلع

ثم نقوم بعمل دالة الرسم و التي نمرر اليها النقاط التي نريد الرسم عليها

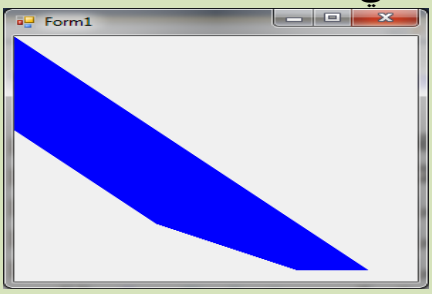
وعند تشغيل البرنامج تظهر لنا الصورة الاتية



كما توجد دالة ملئ المضلع التي تسمى FillPolygon ولكنها تختلف عن الدالة Draw في قلم الرسم فهو ليس قلب بل فرشاة Brush كما اوضحنا من قبل وعند كتابة الدالة Fill بهذة الطريقة

```
e.Graphics.FillPolygon(Brushes.Blue, po)
```

مكان Drae فيظهر لنا الشكل الاتي



7- رسم نصوص String

هنا سنتوقف قليلا لان هناك بعض المفاهميم يجب ان نفعرفها عندما تريد رسم النص
احب ان انوه ان رسم النص مهم كما هو رسم باقي العناصر لان رسم النص سيفيدك في كثير من الاستخدامات كرسم التقارير عمل رسم بياني فتريد توضيح البيان بشكل جمالي وغيرها من الاستخدامات التي ستجد نفسك تحتاج الى رسم النصوص
لذا سنتوقف لنتعرف على بعض التصنيفات التي تشارك في رسم النصوص
التصنيفان هما تصنيف الFont و String Format الذان يقومات التعامل مع شكل النص و طريقة كتابته

Font

هو تصنيف مسئول عن عملية بناء الخط ما هو نوعه وما هو النمط الذي نكتب به هل هو سميك ام مائل ام فوق خط

عند استدعاء هذا التصنيف يجب عمل علمية انشاء new وهي الدالة الوحيدة التي يمكن من خلالها عمل تعديل في شكل الخط لان كل الخصائص التي يحتويها هذا التصنيف خصائص للقراءة فقط Read only property يعني لا يمكن التعديل عليها بعد انشاء الكائن

وعند انشاء الكائن تجد هناك عدة دوال انشاء سنتناول بعض منها في الجدول الاتي

New (*Font Family* , *Size*)	هذه الدالة مسئولة في عملية الانشاء بحيث يجب ادخال اسم الخط في *Font Family* ادخال حجم الخط في المعامل*Size*
New (*Font Family* , *Size* , *FontStyle*)	هذه الدالة شبيهه بسا بسابقتها في المعاملات ولكن تزيد عنها في معامل *FontStyle* : هذا المعامل يعطيك عمدة خيارات تحديد نمط النص اليك بعضها *Bold* لجعل الخط سميك *italic* لجعل الخط مائل *underline* لجعل النص فوق الخط *Regular* لجعل الخط عادي *strikeout* لجعل النص ينتصفه خط
New (*Font Family* , *Size* , *FontStyle* , *GraphicsUnit*	هو نفس الدالة السابقة ولكن المعامل الزيادة هو يعطيك خيارارت تغيير وحدة القياس من البكسل او مللي متر وغيرها اليك الخيارات *inch* وحدة القياس بوصة *pixl* وحدة القياس بكسل *millimelt* وحدة القياس بالمللي *inch* وحدة القياس بالنقطة كما يوجد وحدات قياس اخرا وماذكرنا كان اهمها واكثرها ايستخداما

مع العلم ان خصائص هذا الكائن خصائص للقراءة كما وضحنا فهي تاخذ القيم التي اجخلناها في عملية الانشاء

StringFormat هو مسئول عن طريقة كتابة النص هل من اليمين لليسرا ام من اليسار لليمين ام من الوصط ام يكتب عمودي من اعلى لاسف وغيرها واليك اهم خصائص

سوف نشئ الكائن انشاء عادي new() ونستعرض اهم خصائصه في الجدول العادي

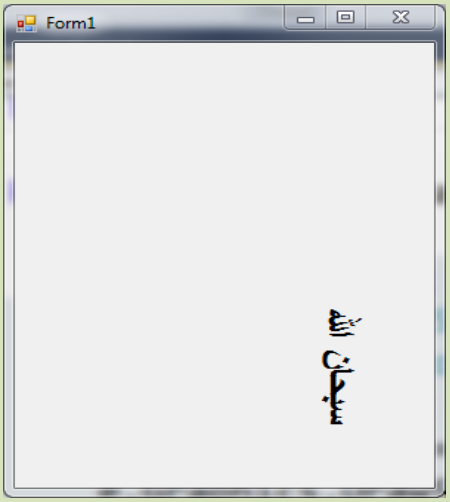
Alignment	تحدد مكان الخط في الميحد الذي يكتب فيه هل يكتب يمين المحيط ام يساره ام في الوسط
FormatFlags	تحدد طريقة كتابة الخط من اليمين لليسار ام من الاعلى للاسف كما هو موضح في الاختيارت الاتية *DirictionRightToleft* يكتب من اليمين لليسار *DirectionVertical* يكتب عمودي من الاعلى للاسف *DisplayFormatControl* يكتب من اليسار لليمين

بعد ما استعرضنا التصنيفين المسئولين عن رسم الخط يمكن ان نقوم برسم الخط بسهولة بعد ذلك فاليك هذا المثال

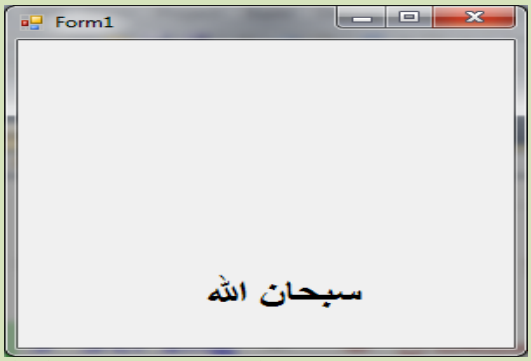
لنفترض اننا نريد رسم نص مكتوب عليه سبحان الله على الفورم فقوم بكتابة الكود الاتي على الحدث Paint في الفروم

```
Dim s As String = "سبحان الله "
Dim sf As New StringFormat()
Dim f1 As New Font("Times New Roman", 20, FontStyle.Bold)
sf.HotkeyPrefix = Drawing.Text.HotkeyPrefix.Hide
sf.FormatFlags = StringFormatFlags.DirectionVertical
e.Graphics.DrawString(s, f1, Brushes.Black, 200, 200, sf)
```

في السطر الاول قمن بالاعلان عن متغير نصي وكتبانا عليه كلمة سبحان الله
في السطر الثاني اعلنا عن متغير ينشئ نص من نوع StringFormat
في السطر الثالث اعلنا عن انشاء كائن الخط وادخلنا عليه المعاملات كما وضحنا سابقا
في السطر الرابع عدلنا على خصائص الStringFormat
في السطر الخامس كتبنا دالة رسم النصوص فادخلنا عليها معامل اسم النص و الخط و لونه وحددنا مكانه على الفورم و الدخلنا المتغير Sf المسئول عن الكالئن StringFormat
وبعد التشغيل تظهر الصورة كلاتي



تلاحظ ظهورها بالشكل العمودي واذا اردت على لشكل افقي عدل الخاصية FormatFlags الى StringFormatFlags.DirectionRightToLeft فتظهر الصورة كلاتي



سنشرح الان الاختيارات التي تستخدم في رسم النصوص ستجد لديك 6 اختيارات واهمها الاختيارات فيها هي

DrawString(*String*,*Font*, *Brusth*,*x*,*y*)	*String* : يحدد النص الذي نكتبه *Font* يحدد شكل الخط من اي متغير يحمل نوع تصنيف الخط *Brusth*: يحدد لون الخط لكن من النتغير يحمل نوع الفرشاه *المعاملان* X, y نوعهما Single وهم يحددان الاحدثيات السينية و الصادية وكل الخصائص التي تحدد موقع المستطيل في رسم النص من النوع single
DrawString(*String*,*Font*, *Brusth*,*rectangleF*	هي نفس المعاملات السابقة لكن الغير هو المعامل RectangleF الذي يضع نص في مربع بحيث لو كان النص طويل يمكن ان يكون

	هناك التفاف للنص فيكتب في عدة اسطر
DrawString(String,Font, Brusth , pointF	String : يحدد النص الذي نكتبه Font يحدد شكل الخط من اي متغير يحمل نوع تصنيف الخط Brusth: يحدد لون الخط لكن من النتغير يحمل نوع الفرشاه pointF : يحدد موقع النص ولكن كما قلنا ان الخاصيتين X, y نوعهما Single وكل خصائص النقطة و المستطيل في رسم النص من هذا النوع
DrawString(String,Font, Brusth ,x,y ,StringFormat)	ستجد نفس الاختيار الاول ولكن يوجد معامل StringFormat وهو يستقبل متغير من نوع StringFormat
DrawString(String,Font, Brusth ,rectangleF , StringFormat	ستجد نفس الاختيار الثاني ولكن يوجد معامل StringFormat وهو يستقبل متغير من نوع StringFormat
DrawString(String,Font, Brusth , pointF, StringFormat	ستجد نفس الاختيار الثاني ولكن يوجد معامل StringFormat وهو يستقبل متغير من نوع StringFormat

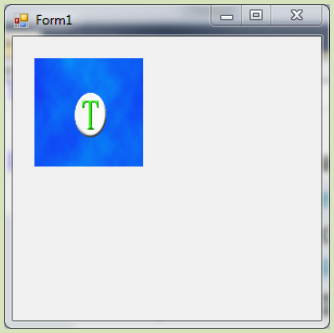
8- رسم صورة image

عند رسم الصورة ستجد الدالة سهلة جدا يمكنك فهمها بسهولة على الرغم انها تعطيك 30 اختيار ولكن اشرح واحدة منها وستجد الباقي متاشبهاة معها

سوف نقوم برسم صورة على الفورم فعليك كتابة الكود الاتي على الحدث paint

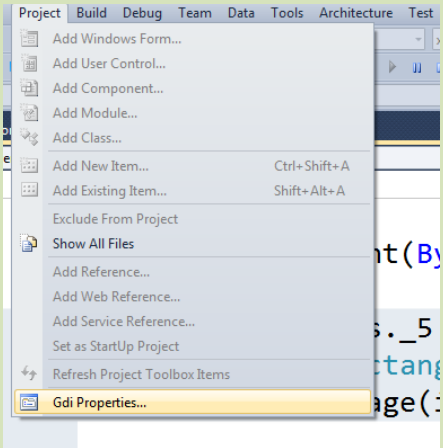
```
Dim img As Image
img = My.Resources._5
Dim rec As New Rectangle(20, 20, 100, 100)
e.Graphics.DrawImage(img, rec)
```

وعند تشغيل البرنامج ستظهر الصورة الاتية

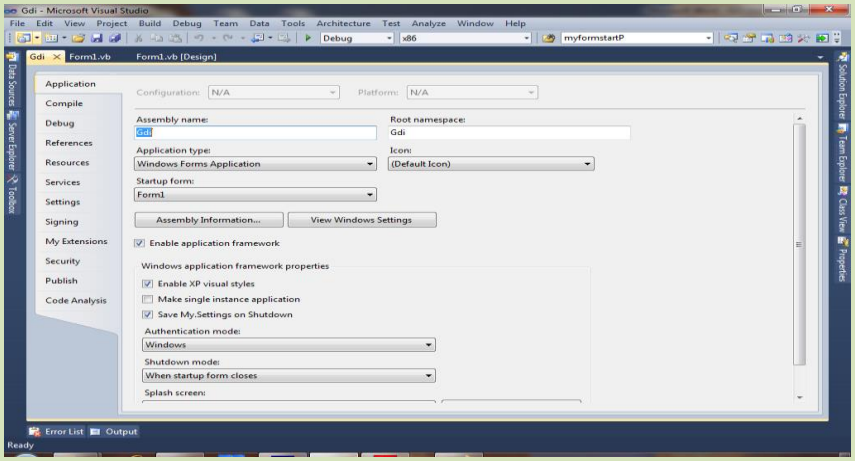


نحب ان نوضح من السطر الاول اننا اعلنا ان متغير نوع صورة
في السطر الثاني قمنا صورة من مكان ال resourec الخاص بالمشروع وسوف اعلمك كيف تدخل الصورة للملف بعد شرح الكود
في السطر الثالث اعلنا عن متغير لانشاء كائن المستطيل وحدد مكانه و طوله وعرضه وهو المستطيل الذي يمثل الاطار الذي يحمل الصورة
في السطر الرابع كتبنا الدالة التي تقوم برسم الصورة وكما قلت هي دالة سهلة جدا بها معاملين
Image وهي يستقبل الصورة
Rec وهو يحدد اطار الصورة

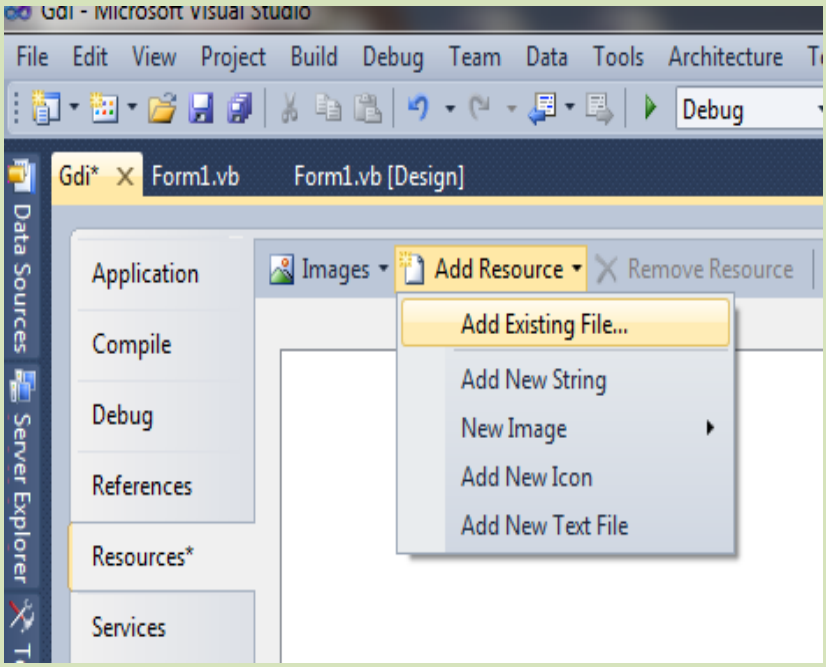
كنا في السطر الثاني استدعينا الصورة من ملف يسمى Resource ولكي تستطيع ادخال صورة للملف اتبع الخطوات الاتية
افتح قائمة project ثم اختر properties في مشروعي يسمى ;GDi properties كما هو موضح في الصورة الاتية



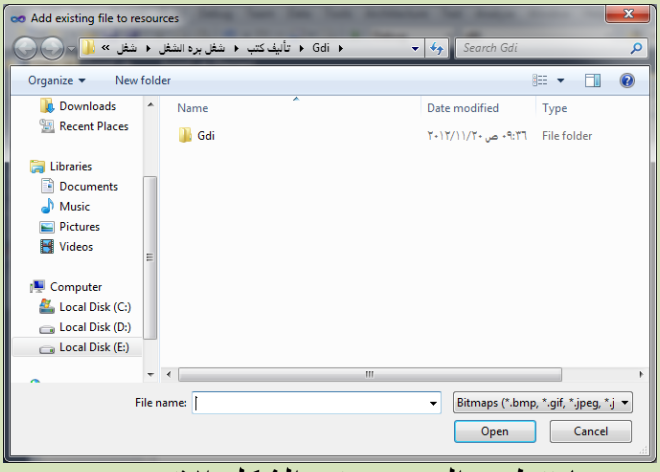
فتظهر لك الشاشة الاتية



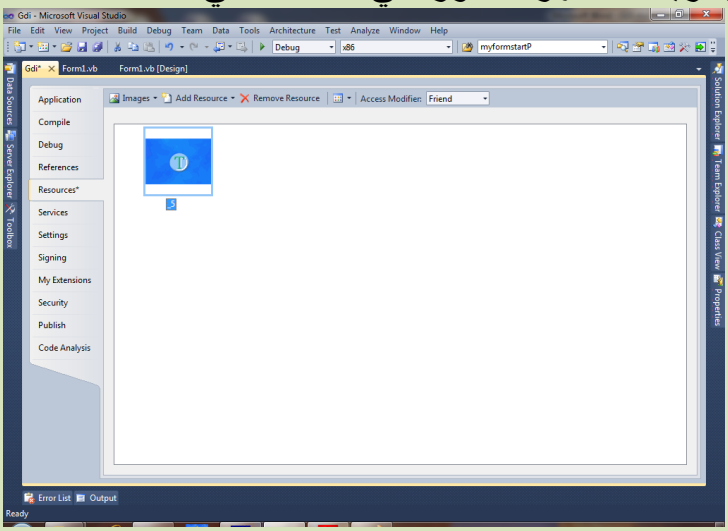
فنختار تبويب Resource فنختار منه Add Resource ثم نختار Add Existing File
ومنه نختار الصورة كما موضح في الصورة الاتية



فيظهر صندوق الحوار الخاص بالبحث عن الصورة



فنختار الصورة التي نريدها فتظهر الصورة في الشكل الاتي



هذا الشرح يكون في جميع اصدارات الفيجوال بيسك 2008 و 2010 و 2012

الى هنا احب ان اتركك تتعرف على باقي الرسمات بنفسك ولكني ساعطيك التلميح الناسب لفهم الرسم

9- رسم باي Pie

الباي او الذائرة الغير متكتملة تستخدم في الرسم البياني ودالة رسمها مشابها للدالة التي ترسم Arc

10- رسم منحى Bezier

بشبه بشكل كبير الدالة المسئولة عن رسم المنحنى

11- رسم منحنى مغلق Close Carve

بشبه بشكل كبير الدالة المسئولة عن رسم المنحنى او المضلع

والى هنا تعلمنا كيف نرسم و الان حان الوقت ان نتعلم كيف نلون الرسم بالفرشاة و القلم
انصحك ان تتدرب جيدا على هذا الفصل فهو اهم فصل في الكتاب لانك ستجد تشابه كبير في باقي الفصول وكل فصل مبنى على الفصل الذى هو قبله

ملخص الفصل الثالث

1- تعرفنا اعلى اهم الادوات المستخدمة في الرسم من تحديد النقط و المستطيل
2- تعرفنا على رسم الاشكل الاتي
- المستطيل
- البيضاوي
- النص
- الصورة
- المضلع
- المنحنى

تمارين الفصل الثاني و الثالث

صمم برنامج به زر اذا قمنا بتشغيل البرنامج يرسم مستطيل في النقطة (50, 50) بطول 100 و عرض 50 اذا نقرنا على الزر يتحول الى بيضاوي بنفس المكان و الاطوال و المستطيل و البيضاوي مضللين

تلميح

بعض الضغط على الزر يجب عليك مسح الفروم قبل الرسم عليها ذلك عن طريق الاجراء Clear الموجود في التصنيف Graphic

الفصل الرابع

الفرشاه و القلم

في نهاية الفصل ستكون قادر على

- ما هي انواع الفرشاة
- ما هو القلم وكيف يستخدم
- ما هي الالوان وكيفية التعامل معها

في الفصل السابق تعلمنا كيف نرسم ونستخدم ادوات الرسم ولان حان وقت تلوين الرسم عندنا اداتين للرسم اداة التلون التظليل وهي الBrush الفرشاه اما تلوين الاطار الخارجي في للقلم pen

ومن اجمل ما ستجده في هذه المكتبة GDi انك ستجد بها انواع كثير ستعجبك في التلوين سواء كانت الفرشاة ام القلم فسوف نتعرف سويا عن انواع الفرشاه

تنبيه هام جدا

--يجب ان تعرف جيدا ان الفرشاة Brush مقترنة اقتران كبير مع الدالة Fill في التصنيف Graphic و القلم pen مع الدالة Draw

انواع الفرشاه

في هذا الفصل سنتعرف بشكل كبير على فضاء الاسماء drowing2D . Drowing لذا من الافضل ان تقوم بعمل استيراد للفضاء عن مع استيراد فضاء الاسماء Drawing في الكود الاتي

```
Imports System.Drawing
Imports System.Drawing.Drawing2D
```

فهي تحمل جميع انواع الاقلام و الفرشاه

- Hatch brush

لو احببت ان تزخرف الاشكال التي ترسمه فسوف تقوم هذه الفرشاه بهذه المهم الليك طريقة استخدامها

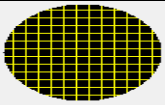
قم بالاعلان عن متغير من نوع hatchBrush وقم بانشاءه عن طريق الدالة new فتجد المعاملات الاتية

HatchStyle

هذا المعامل مسئول في توليد اشكال الزخارف

ForeCoor اللون الاول
BackColor اللون الثاني

اليك بعض الاشكال التي تولها الفرشاه يمكن تلخيصها في الجدول الاتي

	BackwardDiagonal
	Cross
	DarkDownwardDiagonal
	DarkHorizontal

واترك لك ان تتعرف على باقية الاشكال

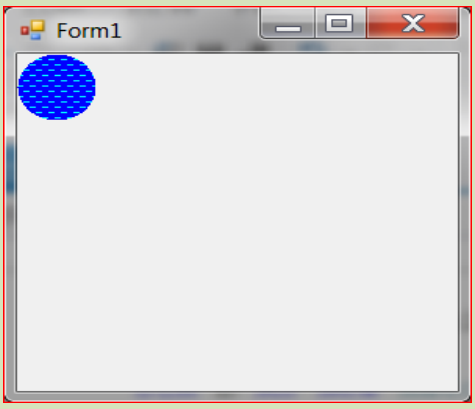
مثال

سنقوم برسم دائرة مزخرفة

قم بفتح مشروع جديد واكتب الكود الاتي في الحدث paint

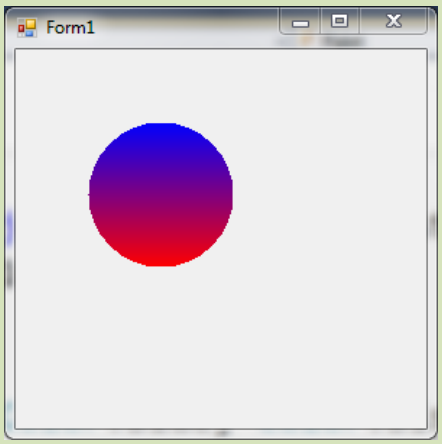
```
Dim b As New HatchBrush(HatchStyle.DashedHorizontal, Color.Aqua, Color.Bisque)
e.Graphics.FillEllipse(b, 0, 0, 50, 50)
```

في السطر الاول انشأنا الفرشاه واعطينا اللونين الاول و الثاني وحددنا النوع
في السطر الثاني قمنا برسم الدائرة
شغل البرنامج ستجد الاتي



- Linear Gradient Brush

الفرشاة الخطية التي تقوم برسم بلولنين متدرجين مثل هذه الصورة



- ولكي تجيد استخدام هذه الفرشاة الجميلة عليك ان تستخدم متغير يقوم بعمل انشاء للتصنيف Linear Gradient Brush

ولكي تقوم يبناء اذا الكائن يعرض عليك 8 خيارات كما هو موضح في الجدول الاتي

الدالة	الشرح
`Dim brush As New LinearGradientBrush(point, point2, color1, Colo2)`	ستجد ان هناك اربع معاملات *Point* نقطة البداية *point2* نقطة النهاية *color1* اللون الاول *Colo2* اللون الثاني
`Dim brush As New LinearGradientBrush(rectangle, color1, Colo2 , Angle )`	*Rectangle* مسطيل يحيط الشكل *color1* اللون الاول *Colo2* اللون الثاني *Angle* زاوية ميل تدرج الالوان
`Dim brush As New LinearGradientBrush(rectangle, color1, Colo2 , linearGradientMode)`	مثل المعامل في الدالة السابقة ولكن بدل الزاوي تجد معامل يحدد شكل الفرشاة

اما باقي الخيارات فهي تستخدمم عندما تكون وحدة القياس مللي متر

مثال جرب ان تقوم بعمل رسم مستطيل مظلل بالفرشاة وسنختار الخيار الاول ونطبقه كما يلي

افتح مشروع جديد واكتب هذا الكود عند الحدث paint للنموذج

```
Dim rec As New Rectangle(0, 0, 200, 200)
Dim p As New Point(200, 200)
Dim p2 As New Point(0, 0)

Dim brush As New LinearGradientBrush(p2, p, Color.Black, Color.White)
e.Graphics.FillRectangle(brush, rec)
```

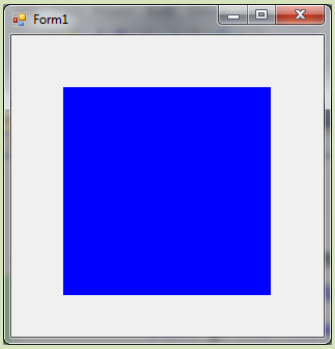
اظن انك بدات تعرف معظم الكود من النقاط و المستطيل وطريقة الرسم وعند تشغيل البرنامج ستجد هذه الصورة

- Solid Brush_

هذه الفرشاه هي التي كنا نستخدمها عندما كنا نتعرف على نستخدم دالة رسم الاشكال التي ترسم من الدالة Fill حين ان معامل الالوان كان عبارة عن هذه النوع من الفرشاه هي فرشاة اللون الواحد وعند انشاء الفرشاه يكون المعامل لون واحد فقط كما هو موضح في الكود الاتي

```
Dim brush As New SolidBrush(Color.Blue)
```

ويكون الشكل كلاتي



- Texture Brush

هذة الفرشاة تستخدم في تلوين الشكل بالصورة أي يتسترد صورة وتظلل بها البرنامج
وعند انشاء هذه الدالة ستجد عندك 8 خيارات وسنستعرض بعضها وستجدها سهلة جدة

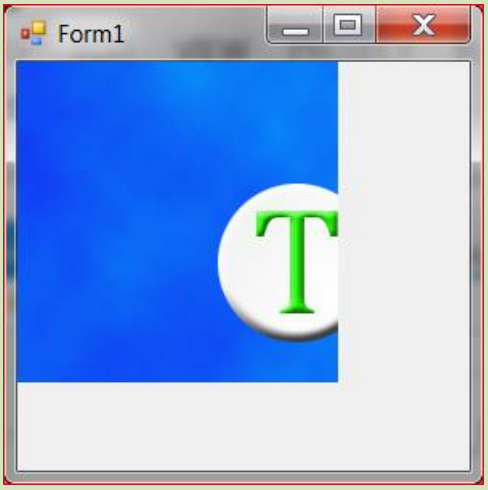
المعامل هو يطلب ادخال صورة	`Dim brush As New TextureBrush(image)`
Warpmode يحدد هل الصورة في شكل عادي ام مقلوب	`Dim brush As New TextureBrush(image, Warpmode)`
Rectangle يقوم تحدد حجم الصورة ثم يقسمها على المربع	`Dim brush As New TextureBrush(image, rectangle)`
نفس ما سبق ولكنه يوزع الصورة على الشكل و يقوم بتحدد حجمها	`Dim brush As New TextureBrush(image, rectangle, Warpmode)`

مثال

افتح مشروع جديد و اكتب فيه الكود الاتي

```
Dim b As New TextureBrush(My.Resources._5)
e.Graphics.FillRectangle(b, 0, 0, 200, 200)
```

في السطر الاول انشاءنا متغير يحمل نوع الفرشاه واعطيناه الصورة وبطعا شرحنا كيف يتم ادخال الصورة في resource
في السطر الثاني يشرح رسم دائرة تحمل الفرشاه
افتح البرنامج وستجد الاتي



- Path Gradient Brush

هذه الفرشاه هي من اجمل ما اعطيه الدوت نت لانها تحتوي على الاشكال التي تدعمها الدوت نت وكي تتعرف على هذه الدالة عليك معرف تصنيفين هامين لرسم الفرشاه

GraphicsPath وهو تصنيف يقوم بطلب اضافة الاشكال التي تدخل في شكل الفرشاه عن

طريق الاجراء Add ستجد مثل الدالة التي نسميه Fill ولكنها تكتب على كلى FillRectange كذلك نسمي هذه الدالة Add وكناها AddRectangle و AddEllipes وهكذا

PathGrideentBrush هذا التصنيف هو المسئول عن وضع شكل للفرشاه ويحتيوي على خاصيتين مهمتين

SurroundColors وهو يأخذ مصفوفة من الالوان
CenterColor وياخد اللون الذي يقع في وسط الشكل

ولكي نستطيع فهم ما يحدث قم بعمل مشروع جديد وضغ الكود في الحدث Paint للنموذج

```
Dim c As Color = Color.Blue
Dim cco As Color = Color.BurlyWood
Dim sco() As Color = {c}
Dim rec As New Rectangle(0, 0, 100, 100)
Dim path As New GraphicsPath()
path.AddRectangle(rec)
Dim brush As New PathGradientBrush(path)

brush.SurroundColors = sco
brush.CenterColor = cco
e.Graphics.FillRectangle(brush, 0, 0, 200, 200)
```

متغيرات لاخذ الالوان

متغيرات لتحديد الشكل الذي نريده

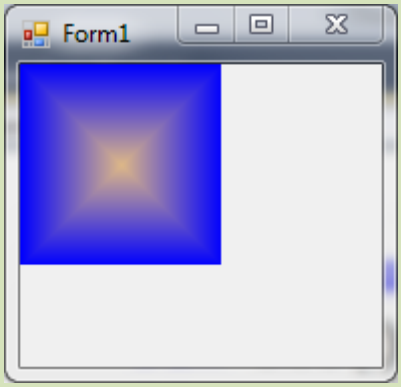
لاعداد الفرشاة

الرسم

من السطر الاول للسطر الثالث اعلانا عن متغيرات تاخذ الالوان التي نريد بها تلوين الشكل كما وضعنا مصفوفة الوان حتى ستساعدنا لاضفة قيمة للون المحيطي ***SurroundColors*** في التصنيف **PathGradientBrush**

في السطر الرابع اعلنا عن المستطيل الذي ساخده المتغير الذي سينشئ **GraphicsPath**
في السطر الخامس اعلنا عن المتغير الذي ينشئ التصنيف **GraphicsPath**
في السطر السادس افقنا المستطيل عن طريق الدالة **AddRectangle** و التي نضيف اليها المعامل الخاص باذافية المستطيلات
في السطر السابع انشانا متغير يحمل التصنيف **PathGradientBrush** ومن دالة الانشاء New ستجد عدة خيارات خذ افضلها وهو ***path*** الدي ياخذ المتغير path
في السطر الثامن و التاسع اضفنا الى خصائص الالوان مع العلم ان خاصية ***SurroundColors*** عي مصفوفة اللوان
في السطر الاخير اعطينا امر الرسم

عند تشغيل البرنامج نجد هذا الشكل



وبذلك نكون انتهينام من الفراشاة التي تظلل الاشكال

انواع القلم

القلم هو المسئول عن التلوين يحيث انه يرسم محيط الشكل وهو مرتبط مع الدالة Drow كDrawRectangle ولكي ننشء القلم سنجد ان دالة النشاء تعطيك اربعه خيارات وهي

`Brush` هو يحدد الفرشاة ويمكنك عمل فرشاة من الانواع التي شرحناها `Width` سمك الخط	`Dim pen As New Pen(brush, Width)`
`Color` اللون التي نرسم به	`Dim pen As New Pen(Color, Width)`
كما ترى اللون بدون تحديد السمك	`Dim pen As New Pen(brush)`
هنا الفرشاه فقط	`Dim pen As New Pen(color)`

ستجد في الخاصة Pen عدة خصائص رائعة ستساعدك في الرسم واليك اهمها

Brush	فرشاة
Color	لون
Dash Style	شكل الخط منقط ام متصل
Dash Cap	نهاية كل نقطة
Start Cap	شكل بداية الخط هل دائري ام سهم ام مشطوف
End Cap	شكل نهاية الخط

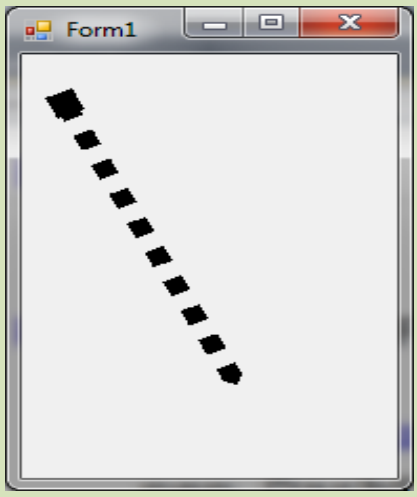
Width	السمك

مثال

اكتب هذا الكود في الحدث Paint في الفورم

```
Dim pen As New Pen(Brushes.Aqua, 10)
pen.DashStyle = DashStyle.Dot
pen.StartCap = LineCap.SquareAnchor
pen.EndCap = LineCap.Triangle
e.Graphics.DrawLine(pen, 20, 30, 100, 200)
```

في السطر الاول علنا عن متغير ينشئ كائن من نوع قلم ومررا اليه اللون والسمك
في السطر الثاني اعطينه شكل انقاط الخط
في السطر الثالث اعيطناه شكل بداية الخط
في السطر الرابع اعطيناه شكل نهاية الخط
في السطر الخامس رسمنا الخط
وعند تشغيل البرنامج سنجد الصورة الاتية



وبذلك نكون قد انتهينا من شرح انواع الاقلام و الفرشاة عليك بطبيق كل ما شرحته مرة اخرى لتزيد من كفائتك سننتقل الان الى التعامل مع الالوان

الألوان

في هذا الجزء سنتعرف على اللالوان بحيث يجب علينا اختيرا اللون المناسب ليظهر الشكل جميل لذا طورت الدوت نت ثلاثة طرق لاختيار الالوان

- **Qbcolor**

هذه الطرثة معروفة في لغة الفيجوال بيسك 6 وهي اختيار 16 لون اساسي وهي غير شائعة الاستخدام

- **Default**

هي الالوان المستخدمة في النوع Color الذي استخدماه طوال عملنا

- **Rgb**

هي الدالة التي تعتمد على ثلاثة الوان اساسية و التي ندمجها في بعضها و الاوان هي red الاحمر و Green الاخضر و blue الازرق فيتم الدمج بين الالوان عن طريق دالة دخل التصنيف **Color** تسمى **FromArgb** وهي تعلتمد على اربعة معاملات وهي

A = لقياس مدى شفافية الشكل

R = يحدد درجة اللون الاحمر

G = يحدد درجة اللون الاخضر

B = يحدد درجة اللون الازق

ولكي نتعامل مع اللون بشكل واضح اكتب الكود الاتي في الحدث paint للفورم

```
Dim c As New Color
c = Color.FromArgb(255, 0, 50)
Dim p As New Pen(c)
e.Graphics.DrawRectangle(Pens.PaleGoldenrod, 50, 50, 100, 100)
```

وبهذه الطريقة قمنا بعتعديل اللون بالشكل الذي نريده

وبنهاية هذا الفصل استطيع ان اؤكد لك انك استطعت ان تحيط باهم الادوات التى نريدها فى الرسم فى الفصول القادمة سوف نعطيق تطبيقات عملية على الرسم فى الواقع سوف نتعرف على بعض الادوات التى تجعل منك مبرمج محترف فى الرسم باللغة الفيجوال بيسك لان الرسم سيفيدك فيما بعد فى اضافة شكل جمالى للواجهات الرسومية و عمل الرسم البيانى و طباعة التقارير لذا عليك ان تتدرب على ما سبق بكثافة ثم انتقل الى الفصل القادم الذى سيعطيك المزيد من الادوات التى تؤهل للعمل فى مجال الرسم

ملخص الفصل الثالث

- تعرفانا على اهم طرق لاستخدام القلم و انواعه وشكل نهاية و بداية الخط
- انواع الفرشاة و التي تتلخص في
  - Hatch brush
  - Linear Gradient Brush
  - _Solid Brush
  - Texture Brush
  - Path Gradient Brush
- الالوان وكيفية تغييرها وما هي الدوال التي تنشئها

تمارين الفصل الرابع

صمم برنامج به زر اذا قمنا بتشغيل البرنامج يرسم مستطيل في النقطة (50, 50) بطول 100 و عرض 50 اذا نقرنا على الزر يتحول الى بيضاوي بنفس المكان و الاطوال و المستطيل و البيضاوي مضللين بالفرشاه التي تحضر الصورة في الشكل المستطيل و فرشاة تدريج الالوان في البيضاوي

الفصل الخامس

ادوات مهمة

في الفصول السابقة تعلمنا كيف نستطيع رسم الاشكال ووضع الصور على الاسطح كالفورم و أي اداة و لان سنقوم بتناول ادوات تجعل منك مبرمج محترف في هذا المجال وستطيع ان تفتح لنفسك مجال اكبر في سوق العمل لانك اذا تدربت على ادوات الرسم و تدربت على الادوات التي سنشرحها فاني اؤكد لك انك ستصبح محترف في مجال صناعة الادوات وتطوير شكل واجهة الرسمات للمستخدم وغيرها من الاستخدامات

في نهاية الفصل ستكون قادر على

- ما هو التصنيف ماتريكس
- ما هي الاداة print Document
- كيف نغير شكل ادوات التحكم User Control

Matrix

هو تصنيف يقوم بادخال بعض التعديلات في الرسم بحيث يكسبه مرونة اكثر ويسهل على المبرمجين كتابة اكواد كثيرة لكي تصل للذي تريده .

تعلمنا في الفصول السابقة اننا اذا اردانا رسم أي شكل فاننا نعتمد على تحديد مكانه عن طريق تحديد موقعه بالنسبة للنقطة التي تقع اعلى يسار الفورم فلو انك تريد رسم شكل بياني او رسم تغيير المحاور بحيث تكون اسفل يمين النموذج للقيام باستخدامات اخري
تخيل انك تعمل في تحريك الصور و الرسمات وتريد تدوير الصورة او تغيير الطوال المستطيل كل ما ذكرناة من قبل لا يحل الا باستخدام الاداة ماتريكس ولكي نستخدم هذه الاداة فعلينا معرفة مكانها على المكتبة GDi هي موجودة دخال فضاء الاسماء Drawing2D لذا يجب عليك ان تستورده كما تعلمنا في الفصل السابق

```
Imports System.Drawing
Imports System.Drawing.Drawing2D
```

ثم نقوم بعمل متغير يقوم بانشاء الكائن من نوع matrix ولكي تقوم بانشاء الكائن سوف يعطيك عدة خيارات فافضل حاليا ان تنشئه بدون اضافة أي معاملات هكذا

```
Dim m As New Matrix
```

ستجد التصنيف مليء بدوال و اجراءات اكثر من خصائص وسوف نتناول اهمها في السطور القادمة تابع معي

- **رسم شكل اكبر من حجمة او اصغر من حجمه** _ *بالاجراء Scale*

لقي هذه الفقرة سنتناول كيف تستخدم الاداة Matrix مع الاجراءِ Scale ولعمل ذلك عليك اتباع الخطوات الاتية

سوف نرسم المتسطيل بحجم اكبر من حجمه 5 مرات فاتبع الخطوات الاتية

1- عرف المستطيل

```
Dim rec As New Rectangle(0, 0, 50, 50)
```

2- عرف ماتريكس

```
Dim m As New Matrix
```

3- استخدم الاجراء Scale

```
m.Scale(5, 5)
```

حيث ان المعاملات التي تتمثل في 5 و 5 عبارة عن

Scalex فهي تعدل الطول من الاتجاة X

Scaley فهي تعدل الطول من الاتجاة y

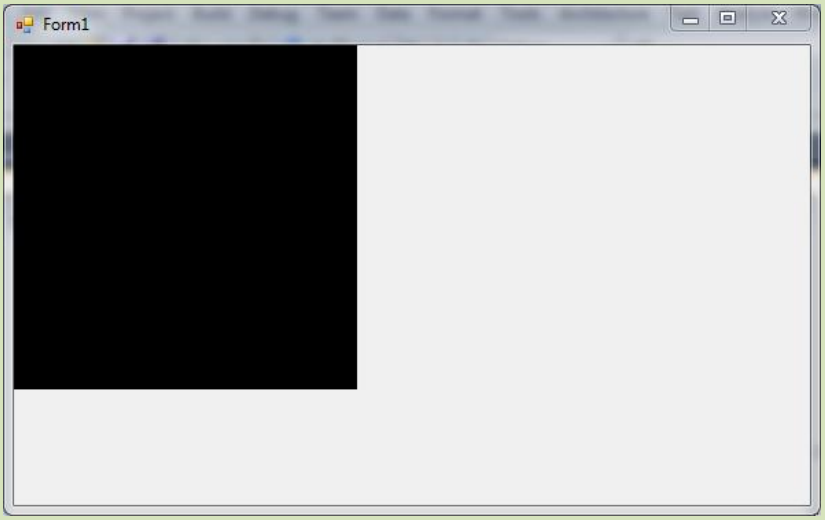
4- نقوم بتمريره للتصنيف Graphic حيث ان التصنيف Graphic سوف يقوم بعمل تغيير في اطوال المستطيل

```
e.Graphics.Transform = m
```

5- سوف نقوم برسم المستطيل

```
e.Graphics.FillRectangle(Brushes.Black, rec)
```

الى هنا لو كمنا برسم المستطيل سنجد ان طولة اكبر من الطول المطلوب بخمس مرات كما في الشكل الاتي



وكي نتاكد من الذلك ما رايك باكمال باقي الخطوات

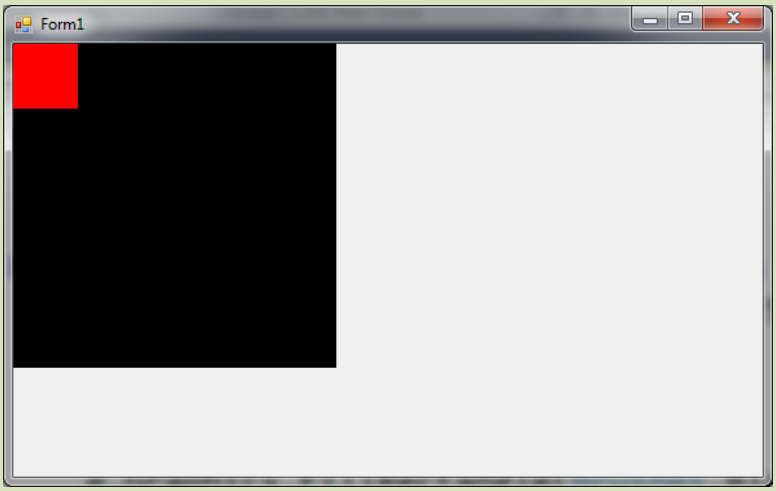
6- قم باعادة الطول الى الطول الطبيعي

```
e.Graphics.ResetTransform()
```

7- قم برسم المستطيل مرة اخرى

```
e.Graphics.FillRectangle(Brushes.Red, rec)
```

قم ستشغيل البرنامج مرة اخرى ولاحظ اننا رسمنا المستطيل الكبير باللون الاسود و المسطيل بطوله العادي باللون الاحمر

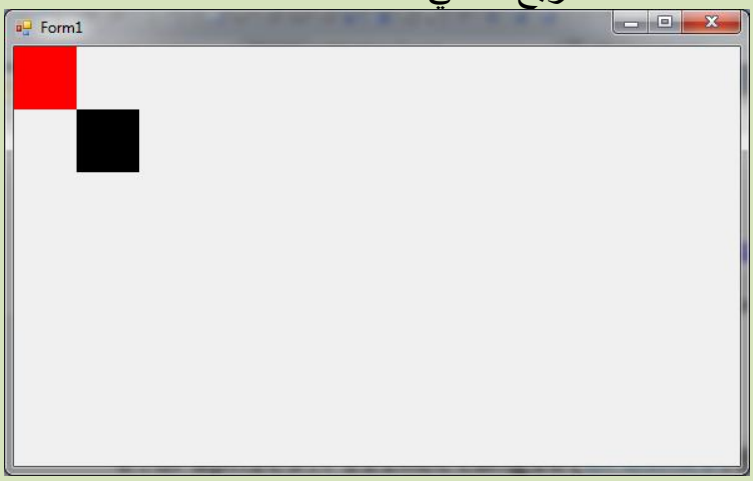


- **تغيير مكان نقطة البداية (0 و 0) من اعلى يسار النموذج للمكان الذي نريده بالاجراء Translate**

ما رايك لو اتبعنا الخطوات السابقة ولكن في الخطوة الثالثة نغير الدالة الى الدالة Trsnlate وليك الكود بالشكل المعدل

```
Dim rec As New Rectangle(0, 0, 50, 50)
Dim m As New Matrix
m.Translate(50, 50)
e.Graphics.Transform = m
e.Graphics.FillRectangle(Brushes.Black, rec)
e.Graphics.ResetTransform()
e.Graphics.FillRectangle(Brushes.Red, rec)
```

واذا شغلنا البرنامج نلاحظ شكل المربع كلاتي

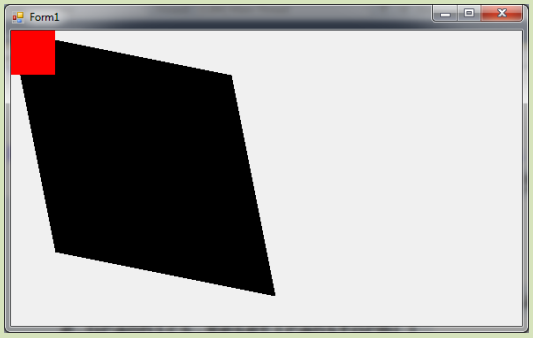


- **تغيير شكل المستطيل من خلال دالة Shear**

سوف نستخدم نفس الكود ولكن تعديل الخطوة الثالثة كما في المثال السابق كما في الكود الاتي

```
Dim rec As New Rectangle(0, 0, 50, 50)
Dim m As New Matrix
m.Shear(5, 5)
e.Graphics.Transform = m
e.Graphics.FillRectangle(Brushes.Black, rec)
e.Graphics.ResetTransform()

e.Graphics.FillRectangle(Brushes.Red, rec)
```

وعند تشغيل البرنامج ستجد الشكل كلاتي



- **عمل دوران للشكل عن طريق الدالة Rotate**

لكي يكون المثال واضح سنعدل المستطيل المتغير في المثل السابق وندخل صورة كما شرحنا في الفصل الثاني ونقوم بتدوير الصورة ولكن سوف تلاحظ ان الصورة يجب ان تدور على حول محور و النقطة المعروفة للدالة Rotate ان المحور هو النقطة (0,0) ولكي تغير المحور سوف نستخدم الدالة RotateAt لان بها معامل من نوع نقطة يحدد محور الدوران
ولكي نستفيذ اكثر تابع معي المثال الاتي

مثال في هذا المثال سوف نقوم بادخل صورة لننظر كيف يتم تدويرها بدل من المستطيل فاكتب الكود الاتي في الحدث Paint

```
1 Dim rec As New Rectangle(100, 100, 100, 100)
2 Dim img As Image = My.Resources._5
3 Dim p As New PointF(rec.X + rec.Width / 2, rec.Y + rec.Height / 2)
4 Dim m As New Matrix
5 m.RotateAt(30, p)
6 e.Graphics.Transform = m
7e.Graphics.DrawImage(img, rec)
8 e.Graphics.ResetTransform()
```

في السطر الا ول علنا عن مستطيل وحدد اطواله
في السطر الثاني اعلنا عن متغير من نوع صورة وادخلنا عليه الصورة

في السطر الثالث ادخلنا النقطة المحورية التي تمثل محور الارتكاز التي تدور حولها الصورة وقلنا ان يتم وجودها في منتصف السمستطيل

في عملية تحديد النقطة وقفة كثير ما يقع فيها المبرمجين حيث يكون هنا تلخبط في تحديد موقع المكان الذي الرسم فيه

لقد حددنا ان الذي يحدد الموقع هي الخواص X , y , left ,top هي التي تتحكم في الموقع وليس الخواصWidth , hight فهما يحددان الحجم

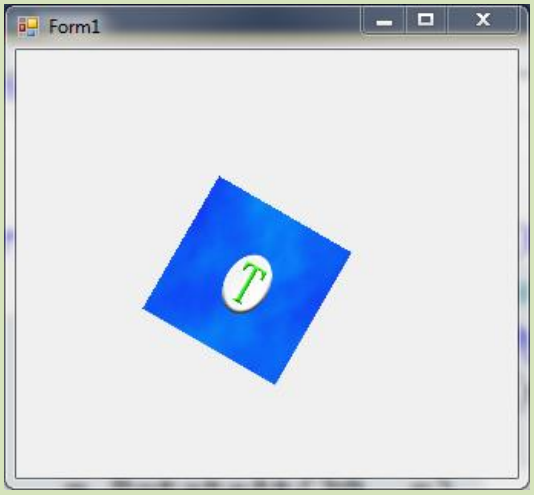
ملحوظة هامة جدا

في تحديد موقع النقطة قلنا ان مكان النقطة على المحور X يسكون مكان النقطة الرئيسة في منتصف المستطيل وهي 100 مضاف اليها عرض المستطيل مقسوم على 2 فيكون في منتصف المستطيل
كثير من المبرمجين يخطئون تحديد الموقع ويكتب مباشرة في تحديد المحور السيني العرض مقسوم على 2 وهذا خطا الان العرض المقسوم على 2 يساوي 50 وليس 150 وبالتالي لن تكون النقطة التي نريد المستطيل بالدوران حولها هي النقطة الواقعة في منتصف المستطيل بل ستكون خارج المستطيل لذا يجب ان توجد النقطة في منتصف المستطيل بتحديد الاحداثي السيني (X) و الصادي (Y)الذان يحددان مكان المستطيل او الصورة ثم نضيف عليها نص العرض اذا كنا نحدد المحور السيني (X) و نصف الطول في تحديد المحور الصادي(Y)

في السطر رابع و الخامس نحدد المتغير ماتريكس و تستخدم الدالة RotateAt و التي بها معاملين *Angle* تحديد زاوية الدوران وهي تشبه الزاوية التي ترسم القوس Arc وقد شرحناها في الفصل الثاني
Point نقطة محور الارتكاز التي تدور حولها الصورة وقذ اعطينها قيمة النقطة التي اعلنا عنها في الفصل الثالث

في السطر السادس نمرر متغير ماتركس للتصنيف Graphic
في السطر السابع نرسم الصورة
في السطر الثامن نعيد التصنيف Graphic للوضع الطبيعي

نفتح البرنامج ونجد الشكل الاتي



وبذلك نكون انتهينا من التصنيف Matrix وسوف نتعلم ادوات جديدة في عملية رسم البيانات

Print Document

في هذه الاداة سنتعلم الرسم و الطباعة بحيث تستطيع طباعة الرسم الذي نقوم برسمه ولكي نجد هذه الاداء يجب استدعائه بعد ان نقوم باستيراد فضاء الاسماء print كما هو موضح في الكود الاتي

```
Imports System.Drawing
Imports System.Drawing.Drawing2D
Imports System.Drawing.Printing
```

واريد ان انوه ان التصنيف Print Document هو التصنيف الاساسي و الذي يحمل كل التصنيفات الموجودة فضاء الاسماء printing فكل ما سون نتعلمه في التصنيف PrintDocunten ستجده منه تصنيف مرسوم في فضاء الاسماء Printing .

وليك اهم الخصائص التي تخص print Document

1- Default Page Sitting هي خاصية من نفس التصنيف Page Sitting و التي تعطي القيم الافتراضية للصفحة وليك اهم الخصائص في هذه الخاصية

Bound	هي خاصية للقرائة فقط تأخذ نوع المستطيل لتحديد احجام الصفحة وتحدد من الخاصية LandScape
Color	هذه الخاصية تحدد اذا كان يمكن طباعة الصفة بلون ام لا فهي خاصية من نوع Booleam
HardMargianX	خاصية تحدد الهامش اليساري للصفحة وتحدد قيمتها بوحدة القياس بالانش
HardMargianY	خاصية تحدد الهامش العلوي للصفحة وتحدد قيمتها بوحدة القياس بالانش
LandScape	خاصية تحدد اذا كانت الصفحة طولية او عرضية
Margains	خاصية تحدد الهوامش التي تخص الصفحة وهي تشبه في خصائصها خصائص المستطيل
Page Size	هذه الخاصية تحدد شكل الصفحة وبها عدة خصائص سوف نتناولها لاحقا
Paper Source	هذه الخاصية تحدد نوع الصفحة
Printable Aria	هذه الخاصية تحدد مكان الطباعة المسموح به وهي التي تحدد الهوامش في الخاصية الخاصية Bound
Printer Reselution	هذه الخاصية تحدد عدد النقاط الموجودة في الصفحة وقد لا نتطرق في هذه الخاصية

طبعا من يتعامل مع برنامج الوورد سوف يستطيع فهم هذه الخصائص
Page Size سوف نوضح خصائصها في الجدول الاتي

Hight	لتحديد طول الصفحة
Width	لتحديد عرض الصفحة
Kind	هي للقراءة فقط وتحدد نوع الصفحة

Docurmtn Name : اسم الصفحة

Printer Siting هذه الخاصية تتعامل مع الطابعة بشكل مباشر مع الطابعة واليك اهم خصائصها

CanDobliex	خاصية للقرائة فقط تتاكد اذا كانت الطابعة بها امكانية الطباعة على الوجهين
Copies	تحدد عدد النسخ المطبوعة
DeffutlPageSittng	تحدد خصائص الصفحة وهي الخصائص التي شرحناها
Duplex	لاعطاء الطباعة على ورق مقوى
FromPage	تحدد رقم اول صفحة للطباعة

IsDefoultPrinter	تحدد اذا كانت الطابعة المطلوبة هي الطابعة الافتراضية المعرفة على الجهاز
PrinterName	اسم الطابعة
Support Color	اذا كانت تدعم الالوان
Print To File	اذا كانت ترسل الورق المطبوع للملف
isValide	اذا كانت الطابعة معرفة على الجهاز
MaxmumCopies	للقراءة فقط يحدد الحد الاقصى لعدد النسخ

سوف ناخذ مثال مبسط لشرح اهم الخصائص في هذا التصنيف

سوف نقوم برسم مستطيل يقوم بتغيطة المساحة التي تغطي الهامش ثم بعد ذلك نغير الصفحة من حالة ان تكون عمودية تكون في الوضع العرضي Land Scap

فتح فورم جديد وقم بوضع اداتين من صندوق الادوات وهي
PrintDocumetn
Primt Preview Dailog

ونكتب الكود الاتي في الحدث Buttonثم نضع زر

ثم نقوم بكتابة الكود الذي في الاجراء المفعل بالحدث Paint للاداة PrintDocument
وهو الحدث PrintDocument1_PrintPage
وهو

```
Dim x, y, h, w As Integer
x = PrintDocument1.DefaultPageSettings.PrintableArea.X
y = PrintDocument1.DefaultPageSettings.PrintableArea.Y
h = PrintDocument1.DefaultPageSettings.PrintableArea.Height
w = PrintDocument1.DefaultPageSettings.PrintableArea.Width
e.Graphics.FillRectangle(Brushes.LightBlue, x, y, w, h)
```

في السطر الاول اعلنا عن متغير يحمد مستطيل سوف يرسم دخال الهامش
من السطر الثاني للسطر السادس اعلنا عن اربع متغيرات تقوم بتحديد ابعاد المستطيل الذي يسيشكل برواز الصفحة بحيث ان النقطة X تحدد يسار الصفحة و y يحدد اعلى الصفحة و W,h يحددان عرض و طول الصفحة

في السطر السابع رسمنا مستطيل يغطي كل الصفحة
في السطر الثامن مستطيل يرسم داخل الاطار

ثم نعلن عن منغير ينشى اداه اخرى من نوع Print Document

```
Private p As New PrintDocument
```

ثم ننتقل للزر الذي في الفورم ونكتب في الحدث الخاص بالضغط على الزر Button1_Click

```
p.DefaultPageSettings.Landscape = False
PrintDocument1 = p
PrintPreviewDialog1.Document = PrintDocument1
PrintPreviewDialog1.ShowDialog()
```

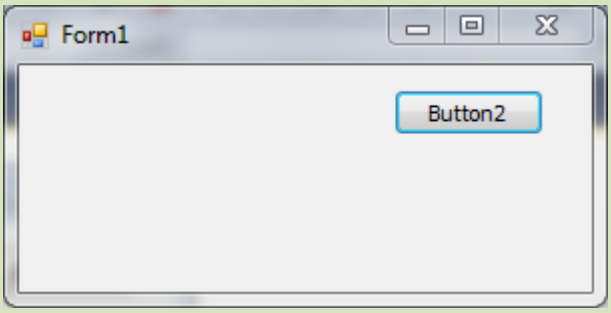
في السطر الاول نقوم بتغيير خاصية Land Scape الى False لكي يكون وضع الصفحة افقي
في السطر الثاني الاداة print Document تأخذ الخاصائص التي توجد في المتغير p
في السطر الثالث الخاصية التي تخص الاداة printPriviewDialoge وتسمى Document تاخذ كل الاداة Print Document
في السطر الرابع نقوم بامر الظهار للاداة PritntPriviewDailog

حتى يصبح الشكل النهائي الكود كلاتي

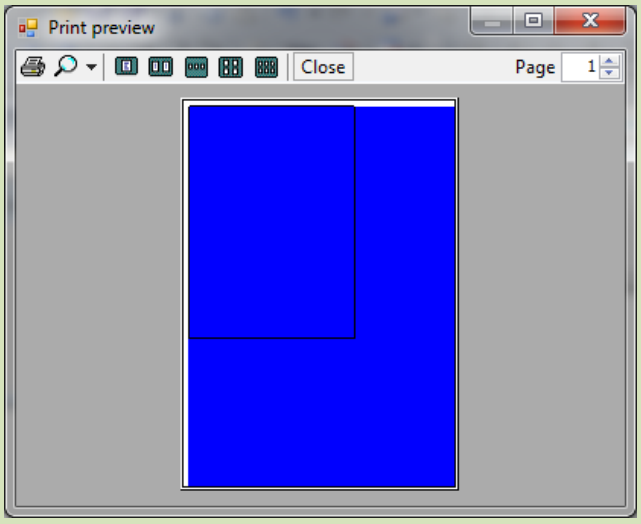
```
Imports System.Drawing
Imports System.Drawing.Drawing2D
Imports System.Drawing.Printing
Imports System.Drawing.Printing.PrinterSettings

Public Class Form1

    Private p As New PrintDocument
    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Bu
        p.DefaultPageSettings.Landscape = False
        PrintDocument1 = p
        PrintPreviewDialog1.Document = PrintDocument1
        PrintPreviewDialog1.ShowDialog()
    End Sub
    Private Sub PrintDocument1_PrintPage(ByVal sender As Object, ByVal e As System.Drawing.Printing.
        Dim x, y, h, w As Integer
        x = PrintDocument1.DefaultPageSettings.
        y = PrintDocument1.DefaultPageSettings.PrintableArea.Y
        h = PrintDocument1.DefaultPageSettings.PrintableArea.Height
        w = PrintDocument1.DefaultPageSettings.PrintableArea.Width
        e.Graphics.FillRectangle(Brushes.LightBlue, x, y, w, h)
    End Sub


End Class
```

sender As Object

وعند تشغيل البرنامج ستظهر الشاشة الاتية



وعند الضغط على الزر تظهر لنا الشاشة الاتية



قد تجد نفسك حائرا بين الادوات المستخدمة على فورم فسوف اعطيك نبذة مختصرة عى هذه الادوات

هذه الادوات تخص الطباعة هي

PrintDocument

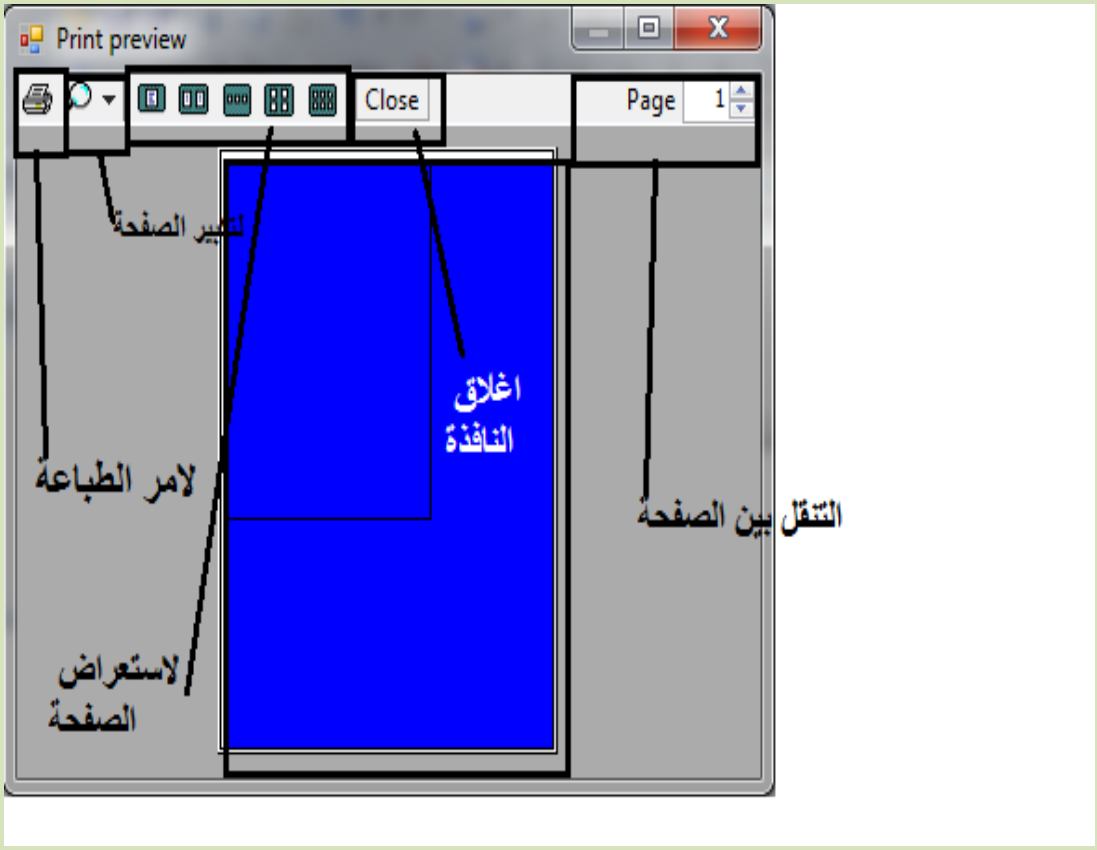
وهي الاداة التي تعتبر عصب ادوات الطباعة في الرابط بين باقي الادوات واننا قذ شرحنا خصائص بالتفاصيل سابقا

ولكن ننوه ان الحدث Paint للاداة هو الحدث الاساسي في عملية الرسم ويتم تفعيله من الادوات الثانية كما سنوضح الان

PrintPriviwDialoge

وهي اداة مسئول عن عرض الطباعة كما

شرحنا في المثال السابق وتلاحظ الاجراء ShowDailoge يقوم بتفعيل الحدث print عند الاداة PrintDocument
ونستعرض مكونات النافذ التي تظهر عن تشغيل الاداة



PageSetupDailog

هذه الاداة تقوم بعمل تغيرات في شكل الاداة printDocument بحيث يجب تغير في حجم الورقة ووضع الورقة و الهوامش ونوع الورقة وغيرها وليك استعراض بعض ادواتها عند تشغيل البرنامج وايضا بها خاصية من PrintDocument مهمتها تعديل في الخصائص فهي تسهل عليك كتابة كود كبير للتعامل مع الورق وتترك من خلالها الخيار للمستخدم في تحديد شكل الورقة واليك شكل ادوات تغيير شكل الورقة

وبالتالي ستكون مهمتك في البرمجة هو ان تجعل خصائص الاداة PrintDocument المعدلة من قبل المستخدم من هذه الاداة PageSetupDailog ثم اعطائها للاداة PrintPriveioDailog لتقوم هي بطباعتها

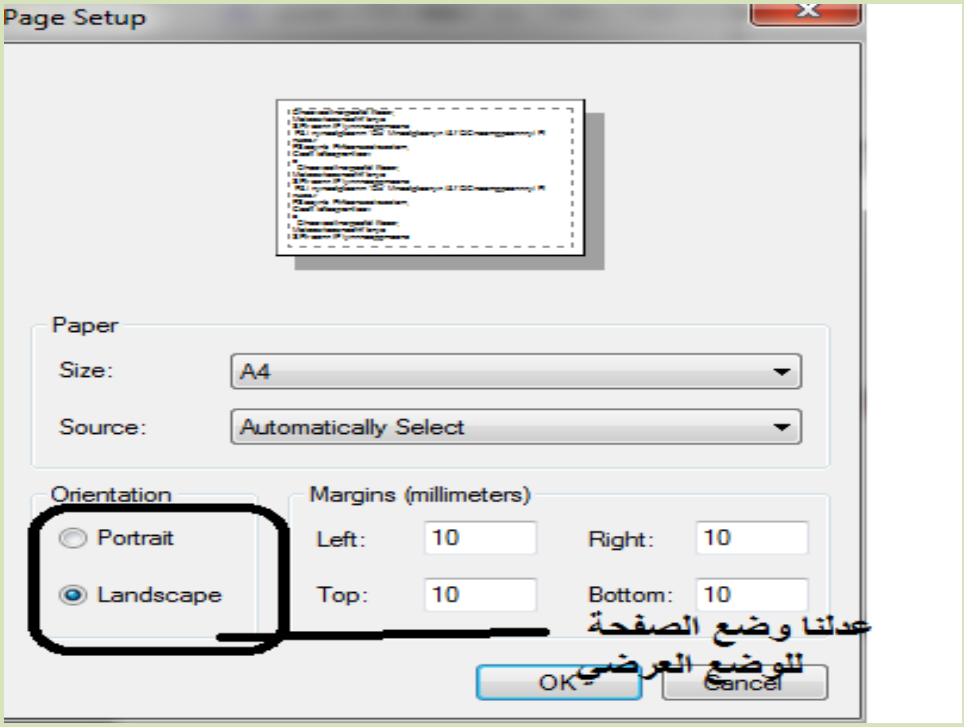
لكي تستوعب هذه الكلام عد الى المثال السابق قم بتعديل الكود المكتوب في الحدث الخاص بضغط الزر الى

```
PageSetupDialog1.Document = PrintDocument1
PageSetupDialog1.ShowDialog()
PrintDocument1 = PageSetupDialog1.Document
PrintPreviewDialog1.Document = PrintDocument1
PrintPreviewDialog1.ShowDialog()
```

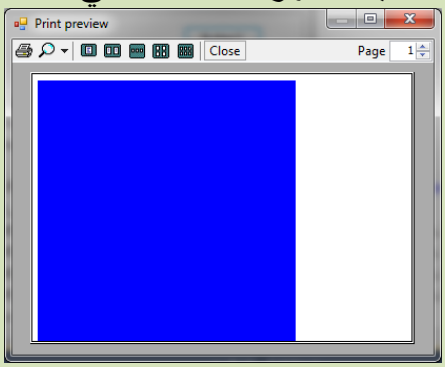
في السطر الاول قمنا بتعديل الخاصية Document بحيث يتاخذ الاداه printDocument ثم اظهرناها لكي يتم التعديل عليها
ثم اعطينا الخصائص المعدلة الى الاداه printDocument ثم الى الاداه printPriveiwDailog

عند تشغيل البرنامج شتظهر الشاشة كلاتي

عند الضغط على الزر تظهر الشاشة التي تخص الاداه PrintSetupDailoge كلاتي



وعند اغلاق الشاشة وفتح الشاشة الاتية فتظهر الشاشة كلاتي



وبهذه الطريقة نكون قد تعلمنا ادوات الطباعة

تنويه هاااااااام

اتذكر اننا كنا عدة مرات نقول انه يمكن تحويل وحدات القياس التي تقاس بها الصفحات من البكسل او البوصة الى المللي متر فيمكن عمل ذلك من خلال التصنيف Graphic من خلال الكود الاتي

```
e.Graphics.PageUnit = GraphicsUnit.Millimeter
```

مع العلم ان المتغير e هو المتغير الموجود في الاجراء الحدث Paint
فيمكن لاي متغير يحمل نوع Graphic ان يقوم بهذه الوظيفة

```
Dim D as Grahpics
```

```
D.PageUnit = GraphicsUnit.Millimeter
```
وقد شرحنا الخيارات المستخدمة في تحديد وجدة القياس في الفصل الثاني

التعامل مع ادوات التحكم

ننتقل الان الى اهم التطبيقات العملية على هذه المكتبة و التي يتعامل معها كل المبرمجين واللتي تخص ادوات التحكم فيمكنك من خلال هذه المكتبة تغيير شكل الادوات يعني يمكنت عمل زر شكله دائري تجعل خلفية الزر بلونين تجعل للنص ظل وهكذا بهذه الطريقة يضيف للواجهة الرسومية اناقة تجذب اليها المتسخدمين

ولان ننتقل الى عمل فورم على شكل دائري فهي طريقة سهلة جدا

اعمل مشروع جديد واكتب الكود الاتي على الفورم في الحدث Paint

```
Dim i As New Rectangle(0, 0, Me.Width, Me.Height)
Dim gp As New GraphicsPath
gp.AddEllipse(i)
e.Graphics.FillEllipse(Brushes.Brown, i)
Me.Region = New Region(gp)
```

في السطر الاول نقوم بعمل متغير ينشئ مستطيل على طول و عرض تساوي طول وعرض الفورم

في السطر الثاني انشانا متغير يحمل التصنيف GraphicPath

في السطر الثالث اضفنا شكل بيضاوي يحمل المستطيل الذي اعلنا عنه في السطر الاول

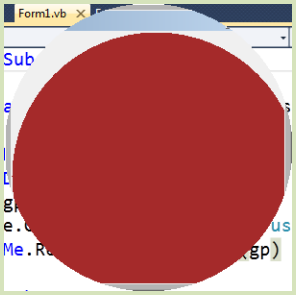
في السطر الرابع رسمنا دائرة مضللة على نفس الاطوال

ثم افي السطر الاخير استدعنا الخاصية Region

هذه الخاصية تقوم بعمل تحديد لشكل الفورم او أي اداة تحكم ك الزر وغيرها عند استدعائها ننشئ تصنيف من نوع Region هذا التصنيف عند انشائه نقوم باعطاء قيم لدالة الانشاء new هذه الدالة تعطيك 5 خيارات ساتركك تتعرف عليها لاننا قد شرحنا دوال مثلها سابقا معدا واحدة وهي التي نستخدمها في هذا المثال وهو معامل يضيف الشكل الذي نريد الادة تتحول اليه فيكون من نوع Graphic path التي شرحناها في فصل التعامل مع الفرشاه

الخاصية Region تشبه المقص الذي يقوم بقص الفورم الى الشكل الذي نريد ان يكون في المعامل path

وعند تشغيل الفورم سنجد شكل الفروم تحول الى الشكل الاتي



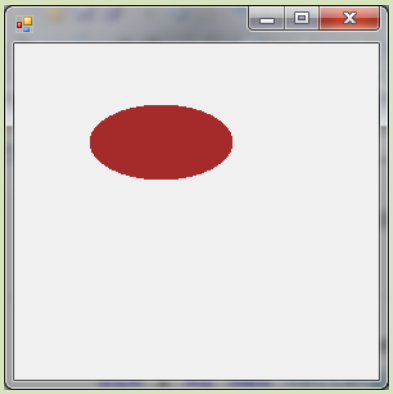
يمكنك كتابة نفس الكود على زر في الحدث Paint
ولكن عدل اطوال المستطيل ليصبح مثل اطوال الزر ويكون الخاصية Region تستدعى من الزر هكذا

Button2.Region = new Region(gp)

فيصبح الكود كلاتي

```
Dim i As New Rectangle(0, 0, Button2.Width, Button2.Height)
Dim gp As New GraphicsPath
gp.AddEllipse(i)
e.Graphics.FillEllipse(Brushes.Brown, i)
Button2.Region = New Region(gp)
```

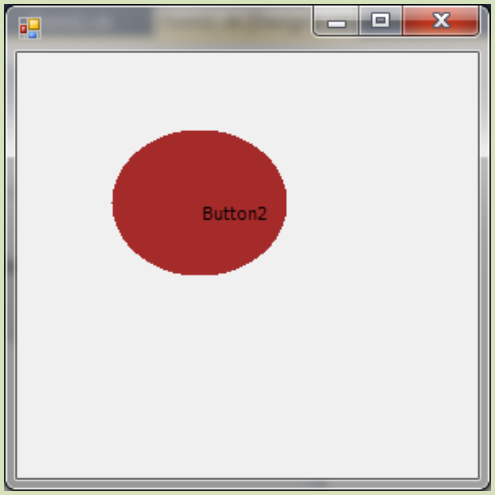
فتجد ان شكل الزر اصبح كلاتي



ولكن ستلاحظ ان النص المكتوب على الزر لا يظهرولكي تحل المشكلة ستقوم بكتابة الكود الاتي بعد الكود السابق

```
e.Graphics.DrawString(Button2.Text, Button2.Font, _
                      Brushes.Black, Button2.Width / 2, Button2.Height / 2)
```

فيظهر الزر على الشكل الاتي



نلاحظ شكل النص غير لائق في مكانه فيمكن التعديل مكانه في المعاملين الخاصين بتحديد مكانه
كما درسنا علي علمية رسم النصوص فهل تذكر ان المعاملات رسم النصوص؟ كانت كلاتي
المعامل الاول يقوم باخذ النص فاعطيناه القيمة الموجدة في الخاصية Text
المعامل الثاني يحدد الخط فاعطيناه الخاصية Fontالتي تخص خطوط الزر
المعامل الثالث خاص بلون الزر ولكن لا يمكن اعطاءه لون الخط للخاصية FareColorلان لون الخط من نوع Color ولكن المعامل من نوع Brush و الخاصية FareColor من نوع Color فلن يقبلها المعامل
المعامل الرابع و الخامس يحددان موقع النص على الفورم

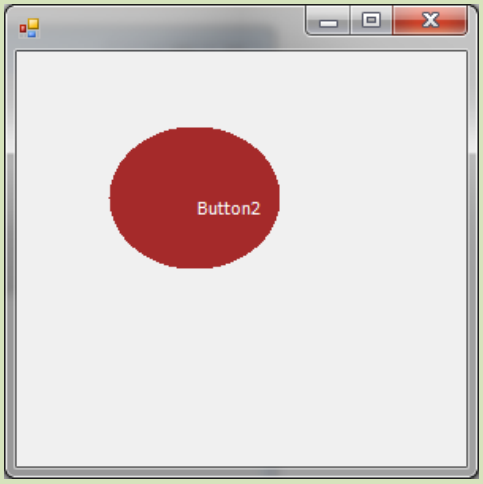
يمكنك ايجاد حل لمشكلة فجعل لون النص الخاص بالزر يساوي لون النص الذي نرسمه في المعامل الرابع الذي هو من نوع Brush وذلك عن طريق استخدام الفرشاه SoildBruah
قم بتبديل قيمة المعامل الخاص بتغيير لون بالكود الاتي

```
New SolidBrush(Button2.ForeColor)
```

بحيث وانت تعطي القيمة للمعامل يمكنك انشاء أي تصنيف فقمنا بانشاء تصنيف SoildBrush واضفنا اليه قيمة لون الخط الخاص بالزر فيكون شكل الكود الخاص برسم النص على الزر كلاتي

```
e.Graphics.DrawString(Button2.Text, Button2.Font, _
                      New SolidBrush(Button2.ForeColor), Button2.Width / 2, Button2.Height / 2)
```

قم بتغيير لون النص وقم بتشغيل البرنامج فيتغير لونه



غيرنا لون النص الى اللون الابيض

ملخص الفصل الخامس

- تعرفنا على التصنيف Matrix وما هي وظائفه
  - Scale
  - Transport
  - Shear
  - Rotate
- تعرفنا على ادوات الطباعة
- **PrintDocumen**
- **PageSetupe**,
- **printPreviewDailog**
- تعرفنا على كيفية تغيير شكل ادوات التحكم

لفصل السادس

تطبيقات عملية

بعد كل ما تعلمناه في الفصول السابقة دعنا الان نطبق تطبيقات عملية على كل ما تعمناه حتى تتعرف كيفية استخدام مكتبة الرسم على ارض الواقع ففي هدا الفصل سوف اشرح لك بعض الامثلة العملية التي تضعك على اول الطريق في استخدام هذه المكتبة

في نهاية الفصل ستكون قادر على

- كيف يتم رسم الاشكال عن طريق الماوس
- تحريك الصور
- تسيقات النصوص

الرسم باستخدام الماوس

هنا ساعلمك كيف تقوم بعمل برنامج شبيه ببرنامج الرسام بحيث تتعرف على كيفية رسم الاشكال من خلال الماوس وكيفية تلوينها وكانك تقوم بعمل برنامج رسام

وفي السطور القادمة سوف نتعمل الرسم على مراحل حتي يسهل عليك الرسم ثم نقوم بعمل برنامج الرسام فتابع معي

المرحلة الاولى الرسم بطريقة تقليدية

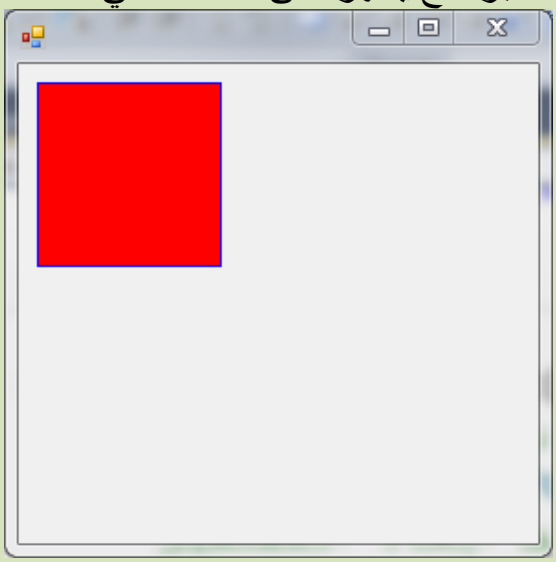
افتح مشروع جديد وقم بكتابة المتغيرات الاتية في قسم الاعلانات العام

```
' تعريف نقطة بداية الرسم
Dim startPoint As Point = New Point(10, 10)
' تعريف أبعاد المستطيل
Dim RectangleSize As Size = New Size(100, 100)
```

ثم اكتب الكود الاتي في الحدث Paint

```
' تعريف المستطيل
Dim Rect As Rectangle = New Rectangle(startPoint, RectangleSize)
' تلوين المستطيل
e.Graphics.FillRectangle(Brushes.Red, Rect)
' رسم المستطيل
e.Graphics.DrawRectangle(Pens.Blue, Rect)
```

اظنك تعرفت على هذا الكود من قبل فقد شرحناه مرارا وتكرارا
واذا فتحنا البرنامج سوف نجد البرنامج يظهر على الشكل الاتي



المرحلة الثانية : سوف نحرك المستطيل عن طريق الماوس ولكي يتم ذلك سوف نقوم ببعض التعديلات في الكود ولكن سوف اكتب لك الكود كله من الاول وكانه لم يكتب من قبل فاكتب الكود الاتي في قسم الاعلانات

```
' نقطة بداية الرسم
Private startPoint As Point = New Point(10, 10)
' أبعاد المستطيل
Private RectangleSize As Size = New Size(100, 100)
' تعريف المستطيل
Private Rect As Rectangle
' تعريف نقطة سوف تستخدم لتحديد مكان نقطة تحريك المستطيل
Private emptyPoint As Point = Point.Empty
' تعريف متغير رقمي يشير الي نقطة
'رسم المستطيل في الاتجاه الأفقي أثناء حركة الماوس
Private x As Integer = startPoint.X
' تعريف متغير رقمي يشير الي نقطة
' رسم المستطيل في الاتجاه الرأسي أثناء حركة الماوس
Private y As Integer = startPoint.Y
```

وسوف نكتب الكود الاتي في الحدث Paint

```
Rect = New Rectangle(x, y, RectangleSize.Width, RectangleSize.Height)
Dim color1 As Color = Color.FromArgb(255, Color.DarkBlue)
Dim color2 As Color = Color.FromArgb(255, Color.Cyan)
Dim lgb As New Drawing2D.LinearGradientBrush(Rect, color1, color2, 45, True)
e.Graphics.FillRectangle(lgb, Rect)
```

في الكود السابق رسمنا مستطيل وظللناه بلولنين

نريد ان نحركه لذا سوف يقوم بتغيير المتغيرين X, y الذان يحددان موقع المستطيل ويتم ذلك عن طريق الماوس ويتم ذالك في اننا نكتب الكود في ثلاثة احداث Mouse Dawn
. Moase Move , mouse UP
اكتب الكود الاتي في الحدثMouseDawn

```
If e.Button = Windows.Forms.MouseButtons.Right Then Return
   emptyPoint.X = e.Location.X
emptyPoint.Y = e.Location.Y
```

في الكود السابق قمنا بالتاكد اذا كان الزر المضغوط هو الزر الايسر واذا كان كذكل نقوم بالاتي
المتغير المسمى emptyPoint وهو مكان الماوس تعطيه نقطة تحديد مكان الماوس وهي يمكن تحديد من خلال المعامل e الموجود في داخل الاجراء وهو يحدد نقطة مكان الماوس عن طريق الخاصية Location

بعد ذلك نكتب الكود الاتي في الحدث mouseMove

```
If emptyPoint = Point.Empty Then Return
' هما نمرر قيم جديدة لنقطة بداية رسم المستطيل في الاتجاه الأفقي
x = e.X - emptyPoint.X + startPoint.X
' هنا نمرر قيم جديدة لنقطة بداية رسم المستطيل في الاتجاه الرأسي
y = e.Y - emptyPoint.Y + startPoint.Y
' هنا نطلب من الفورم أن يعيد رسم المستطيل
Me.Invalidate()
```

في الكود السابق نقول نحدد اذا كانت المتغير EmptyPoint فارق يعني لم يتم تحديد مكان الماوس او بمعنى ادق اذا لم يقم المستخدم بالضغط على الزر الايسر للماوس وتم استدعاء الاجراء الخاص بـmouse Dawn والذي يتم من خلاله تحديد موقع الماوس , اذا كان الشرط مؤكد او صحيح فانه يخرج خارج الاجراء باستخدام كلمة return او يمكن استخدام كلمة اخرى وهي Exit Sub وهي المستخدمة في لغة الفيجوال بيسك 6 وتمت اضافتها في الفيجوال بسك دوت نت

في السطر الثاني يعني لم يتم الخروج من الاجراء فانه يتم تحديد قيمة للمتغيرين X,Y وذلك عن طريق المعادلة الاتية

مكان الماوس – النقطة التي حددت مكان الماوسEmptyPoint + بداية رسم المستطيل

ما معنى هذه المعادلة

مكان الماوس والتي تحدد من خلال المعامل e الموجود في الاجراء فهوي يحدد مكان الماوس

النقطة التي حددت مكان الماوسEmptyPoint وهي تم تحديد من الاجراء الخاص بالحدث mouse Dawn

بداية رسم المستطيل : وهي اول نقطة حددت لرسم المتسطيل ولو قمنا مسحها ستلاحظ انك تحرك المستطيل واذا ضغطت على المستيطل مرة اخرى فانه يرسم الستطيل من مكانه الاول

في السطر الاخير الاجراء invalidate هي مسئولة عن اعادة تحميل الفورم فاي تغير يحدث في خصائص الفورم فهو يقوم باعدة تحميل الفورم تلقائيا حتى تتغير الخصائص في حالة التشغيل البرنامج او run time ويمكن استبادهل باجراء اخر وهو Refresh

سوف نكتب الكود الاتي في الحدث mouse up

```
' هنا نتأكد بأن المستخدم قام باستخدام الباتون الخاص بيسار الماوس
If e.Button = Windows.Forms.MouseButtons.Right Then Return
' هنا نمرر قيمة جديدة الب نقطة باية رسم المستطيل
startPoint.X += e.X - emptyPoint.X
startPoint.Y += e.Y - emptyPoint.Y
' هنا نعيد المتغير الذي يشير الي مكان تحريك المستطيل الي قيمته الأصلية وهي لا شئ
emptyPoint = Point.Empty
```

في هذا الاجراء نعطي الامر لانهاء الرسم حدا لا يتم تحرك المستطيل حتى لو رفعت يدك عن الزر الماوس فيتاكد ان الزر المستخدم هو الزر الايسر فاذا كان الايمن يخرج من الاجراء ويعطي القيم النهائيه للمكان المستطيل عند رفع المستخدم يده عند زر الماوس

ويقوم بمسح الفيم في المتغير emptypoint

وعند تشغيل البرنامج قم بتحريك المواس بعد الضغط على الزر الايسر ستجد المستطيل يتحرك معك

ستلاحظ ان شكل المستطيل يهتز اثناء التحرك فيكون شكله غير لائق لذا انصحك بتعديل خاصية في الفورم وهي double Bofferd وجعلها تساوي True

قد تجد صعوبة في فهم المعادلات لذا انصحك بعد كتابة الكود بعمل تجربة بسيط فتقوم بمسح احد المتغيرات من المعادلات التي كتبناها في الحدث mouseMove ولتكن startPoint وجرب البرنامج سوف تعرف اهمية كل كود في الكود السابق وكما قال جابر بن حيان *يا بني عليك بالتجربة لتصل الى المعرفة* قد لا اكون موفق في توفير ما يكفي من معلومات لتساعدك في فهم الكود لذا انصحك ان تعيد كتابة هذا الكود بمعرفتك وبفهمك وسوف تستطيع فهمها بسهولة بعد ذلك

المرحلة الثالثة نقوم بتغيير حجم المستطيل

سوف نقوم بتعديل بسيط في الكود السابق ولكن ارجو ان تكون قد فهمت الكود السابق لان هذا الكود مبني على الكود السابق لاننا سوف نعدل مكان الكود فبدل ما نستخدم الماوس في تحريك المستطيل سوف نستخدمه في تغيير حجم المستطيل

فسوف نقوم بعمل متغيرين ليقوما بتحديد حجم المتسطيل هما Width, height في قسم الاعلانات العامة وسوف نساويهما بخصائص الطول و العرض في المتغير RectangleSize و السبب هو افترض اننا لم نساويها بالمتغير لن يقوم السمتطيل بالرسم فقم بمساواتها كما في الكود الاتي

```
Private height As Integer = RectangleSize.Height
Private Width As Integer = RectangleSize.Width
```

ثم نعدل في كود الرسم بحيث يكون حجم المستطيل يساوي المتغيرين
فنعدل الكود في الحدث paint

```
Rect = New Rectangle(x, y, Width, height)
Dim color1 As Color = Color.FromArgb(255, Color.DarkBlue)
Dim color2 As Color = Color.FromArgb(255, Color.Cyan)
Dim lgb As New Drawing2D.LinearGradientBrush(Rect, color1, color2, 45, True)
e.Graphics.FillRectangle(lgb, Rect)
```

ثم نعدل الكود الموجود في الحدث mousemove ليتم تغيير حجم المستطيل فنبدل النقطة Strarpoint الى rectangleSize كما في الكود الاتي

```
If emptyPoint = Point.Empty Then Return
Width = e.X - emptyPoint.X + RectangleSize.Width
height = e.Y - emptyPoint.Y + RectangleSize.Height
Me.Invalidate()
```

ثم نعدل الكود في الحدث mouse up ليكون الشكل الاتي

```
 If e.Button = Windows.Forms.MouseButtons.Right Then Return
 RectangleSize.Width += e.X - emptyPoint.X
RectangleSize.Height += e.Y - emptyPoint.Y
 emptyPoint = Point.Empty
```

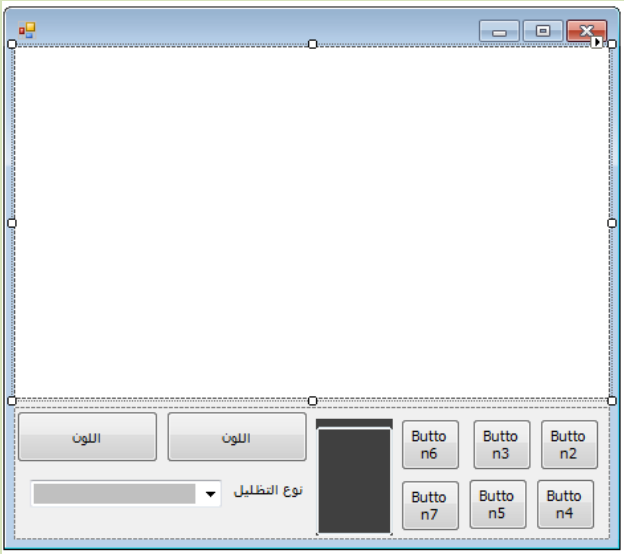
وقم بشغيل البرنامج ستجد انه يعدل في حجم المستطيل

جرب هذه الكود في اشكال اخرى

عمل برنامج رسم

سوف نعطي بعض المبادئ التي تساعد لعمل برنامج رسام اعطينا في السطور السابقة عن طريقة الرسم عن طريق الماوس وهذا هو بيت القصيد ولكن نريد ان نتطبق ما تعلمناه في الرسم عن طريق الماوس بشكل يؤهلك على ان تصمم برامج رسم تضاهي برامج الرسم التجارية فتعلى معي افتح مشروع جديد و قم بعمل الخطوات الاتية

الخطوة الاولى صمم نودج فورم ليكون مثل هذا الشكل



الخطوة الثانية سوف نعدل في تصميمات الازرار التي على اليمني ليكون لانها سوف تقوم بتحديد الاشكال التي نريد رسمها

سوف نقوم الرسم على الاشكال على الازرار التي على يمين الشاشة لانها ستمثل الاشكال التي سيتم رسمها فسوف نقوم تمسح كل النصوص التي على الازرار بحيث تمسح النص في الخاصية Text لتصبح فارغة ثم اكتب الكود الاتي على حدث Paint في الزر الاول Buttom2

```
Dim pens As New Pen(Brushes.Black, 5)
Dim rec As New Rectangle(2, 2, Button2.Width - 10, Button2.Height - 10)
e.Graphics.DrawEllipse(pens, rec)
```

هنا نرسم الزر الاول شكل بيضاوي حتى اذا ضغط المستخدم عليه يجهز القلم الذي سيرسم به الشكل البيضاوي

ثم نجهز الزر المسئول عن رسم المستطيل فنكتب الكود الاتي

```
Dim pens As New Pen(Brushes.Black, 5)
Dim rec As New Rectangle(2, 2, Button5.Width - 10, Button5.Height - 10)
e.Graphics.DrawRectangle(pens, rec)
```

ثم نكتب الكود في الزر المسئول عن رسم الشكل المضلع Button 4

```
Dim pens As New Pen(Brushes.Black, 3)
Dim p1 As New Point(10, 10)
Dim p2 As New Point(30, 10)
Dim p3 As New Point(30, 30)
Dim p4 As New Point(20, 40)
Dim p5 As New Point(20, 20)
Dim p6 As New Point(10, 10)
Dim point() As Point = {p1, p2, p3, p4, p5}
e.Graphics.DrawPolygon(pens, point)
```

الكود في الزر المسئول عن رسم المنحنى Button 3

```
Dim pens As New Pen(Brushes.Black, 3)
Dim p1 As New Point(5, 5)
Dim p2 As New Point(30, 5)
Dim p3 As New Point(30, 30)
Dim p4 As New Point(40, 30)
Dim p5 As New Point(40, 40)
Dim p6 As New Point(5, 40)
Dim point() As Point = {p1, p2, p3, p4, p5}
e.Graphics.DrawCurve(Pens, point)
```

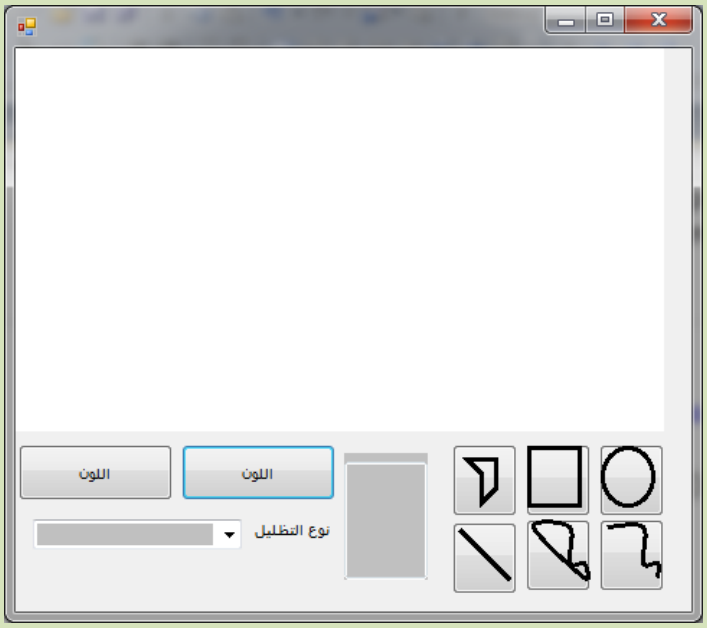
الكود في الزر المسئول عن رسم المنحنى المغلق في Button 7

```
Dim pens As New Pen(Brushes.Black, 3)
Dim p1 As New Point(5, 5)
Dim p2 As New Point(30, 5)
Dim p3 As New Point(30, 30)
Dim p4 As New Point(40, 30)
Dim p5 As New Point(40, 40)
Dim p6 As New Point(5, 40)
Dim point() As Point = {p1, p2, p3, p4, p5}
e.Graphics.DrawClosedCurve(pens, point)
```

الكود في الزر المسئول عن رسم خط مستقيم في Button 6

```
Dim pens As New Pen(Brushes.Black, 3)
Dim p1 As New Point(5, 5)
Dim p2 As New Point(40, 40)
e.Graphics.DrawLine(pens, p1, p2)
```

طبعا نحن قد شرحنا كل ذلك سلفا في الفصول السابقة نحن فقط نطبق ما شرحناه
وعند تشغل البرنامج ستجد ان البرنامج ظهر على الشكل الاتي



الخطوة الثالثة الان نجهز علبة الالوان

الان نجهز علبة الالوان وتلاحظ اننا لدينا علبتين لاننا سوف نظل الاشكال بالفرشاة ذات لونين
فاننا سوف نكتب الكود الاتي في الحدث Click للزر Button 8

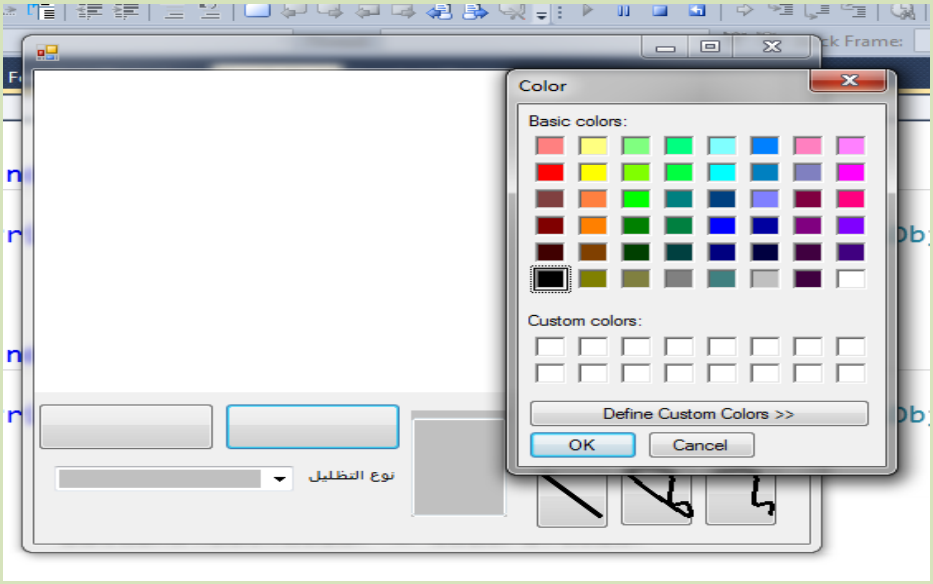
```
Dim colord As New ColorDialog
colord.ShowDialog()
Button8.BackColor = colord.Color
```

في الكود السابق انشأنا كائن الاداة صندوق الادوات وهي شاشة تظهر وتتيح للمستخدم اختيار اللون الذي نرد ان نستخدمه ثم نجعل خلفية الزر مثل اللون الذي اخترناه

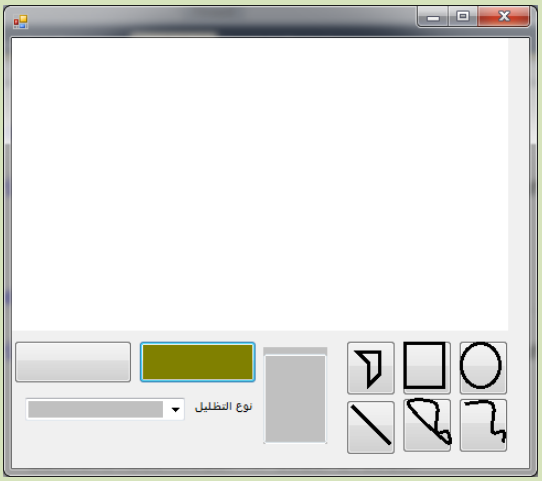
سنضع نفس الكود مع تعديل اسم الزر في الزر الثاني Button 9 في الحدث Click

```
Dim colord As New ColorDialog
colord.ShowDialog()
Button9.BackColor = colord.Color
```

وعند تشغيل البرنامج اضغط على الزر الاول فيظهر صندوق كلاتي



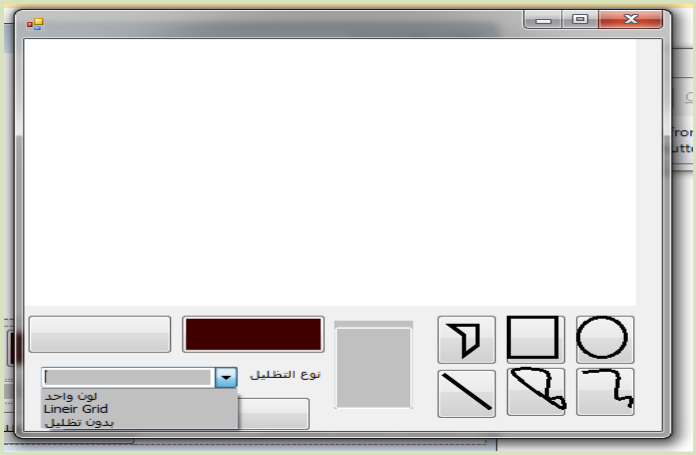
وعند اختيار أي لون والضغط على Ok فتظهر الشاشة كلاتي



ثم نعدل ال ComboBox المسئول عن تحديد طريقة التظليل فنقوم بادخال اربعة عناصر وهي

- لون واحد
- Lineir Grid
- بدون تظليل

ثم قم بتشغيل البرنامج وافتح الCombo فتجد العناصر تظهر كل الشكل الاتي



المرحلة الرابعة وهي تجهيز المربع الرمادي بحيث اذا اردنا رسم مربع يظهر صورة المربع عليه

ننتقل الان الى اخر نقطة في اعداد البرنامج سوف نعد اداة الرسم بحيث انه اذا تم اختيار المربع فتظهر على المربع الرمادي صورة المربع استدعدادا لرسم المربع فنقوم بكتابة الكود

1- نعلن عن متغير من نوع رقمي في قسم الاعلانات العامة سوف نستخدمه في تحديد نوع الاشكال التي نريد رسمها فنكتب الكود الاتي في قسم العلانات العامة

```
Private Shap As Integer = 1
```

طبعا اعطينها القيمة 1 حتى يتم رسم شكل الدائرة عند تشغيل البرنامج مباشرة

2- ثم نكتب الكود الاتي في الحدث paint للGroubBox

```
Select Case Shap
    Case 1
        Dim pens As New Pen(Brushes.Black, 3)
        Dim rec As New Rectangle(2, 2, GroupBox1.Width - 10, GroupBox1.Height - 10)
        e.Graphics.DrawEllipse(pens, rec)
```

```
    Case 2
        Dim pens As New Pen(Brushes.Black, 3)
        Dim rec As New Rectangle(2, 2, GroupBox1.Width - 10, GroupBox1.Height - 10)
        e.Graphics.DrawRectangle(pens, rec)

    Case 3
        Dim pens As New Pen(Brushes.Black, 3)
        Dim p1 As New Point(10, 10)
        Dim p2 As New Point(30, 10)
        Dim p3 As New Point(30, 30)
        Dim p4 As New Point(20, 40)
        Dim p5 As New Point(20, 20)
        Dim p6 As New Point(10, 10)
        Dim point() As Point = {p1, p2, p3, p4, p5}
        e.Graphics.DrawPolygon(pens, point)

    Case 4

        Dim pens As New Pen(Brushes.Black, 3)
        Dim p1 As New Point(5, 5)
        Dim p2 As New Point(30, 5)
        Dim p3 As New Point(30, 30)
        Dim p4 As New Point(40, 30)
        Dim p5 As New Point(40, 40)
        Dim p6 As New Point(5, 40)
        Dim point() As Point = {p1, p2, p3, p4, p5}
        e.Graphics.DrawCurve(pens, point)
Case 5


    Dim pens As New Pen(Brushes.Black, 3)
    Dim p1 As New Point(5, 5)
    Dim p2 As New Point(30, 5)
    Dim p3 As New Point(30, 30)
    Dim p4 As New Point(40, 30)
    Dim p5 As New Point(40, 40)
    Dim p6 As New Point(5, 40)
    Dim point() As Point = {p1, p2, p3, p4, p5}
    e.Graphics.DrawClosedCurve(pens, point)

   Case 6

       Dim pens As New Pen(Brushes.Black, 3)
       Dim p1 As New Point(5, 5)
       Dim p2 As New Point(40, 40)
       e.Graphics.DrawLine(pens, p1, p2)

End Select
```

في الكود السابق كبتان الاتي

في السطر الاول استخدمنا جملة الاختبار للمتغير Shap

<u>في حالة كونه رقم 1</u> فانه سيقوم برسم شكل بيضاوي في المربع حالة استعداد رسم بيضاي

<u>في حالة كونه رقم 2</u> فانه سيقوم برسم شكل المستطيل في المربع

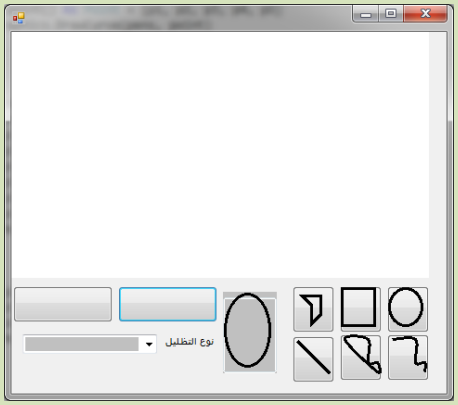
<u>في حالة كونه رقم 3</u> فانه سيقوم برسم شكل مضلع في المربع

<u>في حالة كونه رقم 4</u> فانه سيقوم برسم شكل منحنى في المربع

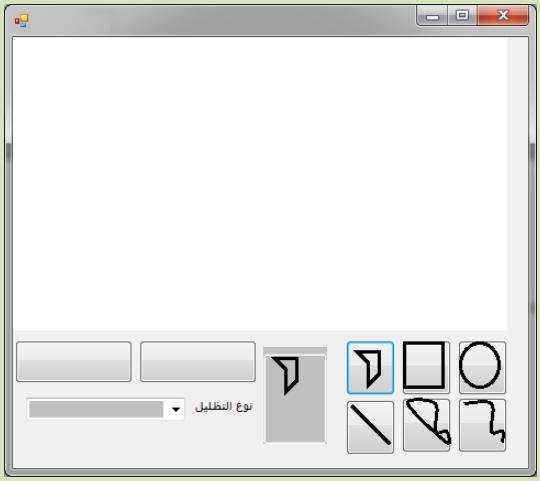
<u>في حالة كونه رقم 5</u> فانه سيقوم برسم شكل منحنى مغلق في المربع

<u>في حالة كونه رقم 6</u> فانه سيقوم برسم شكل خط مستقيم في المربع

واذا شغلنا البرنامج ستلاحظ الاتي



واذا ضغطنا على زر المضلع ستظهر في الشكل الاتي



ولكي تعطي القيم للمتغير يجب عند الضغط على كل زر نعطي القيمة للمتغير Shape لذا نكتب علند النقرClick للزر Button 2 هذا الكود

```
Shap = 1
GroupBox1.Refresh()
```

وعند الزر Button 5

```
Shap = 2
GroupBox1.Refresh()
```

وعند الزر Button 4

```
Shap = 3
GroupBox1.Refresh()
```

وعند الزر Button 3

```
Shap = 4
GroupBox1.Refresh()
```

وعند الزر Button 7

```
Shap = 5
GroupBox1.Refresh()
```

وعند الزر Button 6

```
Shap = 6
GroupBox1.Refresh()
```

تلاحظ الكود GroupBox1.Refresh مهمته هي اعدة تحميل الصندوق حتى يتم الرسم عليه مرة اخرى

وهكذا وبهذه الطريقة نقوم باعداد برنامج الرسم استعداد لرسم الشكل على بيئة الرسم

الخطوة الخامسة سوف نرسم الاشكال على بيئة الرسم وهي pectureBox2

اريدك ان تعرف اننا لن نكتب الكود كله مرة واحدة بل سنكتبه على مراحل فسوف ان تجد انني اعدل في الكود عدة مرات حتى يسهل عليك الفهم

سوف نبدأ برسم الدائرة و المستطيل ولكي نقوم بعمل ذلك سوف نعرف متغيرات اساسية في قسم الاعلانات العامة

```
Dim size As New Size(0, 0)    ' العرض
Dim startDra As Boolean = False 'قفل الرسم
Dim startp As New Point(0, 0)
Dim gr As Graphics
```

في السطور السابقة قمنا بتعرف المتغيرات الاتية

Size: مسؤول عن تحديد حجم الشكل

StatDra: هو بمثابة قفل امان حتى لا يتم رسم أي شيء بدون ان نسمح له

StartPoint : لتحديد مكان الرسم

Gr: هذا المتغيرمن نوع graphics ليقزم بعملية الرسم على اللوحة

ثم ننتقل الى اللوحة البيضاء ولتي اسمها picture2 و التي سوف نرسم عليها الاشكال
فنكتب الكود الاتي في الحدث mouse Dawn

```
startDra = True
startp.X = e.X
startp.Y = e.Y
```

في السطر الاول فتحنا قفل الرسم حتى يتم الرسم
في السطر الثاني و الثالث حددن موقع الرسم على اللوحة بحث اننا اعطينها الاحداثيات السينية و الصادية للمتغير الذي يكون مسئول عن نقطة البداية

ثم نكتب الكود الاتي في الحدث mouse Move
في

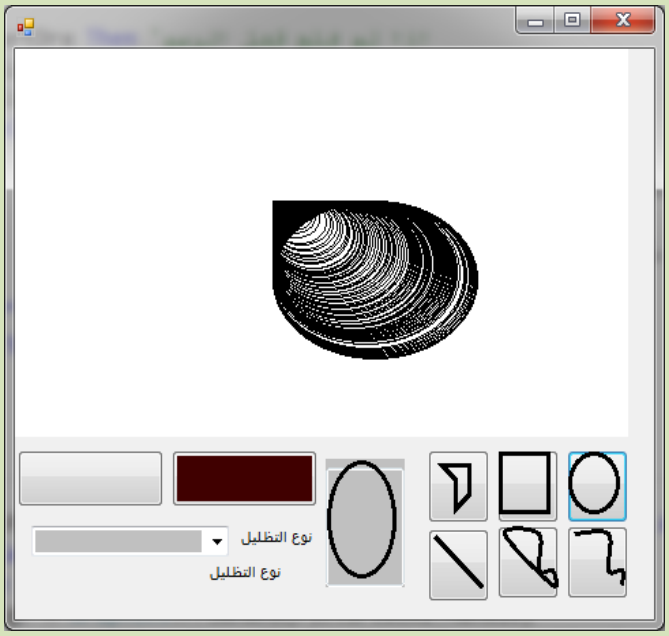
```
1  If startDra Then 'اذا تم فتح قفل الرسم
2      size.Width = e.X - startp.X ' حساب العرض
3      size.Height = e.Y - startp.Y 'حساب الطول
4      Select Case Shap
5          Case 1
6              Dim rec As New Rectangle(startp.X, startp.Y, size.Width, size.Height)
7              gr = Graphics.FromHwnd(PictureBox2.Handle)
8              gr.DrawEllipse(Pens.Black, startp.X, startp.Y, size.Width, size.Height)
9              PictureBox2.Invalidate(rec)
10     End Select
   End If
```

في الكود السابق تاكدنا ان كان قفل الرسم مفتوح ام الا فاذا كان مفتوح يتم الرسم
في السطر الثاني يحدد عرض الرسم بحيث يحدد الإحداثيات السيني للمكان الماوس مطروح منه الاحداثي السيني لنقطة البداية التي حددناها
كذلك في السطر الثالث فيحدد الطول عن طريق الإحداثيات الصادية
من السطر الرابع الى السطر العاشر : فانه يدخل في جملة اختبار شرط بحيث يحدد ما نوع الشكل الذي يريد رسمه فاذا كان رقم المتغير Shap 1 فهو يعني رسم دائرة فننتقل للسطر السادس الذي يقوم بتحديد المستطيل الذي سوف نرسم به الشكل

في السطر السابع نحدد السطح الذي نريد الرسم عليه وهو اللوحة البيضاء picture 2 وقد شرحناها بالتفصيل في الفصل الاول

في السطر الثامن نرسم الدائرة بقلم لونه اسود وبنفس ابعاد المستطيل الذي كتبناه في السطر السادس

في السطر التاسع : هذا الكود مهم جدا فيجب عليك التركيز جيدا في فهمه : نحن عندما نعطيل للكمبيوتر رسم فانه يرسم الصورة بدون ان يممحها لذا لولم نكتب الكود ستلاحظ رسم الدائرة بشكل متكرر كما في الشكل الاتي اذا لم نكتب الكود هذا



تلاحظ وانت ترسم الدائرة وانت تسحب الفارة فان الداشرة تتكرر معك وذا لا يليق لذا فان الامر في السطر التاسع يقوم بمسح للدائرة عند الرسم حتى تحدد الشكل النهائي للدائرة فيجب كتابة الكود هذا ولكن يجب ان تحدد المساحة التي يجب ان يتم مسحها لان لو تركتها بدون تحديد مساحة يعني لو كتب الكود كلاتي

Picture2.invalidate()

بدون ان تدخل المعامل الذي يحدد المساحة التي يجب المسح فيها فقط فانه يمسح اللوحة كلها كلما اردت رسم شيء اخر يعني لو رسم دائرة وتريد رسم دائرة اخرى فانه يمسح الدائرة الاولى

لذا اذا اردت حديد المساحة التي يجب مسح الصورة التي ترسم فيها فكتبا الكود الموجود في السطر التاسع

Picture2.invalidate(rec)

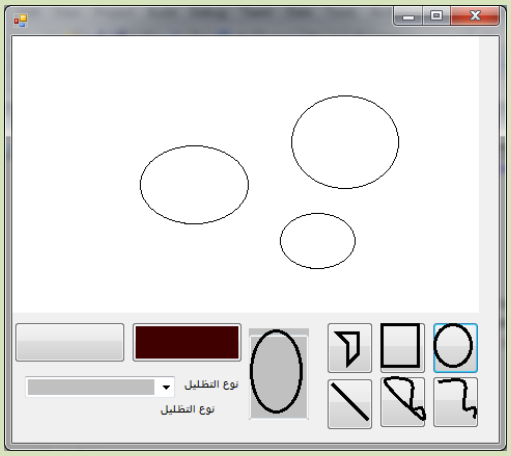
Rec : هو المتغير الذي كان يحدد المستطيل الذي يحدد طول وعرض الدائرة سندخله في المعامل حتى يحدد فقط المساحة التي نريد رسم فيها الدائرة فيقوم بمسح هذه المساحة فقط دون ان يرسم اللوحة كاملة

في الحدث mouse Up نكت بالكود الاتي

```
Select Case Shap
    Case 1
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawEllipse(Pens.Black, startp.X, startp.Y, size.Width, size.Height)

End Select
startDra = False
```

في هذا اكود ايضا استخدمنا نفس الكود الذي شرحناه وهو جملة اختبار الرسم فاذا كان الرسم دائرة فانه يحدد سطر الرسم وهو picture2 ثم يرسم الدائرة في شكلها النهائي بعد تحديد شكلها في الحدث السابق
في السطر الاخير نغلق قفل الرسم

فلو فتحت البرنامج وارسم دائرة ستجده يرسم الدائرة باشكل الذي تريده كما هو موضح في الشكل الاتي



ولكننا لم نصمم البرنامج لكي رسم دائرة فقط فسوف نرسم مربع ايضا لذا سوف نعدل في الكود كما قلنا اننا نعدل الكود عدة مرات ليسهل فهم الكود

ففي الكود المكتوب في الحدث mouse move سنعدل الكود بحيث اننا نعطي في جملة اختبار المتغير Shap الذي حدد رسم الاشكال فاننا نعلطة الحالة الثانية Case 2 وكتب الكود الاتي

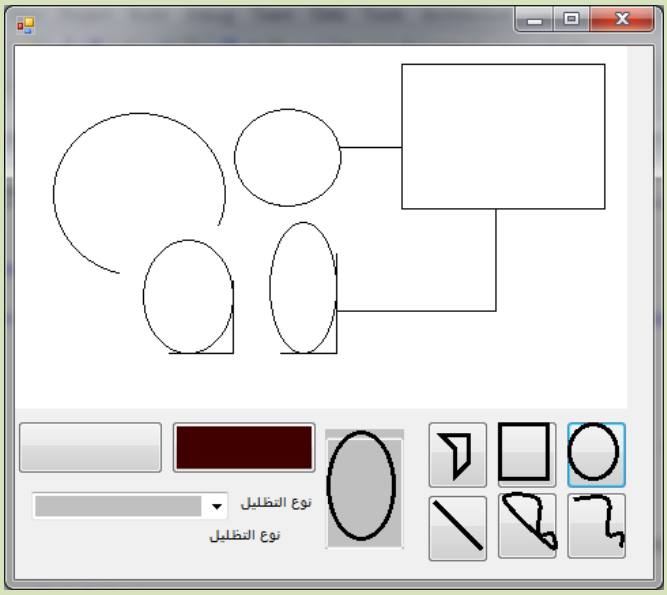
```
Case 2
    Dim rec As New Rectangle(startp.X, startp.Y, size.Width, size.Height)
    gr = Graphics.FromHwnd(PictureBox2.Handle)
    gr.DrawRectangle(Pens.Black, startp.X, startp.Y, size.Width, size.Height)
    PictureBox2.Invalidate(rec)
```

تلاحظ نفس الكود في ال Case 1 ولكن بدل ما اننا نكتب drawEllipse يعني ارسم بيضواي قلنا ارسم Draw Rectangle

وكذلك في الكود الخاص بالحدث Mouse Up فاننا سنعدل الكود الموجود فقط في جملة الاختبار وندخل Csae 2

```
Case 2
    gr = Graphics.FromHwnd(PictureBox2.Handle)
    gr.DrawRectangle(Pens.Black, startp.X, startp.Y, size.Width, size.Height)
```

نفس ما شرحنا في السطور السابقة ولكن عدلنا الكود بحيث يرسم مستطيل بدل دائرة
قم بتشغيل البرنامج وارسم مربع و دائرة وسوف تجده يرسم ما تريد من مرمبع و دائرة



بهذه الطريق وضعتك في اول الطريق وسوف ننتقل لرسم باقي الاشكل

سوف نرسم الان الخط المستقيم في الزر السادس

سنعدل في الكود لكي نتمكن من رسم الخط المستقيم

ففي قسم الاعلانت العامة نعلن عن متغير نشئ كائن نقطة اخرى لتكون نقطة النهاية كما هو موضح في الشكل الاتي

```
Dim size As New Size(0, 0)    ' العرض
Dim startDra As Boolean = False 'قفل الرسم
Dim startp As New Point(0, 0)
Dim EndPt As New Point(1, 1)
Dim gr As Graphics
```

المتغير الذي اضفناه

ثم نعدل الكود في الحدث mouse Move

```
If startDra Then 'اذا تم فتح قفل الرسم
    size.Width = e.X - startp.X ' حسباب العرض
    size.Height = e.Y - startp.Y 'حساب الطول
    EndPt.X = e.X
    EndPt.Y = e.Y

    Select Case Shap
        Case 1
            Dim rec As New Rectangle(startp.X, startp.Y, size.Width, size.Height)
            gr = Graphics.FromHwnd(PictureBox2.Handle)
            gr.DrawEllipse(Pens.Black, startp.X, startp.Y, size.Width, size.Height)
            PictureBox2.Invalidate(rec)

        Case 2
            Dim rec As New Rectangle(startp.X, startp.Y, size.Width, size.Height)
            gr = Graphics.FromHwnd(PictureBox2.Handle)
            gr.DrawRectangle(Pens.Black, startp.X, startp.Y, size.Width, size.Height)
            PictureBox2.Invalidate(rec)

        Case 6
            Dim rec As New Rectangle(startp, size)
            gr = Graphics.FromHwnd(PictureBox2.Handle)
            gr.DrawLine(Pens.Black, startp, EndPt)
            PictureBox2.Invalidate(rec)

    End Select
```

١

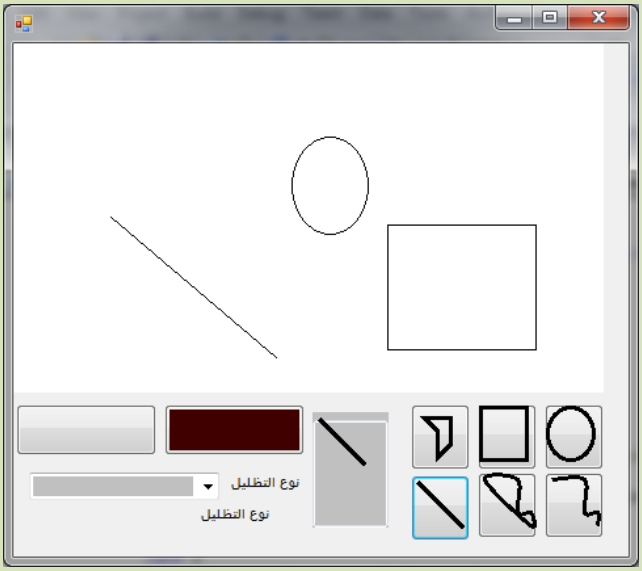
٢

في الرقم 1 قمنا باعطاء قيم للمتغير الذي يحدد النقطة النهاية للخط
في الرقم2 ستجد في الحالة Case 6 والتي ترسم خط مستقيم ولكن تلاحظ اننا كتبنا نفس الكود في الحالتين الاولى والثانية ولكن عدلنا فقط امر الرسم بحيث اننا نحدد ان يكون رسم للخط المستقيم و نحدد نقطة البداية و النهاية
وكذلك في الكود الخاص بالحدث mouse up كما في الشكل الاتي

```
Select Case Shap
    Case 1
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawEllipse(Pens.Black, startp.X, startp.Y, size.Width, size.Height)

    Case 2
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawRectangle(Pens.Black, startp.X, startp.Y, size.Width, size.Height)
    Case 6
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawLine(Pens.Black, startp, EndPt)

End Select
startDra = False
```

فد كتبنا الحالة تكون رقم 6 بحيث يسمح برسم الخط المستقيم ثم كتبنا نفس الكود في الحالتين ولكن عدلنا الكود بحيث يرسم خط مستقيم و حددنا نقطة البداية ونقطة النهاية

اذا فتحنا البرنامج تستطيع رسم المستطيل كما في الشكل الاتي



الى الان ستجد ان عملية الرسم سهلة ولكن سوف ننتقل لرسم المضلع ستجد اختلاف كثير عن الحالات السابقة لذا ارجو ان تركز معي لكي تستطيع رسم المضلح باستخدام الماوس

سنعلن في قسم الاعلانات العامة المتغيرين الاتيين

```
Dim points(50) As Point
Dim pointNo As Integer = 0
```

المتغير الاول points هو مصوفة ل 50 نقطة بحيث انها تجمع النقط التي سوف نتسخدمها في رسم المضلع

في حدث mouse Dawn سوف نعدل الكود بحيث يكون اذا نقرنا بالزر الايسر للماوس فانه يحدد النقط المطلوبة للرسم واذا نقرنا على الزر الايمن فانه يقوم برسم المضلع
ويتم ذلك من خلال الكود الاتي

```
If e.Button = MouseButtons.Left Then                    1
    startDra = True
    startp.X = e.X
    startp.Y = e.Y
    Select Case Shap
        Case 3
            points(pointNo).X = e.X
            points(pointNo).Y = e.Y
            pointNo += 1
            gr = Graphics.FromHwnd(PictureBox2.Handle)

    End Select                                          2
Else
    Select Case Shap

        Case 3
            Dim Mypoints(pointNo - 1) As Point
            Dim i As Integer = 0
            Do Until i = pointNo
                Mypoints(i) = points(i)
                i += 1
            Loop

                gr = Graphics.FromHwnd(PictureBox2.Handle)
                gr.DrawPolygon(New Pen(Button8.BackColor), Mypoints)
                pointNo = 0                             3


    End Select
End If
```

الكود المعلم بالرقم 1 هو جملة شرط تختبر اذا كان المستخدم ضغط على الزر الايمن للماوس او الزر الايسر

بينما المربع الذي يعلم بالرقم 2فبه الكود الاتي

جملة اختبار للمتغير Shap بحيث اذا كانا المتغير يحمل الرقم 3 فانه يتم فيه الاتي

<u>في السطر الاول و الثاني و الثالث</u> بعد كلمة Case 3 فانه عبارة عن ادخال الاحداثيات السنية و الصادية داخل احد عناصر المصفوفة و التي يتم تحديد رقمه من المتغير PointNo و المتغير بعد

ذلك في السطر الثالث يزيد بقيمة 1 بحيث يتم تحديد نقطة اخرى عندما ينقر المستخدم مرة اخر بالزر الايسر للماوس

ثم في السطر الرابع و الخامس يتم رسم النقطة في الموقع الذي تم النقر عليه وهي عبارة عن دائرة صغيرة بحيث العرض و الطول يساوي 1 بكسل وتنهي بذلك جملة الاختبار

في المربع المعلم بالرقم 3

سنعود لجملة الشرط التي تختبر اذا كان النقر بالزرالاسر فتجد كلمة Else التي تقول اذا لم يكن النقر بالزر الايسر فانه سوف يدخل في الجملة لاتية

يتم رسم المضلع على اساس النقط التي حددناها ولكن اذا استخدمنا المصوفة التي بها 50 عنصر سوف ترسم المضلع بشكل عشوائي لذا سنعلن عن مصفوفة بعدد النقط التي تم تحديدها ولكن طبعا مطروح منها 1 ليتم تحيدد اعلى عنصر في المصفوفة فكان الكود كلاتي

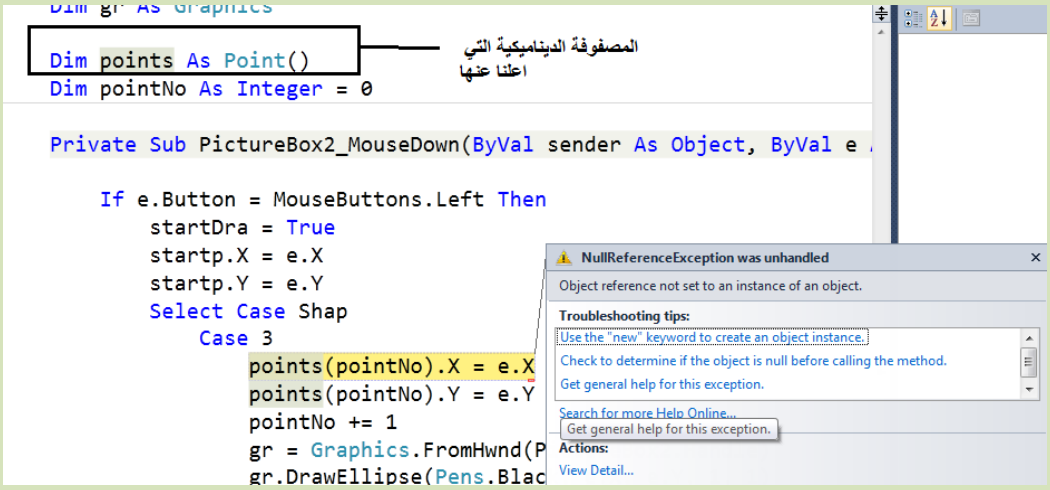
```
Dim Mypoints(PointNo-1)As point
```

بعد ذلك في نعلن عن متغير بقيمة صفر لاننا سوف ننشئ حلقة تكرارية لادخال النقاط التي تم تحديدها من المصوفة الاولى الى المصفوفة الثانية

بعد ان ادخلنا العناصر في الحلقة التكرارية نقوم برسم الضلع على اللوحة على اساس النقط التي حددها المستخدم

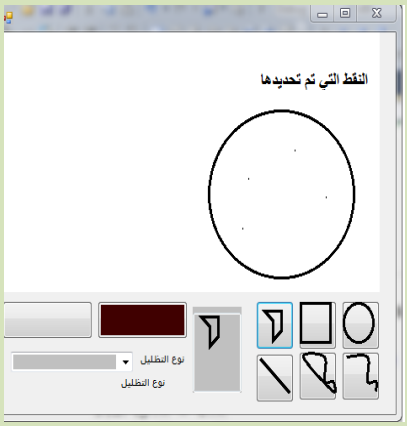
ملحوظة هامة

سبب تحديد 50 عنصر للمصفوفة الاولى وعدم تركها كمفصوفة دينميكية هو انه عندما تقوم بعمل مصفوفة ديناميكية ثم تخصص قيمة لاي عنصر فانه يقوم بعمل استثناء كما هو موضح في الشكل الاتي

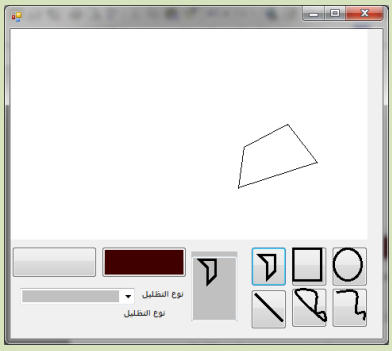


لذا اضطررنا لعمل مصفوفة بها عناصر افتراضية وليكن 51 عنصر ثم ادخلنا عليه العناصر كما اوضحنا في السطور السابقة ثم نريد ان نستخدم هذه العناصر وليكن 3 عناصر فقط لذا يجب انشاء مصفوفة ثانيه بها ال3 عناصر فقط ثم نستخدم هذه العناصر وهذا ما قمنا به عندما اعلنا عن المصفوفة Mypoint ثم ادخلنا عليها العناصر في الحلقة التكرارية .

فاذا شغلنا البرنامج وادخلنا النقاط كما هو موضح في الشكل الاتي



فاذا نقرنا بالزر الايمن سنجد الشكل مرسوم كما هو موضح في الشكل الاتي



بهذه الطريقة تم رسم المضلع ستجد نفس الطريقة في رسم المنحنى و المنحنى المغلق لذا سوف نعدل فقط في الحدث mouse Dawn بحيث سوف نعدل فقط جمل الاختبار بيحث يكون رسم المنحنى برقم 4 و المنحنى المغلق بالرقم 5 ويكون الكود كما هو وذلك في الجملة الاختبار بعد كلمة Else سندخل جملة اختبار للمتغير Shape بحيث يمكن اذا كان المتغير يساوي 3 يتم رسم المضلع و عندما يكون 4 نرسم المنحنى وعندما يكون 5 نرسم المنحنى المغلق

ويت توضيح ذلك في الكود الاتي

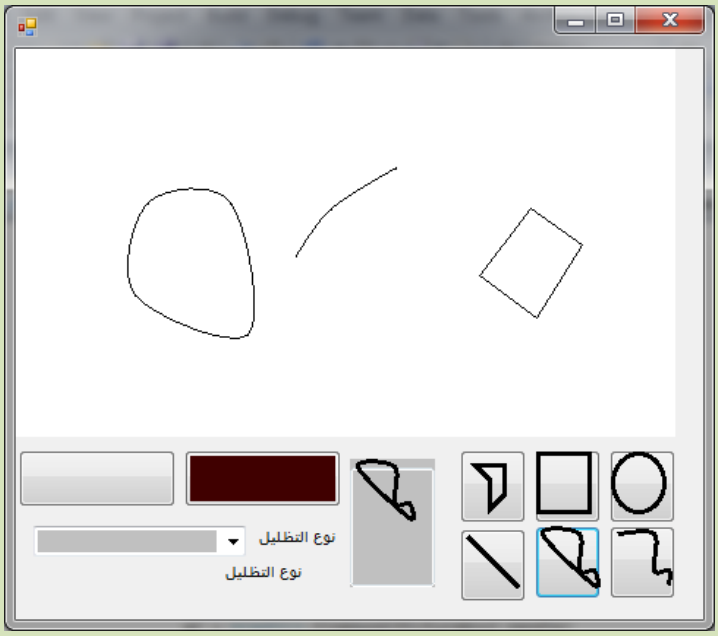
```
Select Case Shap
    Case 3
        points(pointNo).X = e.X
        points(pointNo).Y = e.Y
        pointNo += 1
        gr = Graphics.FromHwnd(PictureBox2.Handle)
    Case 4
        points(pointNo).X = e.X
        points(pointNo).Y = e.Y
        pointNo += 1
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawEllipse(Pens.Black, e.X, e.Y, 1, 1)
    Case 5
        points(pointNo).X = e.X
        points(pointNo).Y = e.Y
        pointNo += 1
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawEllipse(Pens.Black, e.X, e.Y, 1, 1)
End Select
```

الكود السابق عندما تم النقر بالزر الايسر للماوس

```
Select Case Shap

    Case 3
        Dim Mypoints(pointNo - 1) As Point
        Dim i As Integer = 0
        Do Until i = pointNo
            Mypoints(i) = points(i)
            i += 1
        Loop
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawPolygon(Pens.Black, Mypoints)
        pointNo = 0
```

```
        Case 4
            Dim Mypoints(pointNo - 1) As Point
            Dim i As Integer = 0
            Do Until i = pointNo
                Mypoints(i) = points(i)
                i += 1
            Loop
            gr = Graphics.FromHwnd(PictureBox2.Handle)
            gr.DrawCurve(Pens.Black, Mypoints)
            pointNo = 0
        Case 5
            Dim Mypoints(pointNo - 1) As Point
            Dim i As Integer = 0
            Do Until i = pointNo
                Mypoints(i) = points(i)
                i += 1
            Loop
            gr = Graphics.FromHwnd(PictureBox2.Handle)
            gr.DrawClosedCurve(Pens.Black, Mypoints)
            pointNo = 0
    End Select
```

وعند تشغيل البرنامج ننختار المنحنى و المنحنى المغلق و المضلع سنجد الرسم موضح كلاتي



وبذلك نكون قد انتهينا من رسم المضلعات ولكن هل يجب ان يكون الرسم باللون الاسود .... طبعا لا لذا فاننا سنعدل اللون الاسود الى اللون الذي يتم اختياره من مربع الالوان الاول سنعدل اوامر الرسم بحيث يكون مكان كلمة pen.Black الى كلمة

المرحلة السادسة تعديل الرسم بمكن رسمه باللون الذي نريده

```
New pen(Button8.backColo)
```

ويتم تعديل هذه الكلمة على كل اوامر الرسم التي تخص كل الاشكال بحيث
تحدد هذه الكود في الحدث mouseUp للمستطيل و البيضاويو الخط المستقيم
في الحدث mouse Dawn في رسم الاشكال المنحنى و المنحنى المغلق و المضلع
فيتم وضع الكود السابق مكان الكود الذي يحل في مكان القبل
مثل سنعدل الكود الخاص برسم الخط من الكود

```
gr.DrawLine(  Pen .Black startp, EndPt)
```

الى

```
gr.DrawLine(New Pen(Button8.BackColor), startp, EndPt)
```

ولكي تستطيع فهم ما قصدت ساضع الكود الخاص بالرسم وساعلم على الاماكن التي سوف نعدل فيها الكود

الكود الاتي عند رسم المضلع و المنحنى و المنحنى المغلق

```
Select Case Snap

    Case 3
        Dim Mypoints(pointNo - 1) As Point
        Dim i As Integer = 0
        Do Until i = pointNo
            Mypoints(i) = points(i)
            i += 1
        Loop
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawPolygon(New Pen(Button8.BackColor), Mypoints)
        pointNo = 0
```

```
    Case 4
        Dim Mypoints(pointNo - 1) As Point
        Dim i As Integer = 0
        Do Until i = pointNo
            Mypoints(i) = points(i)
            i += 1
        Loop
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawCurve(New Pen(Button8.BackColor), Mypoints)
        pointNo = 0
```

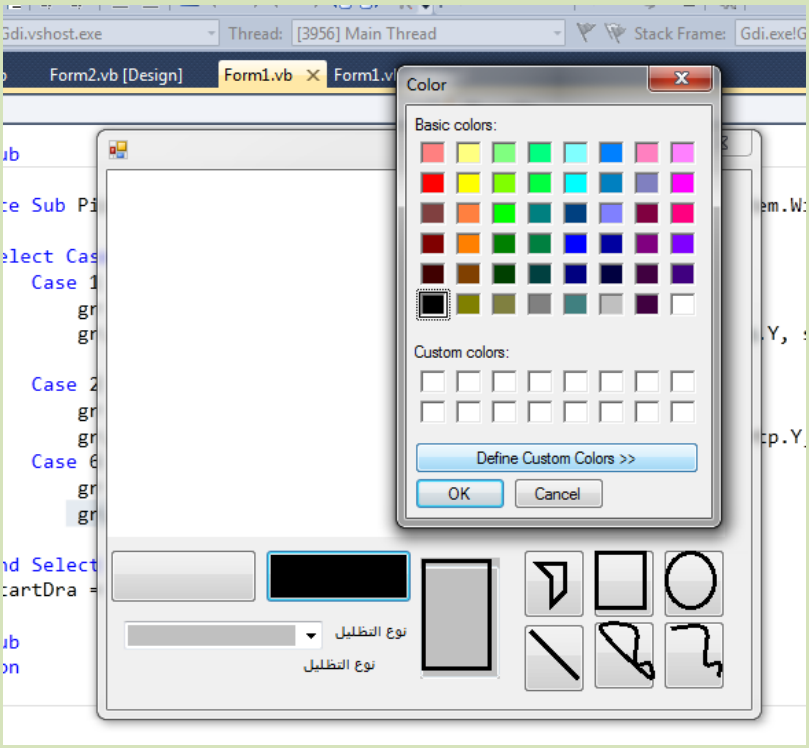
```
    Case 5
        Dim Mypoints(pointNo - 1) As Point
        Dim i As Integer = 0
        Do Until i = pointNo
            Mypoints(i) = points(i)
            i += 1
        Loop
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawClosedCurve(New Pen(Button8.BackColor), Mypoints)
        pointNo = 0
End Select
```

الكود الذي تحته خط هو الكود المعدل

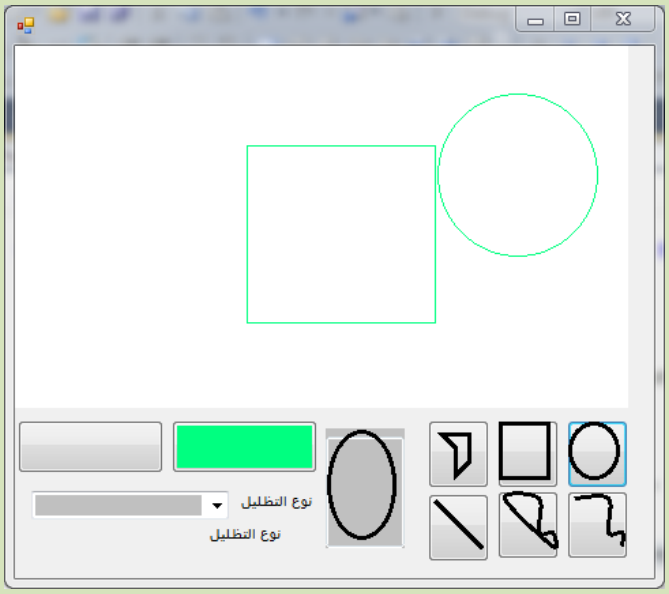
ستجد ذلك ايضا في الكود الخاص برسم المستطيل و البيضاوي و الخط في الحدث mouse Up

```
Select Case Shap
    Case 1
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawEllipse(New Pen(Button8.BackColor), startp.X, startp.Y, size.Width, size.Height)

    Case 2
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawRectangle(New Pen(Button8.BackColor), startp.X, startp.Y, size.Width, size.Height)
    Case 6
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawLine(New Pen(Button8.BackColor), startp, EndPt)

End Select
startDra = False
```

وعند تشغيل البرنامج ستجد انك تستطيع تغيير لون محيط أي شكل تريد ان ترسمه



في الشكل السابق نحدد اللون الذي نريد وليكن لونه اخضر ثم نرسم الشكل الذي نريده فتلاحظ ان اللون تغير



سننتقل الان الى مرحلة جديدة من تصميم برنامج الرسم وهي تظليل الاشكال بحيث لا يكون محيط الشكل هو الملون والشكل لونه ابيض نريد مستطيل مظلل بلون ازرق مثلا لذا يجب تعديل الكود ليتغير لونه بعد الرسم

المرحلة السابعة وهي تظليل الاشكال المرسومة

سنقوم الان بكتاب اجراء فرعي اسمه DrawShape بحيث اننا سنضيف اليه اكواد التلوين كلها

اولا تذكر اننا قمنا بعمل كومبو بوكس ليحدد نوع الرسم الذي نريده لذا سوف نعلن اولا عن متغير من نوع فرشاه Brush في قسم الالعانات العامة

```
Dim Brush As Brush
```

ونعلن عن متغير يحدد اذا كنا نستخدم الفرشاه ام لا

```
Dim IsBrusth As Boolean
```

ثم نقوم كتابة الكود الاتي في الكومبو بكس ليتم تحديد شكل طريقة التظليل ويكون الكود كلاتي

```
Select Case ComboBox1.SelectedIndex
    Case 0
        Dim sol As New SolidBrush(Button8.BackColor)
        Brush = sol
        IsBrusth = True
    Case 1
        Dim rec As New Rectangle(0, 0, 100, 100)
        Dim c1 As Color = Button8.BackColor
        Dim c2 As Color = Button9.BackColor
        Dim lin As New LinearGradientBrush(rec, c1, c2, 0)
        Brush = lin
        IsBrusth = True
    Case 2
        IsBrusth = False
End Select
```

في الكود السابق قمنا بكتابة جملة اختبار حالة اختيار الكمبو بكس

<u>في الحالة رقم 0 يكون</u> التظليل بلون واحد لذا يكون نوع الفرشاه المستخدمة في التظليل هو الفرشاه solideBrush وهي تستخدم بالتظليل بلون واحد كما هو في الكود الاتي

```
Case 0
    Dim sol As New SolidBrush(Button8.BackColor)
    Brush = sol
    IsBrusth = True
```

هذا الكود هو نفس الكود الذي كتبناه في الكومبوا بوكس ولكن قمت بتجزئه الكود ليسهل عليك فهمه

<u>في السطر الاول</u> اعلنا عن متغير ينشاء الفرشاه ذات نوع SolidBrush واعطينا اللون الموجود في مربع الالوان الاول Button8 ثم نعطي للمتغير Brusth الفرشاه التي انشانها ثم نغير

المتغير isBrush ليخبر الكمبيوتر اننا نريد استخدام الفرشاه و لا نريد استخدام القلم في حالة رسم الاشكل كما سنوضح عندما نريد التظليل

<u>في الحالة رقم 1 يكون</u> التظليل بلونين لذا يكون نوع الفرشاه المستخدمة في التظليل هو الفرشاه linesrGradintBrush وهي تستخدم بالتظليل بلونين كما هو في الكود الاتي

```
Case 1
    Dim rec As New Rectangle(0, 0, 100, 100)
    Dim c1 As Color = Button8.BackColor
    Dim c2 As Color = Button9.BackColor
    Dim lin As New LinearGradientBrush(rec, c1, c2, 0)
    Brush = lin
    IsBrusth = True
```

في الحالة الاخيرة يتم الغاء استخدام الفرشاه بحيث انه لا يسمح بايتخدام الفرشاة ويتم استخدام القلم كما هو موضح في الكود الاتي

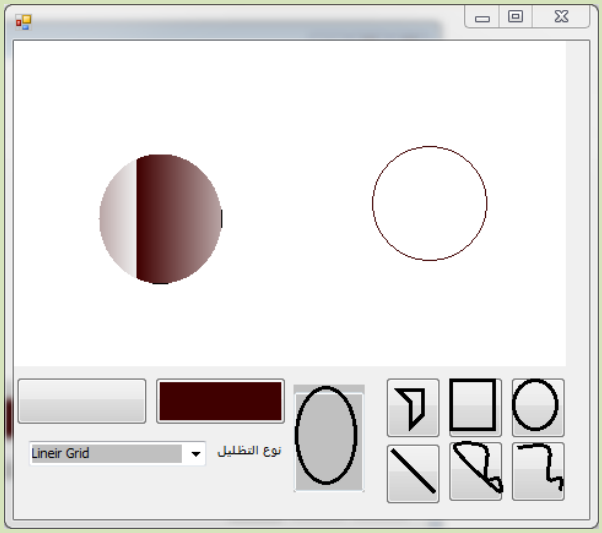
```
Case 2
    IsBrusth = False
```

ننتقل الان في تعديل الاكواد التي تخص الرسم في الحدثين , mous Dawn و mouse Up
وسوف نبدا بالحدث mouseDawn
سنقوم بتعديل الكود الخاص برسم الدائرة حتى يكون في الشكل الاتي

```
Select Case Shap
   Case 1
      If IsBrusth Then
         gr = Graphics.FromHwnd(PictureBox2.Handle)
         gr.FillEllipse(Brush, startp.X, startp.Y, size.Width, size.Height)

      Else

         gr = Graphics.FromHwnd(PictureBox2.Handle)
         gr.DrawEllipse(New Pen(Button8.BackColor), startp.X, startp.Y, size.Width, size.Height)
      End If
```

كما تلاحظ اننا قمنا في الحالة الاولى Case 1 حيث تحتها نقوم برسم الدائرة فقمنا بوضح جملة شرط بحيث انه اذا كان isBrush يساوي True فاننا نقوم برسم الدائرة مملوئة بالفرشاة كما وضحنا في الكود الخاص باعداد الفرشاه واذا كان المتغير isBrush يساوي False فاننا نرسم الدائة مفرغة

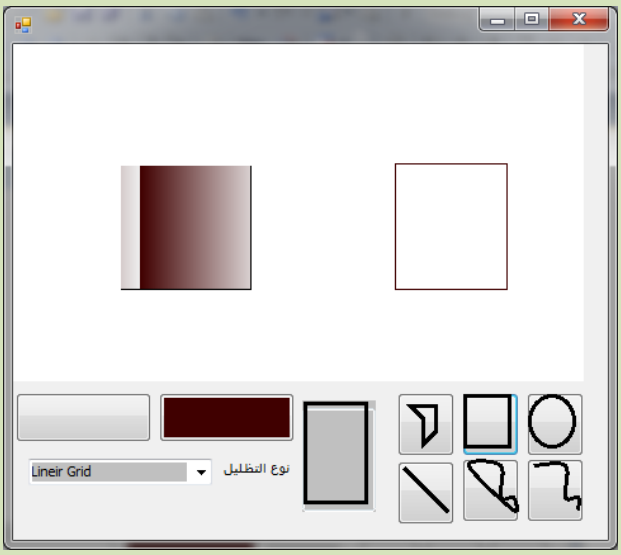
قم بتشغيل البرنامج وعدل الكومبو بكس الى التظليل الخطي Lineir Grid وارسم دائرة فسوف تجدها تظهر في الشكل الاتي



نبدا الان بتعديل الكود الخاص برسم المربع Case 2 كما هو موضح في الكود الاتي

```
Case 2
    If IsBrusth Then
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.FillRectangle(Brush, startp.X, startp.Y, size.Width, size.Height)

    Else
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawRectangle(New Pen(Button8.BackColor), startp.X, startp.Y, size.Width, size.Height)
    End If
```

قم بتشغيل البرنامج ستجد البرنامج يظهر كلاتي



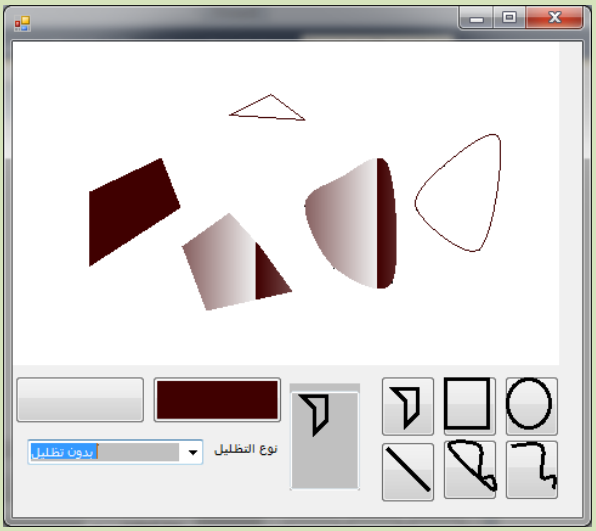
طبعا لا نقدر على التعديل في Case 6 لانها ترسم خط فقط

سننتقل الان الى mouse Up لكي يتم عمل تظليل للشكل المضلع و المنحنى المغلق فقط

فننتقل الى مرحلة الرسم التي تقوم برسم المضلع ورسم المنحنى المغلق ونقوم بتعديل الكود بنفس الطريقة فيكون الكود كلاتي

```
Case 3
    Dim Mypoints(pointNo - 1) As Point
    Dim i As Integer = 0
    Do Until i = pointNo
        Mypoints(i) = points(i)
        i += 1
    Loop
    If IsBrusth Then
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.FillPolygon(Brush, Mypoints)
        pointNo = 0
    Else
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawPolygon(New Pen(Button8.BackColor), Mypoints)
        pointNo = 0
    End If
Case 4
    Dim Mypoints(pointNo - 1) As Point
    Dim i As Integer = 0
    Do Until i = pointNo
        Mypoints(i) = points(i)
        i += 1
    Loop
    gr = Graphics.FromHwnd(PictureBox2.Handle)
    gr.DrawCurve(New Pen(Button8.BackColor), Mypoints)
    pointNo = 0
Case 5
    Dim Mypoints(pointNo - 1) As Point
    Dim i As Integer = 0
    Do Until i = pointNo
        Mypoints(i) = points(i)
        i += 1
    Loop
    If IsBrusth Then
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.FillClosedCurve(Brush, Mypoints)
        pointNo = 0
    Else
        gr = Graphics.FromHwnd(PictureBox2.Handle)
        gr.DrawClosedCurve(New Pen(Button8.BackColor), Mypoints)
        pointNo = 0
    End If
```

تلاحظ ان الكود المعلم عليه بالمربع هو الكود الذي كتبناه وعدلنا فيه
وعند تشغيل البرنامج ستجد انها تظهر على الشكل الاتي



و الى هنا انتهينا من اعطاء نبذه عن عمل برنامج رسام ننتقل الان الى جزء ثاني من تطبيقات على استخدام GDI فبالطبع ستجد ان البرنامج فيه بعض العيوب حاول ايجاد العيوب فيها وتعديلها لن تجد فيها صعوبة سننتلق الان الى تحريك الصور

تحريك الرسوم

تعلمنا في الفصل السابق التصنيف matrix الذي يقوم بعمل بعض التعديلات في الرسم كنقل الرسم او تدوير الصورة وغيرها من التغييرات
سنستفيد من هذا التصنيف في عمل بعض التطبيقات من اهمها تحريك الرسمات التي ستعجب بها كثيرا لانك ستجد نفسك تستطيع عمل بعض العروض من خلالها كما تقوم بعمل ذلك من البرنامج power point الشهير وسوف نعطيك بعض الحركات التي الاساسية في عملية تحريك الرسمات وسوف نستعرض ذلك في الامثلة الاتية

المثال الاول التحريك الافقي و الراسي للرسم

سنقوم بعمل مشروع جديد ونضيف اليه الاده Timer وسوف ندخل الكود داخل الحدث Tick الخاص بالمؤقت Timer الاتي

```
1  Dim gr As Graphics = Graphics.FromHwnd(Me.Handle)
2  gr.Clear(Me.BackColor)
3  Static x, y As Integer
4  x += 1
5  y += 1
6  Dim mm As New Matrix
7  mm.Translate(x, y)
8  gr.Transform = mm
9  gr.FillRectangle(Brushes.Blue, 0, 0, 100, 100)
10 gr.ResetTransform()
```

يجب ان نكرر هنا اننا يجب ان نقوم باستيراد المكتبة Syatem.Draw و System.Darw.Draw2D

في السطر الاول اعلنا عن متغير من نوع Graphic بحيث يمكننا رسم المربع على الفورم
في السطر الثاني قمنا بعمل مسح للنموذج وسوف تتعرف على اهميته في المثال القادم
من السطر الثالث للسطر الخامس سوف نقوم باعلان متغيرين يحددان موقع النقاط التي تحدد مكان المربع الذي سنرسمه وطبعا المتغيرين استاتيكيين Static لاننا نريد ان تتغير قيمتهما كلما استدعينا الاجراء
في السطر السادس نعلن عن متغير من نوع matrix
في السطر السابع و الثامن استدعينا الدالة Translat وقمنا باعطاء معاملاتها المتغيري X,y لانهم يتغيران كل مرة يتم استدعاء الكود ثم اعطينا قيمة المتغير mm الذي هو من نوع matrix الى المتغير gr الذي من نوع graphics من الخاصية transform وقد شرحنا ذلك في الفصل السابق في قسم matrix
في السطر تاسع نقوم برسم المربع
في السطر الاخير نمسح قيمة المتغير mm المتغير Gr بالخاصية restTrsnform

اذا شغلت البرنامج ستلاحظ ان المربع يتحرك بشكل قطري من اعلى يسار الفورم لاسفل يمين الفورم واذا اردت ان يتحرك من اليمين لليسرا ستعطي للمتغير y الذي يحدد الوضع الافقي للمستطيل قيمة ثابتة بحيث ان تلغي الكود المسئول عن اضافة رقم 1 للمتغير والكود الذي ستلغيه هو y+=1
وكذلك يكون نفس الكلام في المتغير x اذا اردت التحريك من اعلى لاسفل

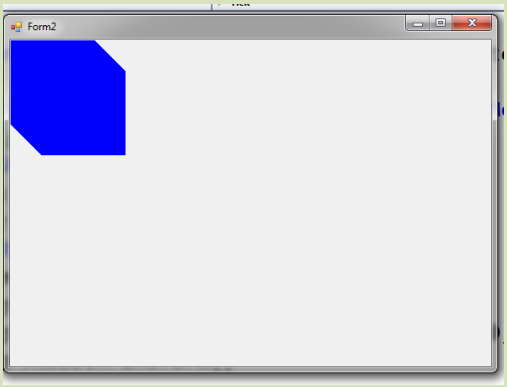
المثال الثاني تكبير حجم المكعب

سوف نستخدم نفس الكود في المثال السابق لكن سنقوم بمسح السطر الثاني منه الخاص بـGr.Clear
كما هو موضح في الكود الاتي

```
Dim gr As Graphics = Graphics.FromHwnd(Me.Handle)
'gr.Clear(Me.BackColor)
Static x, y As Integer
x += 1
y += 1
Dim mm As New Matrix
mm.Translate(x, y)
gr.Transform = mm
gr.FillRectangle(Brushes.Blue, 0, 0, 100, 100)
gr.ResetTransform()
```

اوقفنا عمل الكود

قلنا في المثال السبق اننا سنوضح اهمية السطر الثاني في هذا المثال قم بتشغيل البرنامج ستلاحظ ان النموذج يقوم بتكررا رسم المربعات بدون ان يقوم بمسح المربع السابق فيظهر على شكل مكعب كما في الشكل الاتي



المثال الثالث تدوير المربع حول نفسه

تعلمناها في التصنيف matrix لذا سوف ناخذ الكود في المثال الاول وسون عدل عليه بعض الاكود كما هو موضح في الشكل الاتي

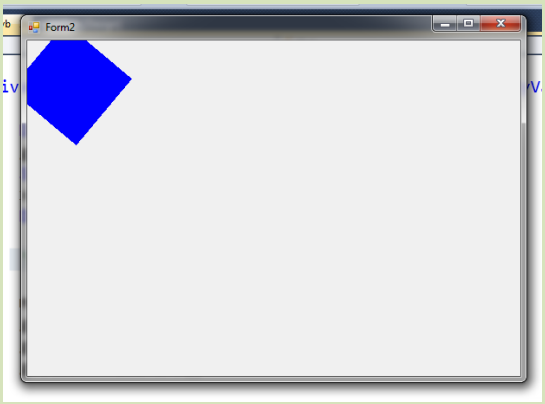
```
Dim gr As Graphics = Graphics.FromHwnd(Me.Handle)
gr.Clear(Me.BackColor)
Static x As Integer
x += 10
Dim mm As New Matrix
mm.RotateAt(x, New Point(50, 50))
gr.Transform = mm
gr.FillRectangle(Brushes.Blue, 0, 0, 100, 100)
gr.ResetTransform()
```

الكود المعدل

في الكود المعدل تلاحظ اننا قمنا بحذف المتغير y لاننا سنحتاج فقط للمتغير X لحساب الزاوية التي يتم تدوير المرع عليها

اما في الكود في السطر الاخير في المربع الذي يقول mm.Rotatat()

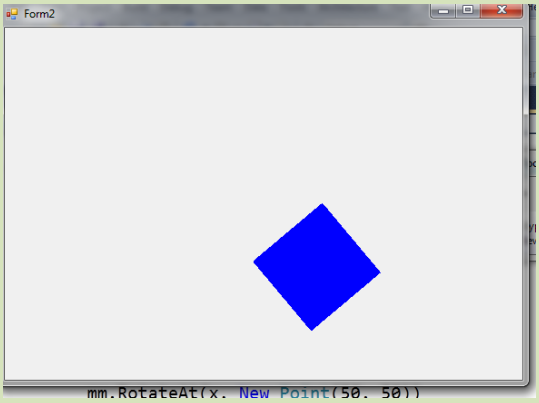
هنا نقوم بعمل المعاملين معامل يعدل الزاوي ومعامل يحدد نقطة محور الارتكاز الذي يدور حوله المربع ووضعنها في المنتصف حيث ان عرض و طول المربع يساوي 100 فان النقطة ستقع على المنتصف في النقطة (50,50) وسيظهر البرنامج عند تشغيله كما هو في الشكل الاتي



واحب ان اوضح انك اذا نقلت المربع بحيث غيرت النقطة التي تحدد مكان المربع من (0,0) الى (10,10) من مكان لاخر فان المربع لن يدول حول نفسه وانما يدور حول النقطة (50,50) و يجب علينا لو اردنا ان ننقل المربع سوف نضيف الكود المسؤول عن نقل المربع من خلال التصنيف matrix كما هو موضع في الكود الاتي

```
Dim gr As Graphics = Graphics.FromHwnd(Me.Handle)
gr.Clear(Me.BackColor)
Static x As Integer
x += 10
Dim mm As New Matrix

mm.Translate(Me.Width / 2, Me.Height / 2)

mm.RotateAt(x, New Point(50, 50))
gr.Transform = mm
gr.FillRectangle(Brushes.Blue, 0, 0, 100, 100)
gr.ResetTransform()
```

اذا شغلنا البرنامج ستجد المربع يدور حول نفسه



المثال الرابع عمل ضهور لصورة من وراء مربع

قم بمسح الكود السباق كله وادخل الكود الاتي

```
1 Dim g As Graphics = Graphics.FromHwnd(Me.
2 g.Clear(Me.BackColor)
3 Dim img As Image = My.Resources._5
4 Static h As Integer = 100
5 Static w As Integer = 100
6 h -= 1
7 w -= 1
8 Dim m As New Matrix
9 m.Translate(Me.Width / 2, Me.Height / 2)
10 g.Transform = m
11 Dim rec As New Rectangle(0, 0, w, h)
12 g.DrawImage(img, 0, 0, 100, 100)
13 g.FillRectangle(Brushes.Black, rec)
14 g.ResetTransform()
```

في السطر الاول و الثاني اعلنا عن متغير المسئول عن الرسم من نوع Graphic وقمنا باعطاء امر المسح في السطر الثاني كما اوضحنا في الامثلة السابقة

في السطر الثالث اعلنا عن متغير من نوع image بحيث انه ياخذ الصورة وقد شرحنا هذا الكود في فصل سابق

من السطر الرابع الى السطر السابع اعلنا عن متغيرين يحددان الطول والعرض للمربع الذي سوف يصغر وتظهر الصورة من ورائه

من السطر الثامن للسطر العاشر قمنا بنقل الصورة الى منتصف الفورم باستخدام التصمنيف matrix

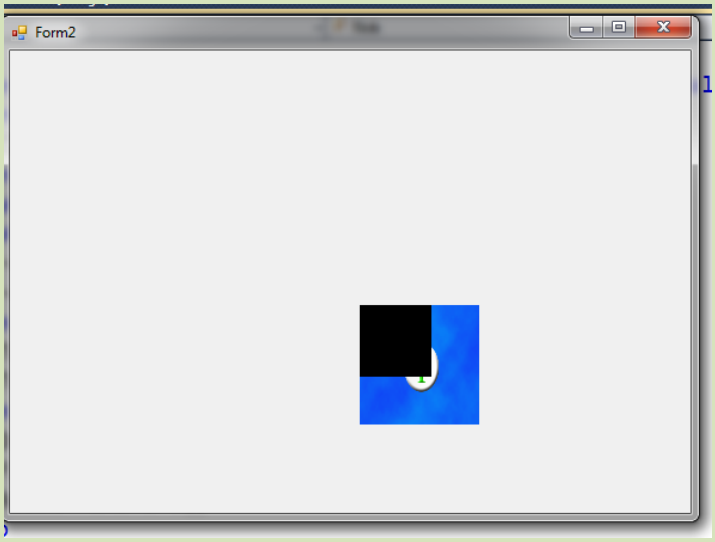
في السطر الحادي عشر اعلنا عن متغير ينشئ كائن المربع الذي سوف يتم تصغيره

في السطر الثاني عشر قمنا برسم الصورة

في السطر الثالث عشر قمنا برسم المربع الذي سوف يتم تصغيره

في السطر الاخير اعدنا rest Transform للمتغير G

قم بتشغيل البرنامج سوف تلاحظ الاتي



الى هنا قد اخذنا نبذة مختصرة عن طريقة تحريك الرسوم سننتقل الى تطبيق اخر

تنسيق النصوص

في الامثلة القادمة سنقوم بعمل تنسيق للنصوص بحيث يمكننا عمل برامج تحتوي على نصوص مجسمة او نصوص بها ظل مما يعطي لراجهة البرنامج جمال وجاذبية

المثال الاول سوف نشرح كيف تقوم بعمل نصوص بها ظل

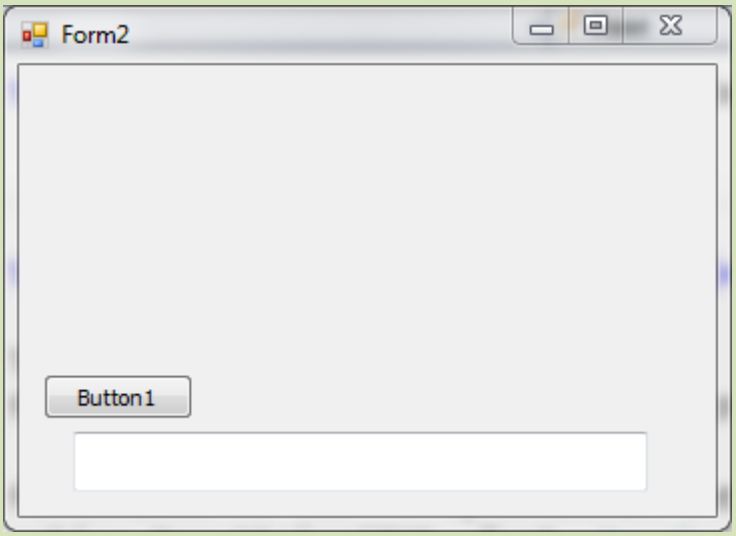
1. افتح مشروع جديد واضف اليه صندوق نصوص و رز
2. قم بكتابة الكود الاتي في الحدث Form1_Paint

```
1   Dim TXt As String - TextBox1.Text
2   Dim font As New Font(TextBox1.Font.FontFamily.Name, TextBox1.Font.Size + 2)

3   Dim fontShadow As New Font(TextBox1.Font.FontFamily.Name, TextBox1.Font.Size)
4   e.Graphics.DrawString(TXt, font, Brushes.Blue, Me.Width / 2, 100)
5   Dim m As New Matrix
6   m.Translate(8, 11)
7   e.Graphics.Transform = m
8   e.Graphics.DrawString(TXt, fontShadow, Brushes.Black, Me.Width / 2, 100)
9   e.Graphics.ResetTransform()
```

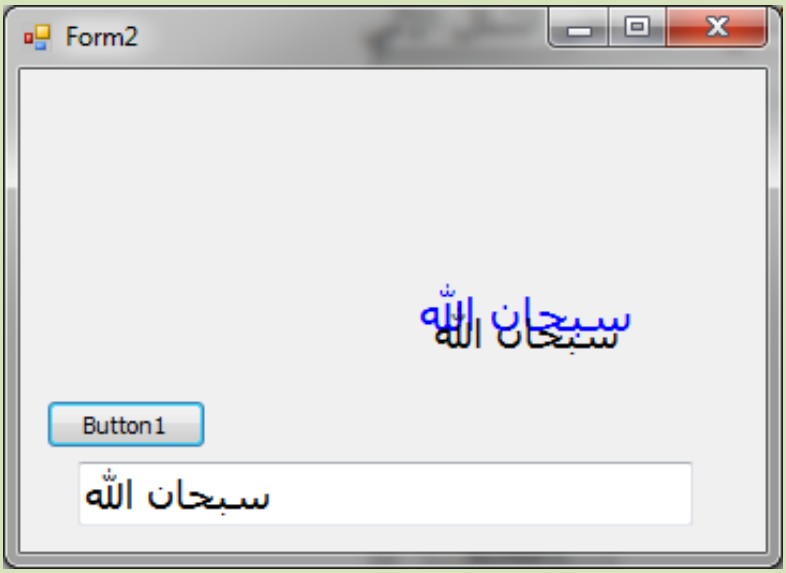
3- واكتب الكود الاتي في Button1_Click

```
Me.Refresh()
```

قم بتشغيل البرنامج فسوف تجد البرنامج يظهر على الشكل الاتي



اكتب النص سبحان الله واضغط على الزر فستجد النص يظهر كلاتي



سوف نشرح الكود الذي كبناه في الحدث Form1_paint

في السطر الاول اعلنا عن متغير يأخذ النص من الخاصية Text للاداة textBox1
في السطرين الثاني و الثالث اعلنا عن متغيرين يقوما بانشاء كائن للتصنيف Font فالمتغير الاول يمثل النص الاساسي و المتغير الثاني يمثل الظل و قمنا باضافة قيم للمعاملين بحيث يكون حجم النص الاساسي يزيد عن النص في الاداة TextBox1 بمدرجتين بينام المتغير الثاني يساوي النص العادي في الاداة textBox1

في السطر الرابع نقوم برسم النص الاساسي بحيث فنقوم باضافة نصوص من المتغير txt ومعامل الخط من المتغير font وعدلنا اللون الى الازق وحددنا موقع النص على الفورم في المعاملين الاخيرين
في السطر الخامس والسادس و السابع نقوم باعلان عن متغير m ينشئ الكائن matrix ثم اعدلنا الدالة Tranlate بحيث ينتقل يبعد الظل عن النص بمقدار يجعله يظهر على انه ظل ثم اعطينها للمتغير e.Graphic
في السطرين الاخيرين رسمنا الظل و اعدنا حالة الرسم كما كانت عليهe.grphic.restTraform

اما الكود المكتوب عند الضغط على الزر Botton1_Click كان الكود الاجراء Refresh()حيث يعيد رسم النموذج بعد التعديل النص على الاداة TextBox

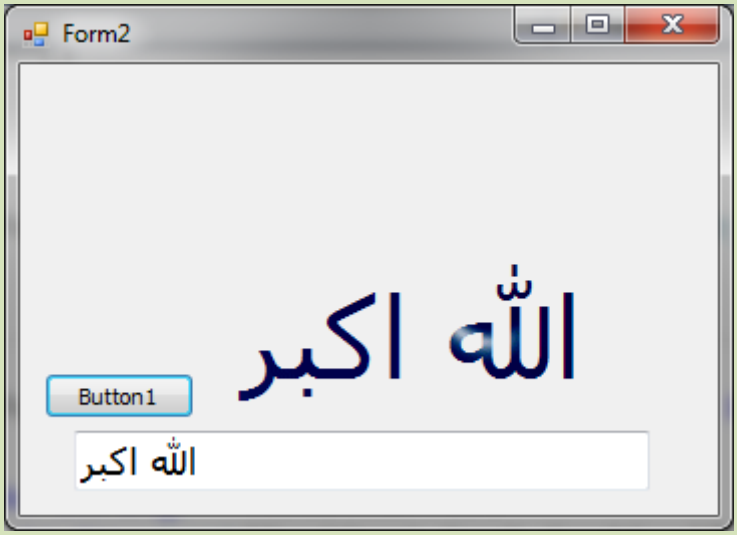
المثال الثاني سوف نشرح كيف تقوم بوضع صورة داخل

من نفس المثال السبق سنعد الكود في الحدث Form_paint

```
Dim TXt As String = TextBox1.Text
Dim font As New Font(TextBox1.Font.FontFamily.Name, TextBox1.Font.Size * 3)
Dim fontShadow As New Font(TextBox1.Font.FontFamily.Name, TextBox1.Font.Size)
Dim b As New TextureBrush(My.Resources._5)
e.Graphics.DrawString(TXt, font, b, 100, 100)
```

في الكود السابق ستجد نفسك تستطيع فهمه لكن في السطر الرابع سوف نتعرف على المتغير الذي ينشئ كائن من نوع texture التي نستطيع من خلالها ادخل صورة
وفي السطر الخامس نقوم بعمل رسم للنص وفي المعامل الخاص باللون نعطيه المتغير b الذي يحمل الصورة

افتح البرنامج واكتب كلمة الله اكبر على صندوق النص واضغط الزر فسوف تجد النص يظهر على الشكل الاتي



الى هنا نكون قد تعرفنا اهم التطبيقات الخاصة بالرسم بالغة الفيجوال بيسك بحيث اننا طبقنا معظم ما تعلمنا

سننتقل في الفصل القادم الى نقطة جديدة في تطبيقاتنا على

الرسم وهي التعامل مع ادوات التحكم

ملخص الفصل السادس

- عمل برنامج شبيه ببرنامج الرسام
- تحريك الصور بالشكل الطولي و العرضي و الدائري
- تعديل تنسيقات النصوص بحيث يكون له ظل او تغير لونه بحيث تظهر صورة عليه

تمرين للقرئ

قم بعمل برنامج يقوم برسم النص الاتي ويكون به طل والنص هو

"كلمتان خفيفتان على اللسان ثقيلتان في الميزان حبيبتان الى الرحمن سبحان الله وبحمده سبحان الله العظيم"

تلميح

ستجد الحل في الفصول السابقة راجع كل التمارين التي قمنا بالتدريب عليها والتي تتكلم عن التعامل مع النصوص مع العلم ان الحل سيكون موجود بين السطور ولكن ساعطيك مساعدة بسيطة فقط وهي انك يجب ان تضع النص على مربع من نوع RectangleF

الفصل السابع
تطوير ادوات التحكم

بعد كل ما تعلمناه في الفصول السابقة دعنا الان نطبق تطبيقات عملية على كل ما تعمناه حتى تتعرف كيفية استخدام مكتبة الرسم على ارض الواقع ففي هدا الفصل سوف اشرح لك بعض الامثلة العملية التي تضعك على اول الطريق في استخدام هذه المكتبة وهذا الفصل ما هو الا امتداد للفصل السابق

في نهاية الفصل ستكون قادر على

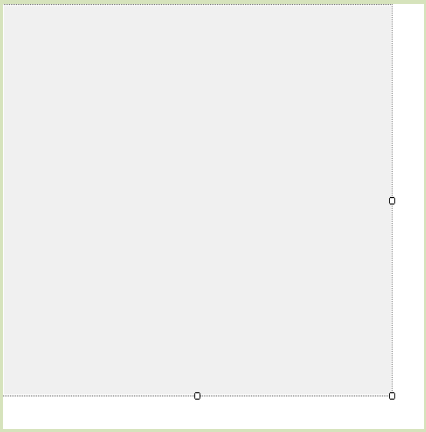
- كيف نغير شكل الفورم
- تطوير شكل الزر بحيث يتم يتغير شكله ولونه

تعلمنا في الفصول السابقة كيف نغير شكل الزر بحيث نجعله دائري ونرسم عليه ففي هذا الفصل سوف نتناول التطبيق بشكل اوسع

تغيير شكل الفورم

المثال الاول تغيير شكل الفورم الى دائرة

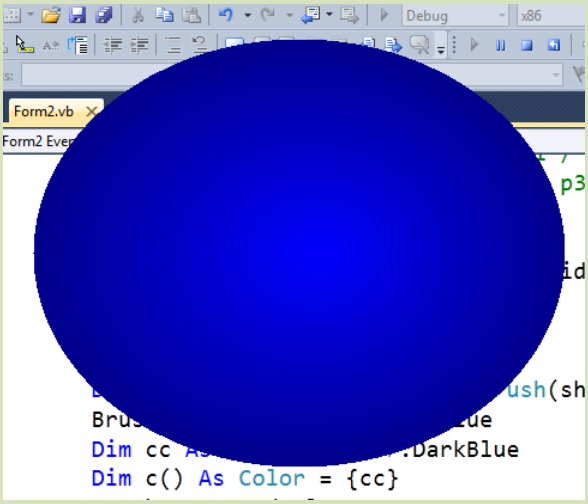
قم بفتح مشروع جديد وقم تغيير خاصية `FormBorderStyle` الى `None` لكي يظهر هكذا



ونقوم بعمل كود في الحدث `Form1_paint` ليكون كلاتي

```
1  Dim rec As New Rectangle(0, 0, Me.Width, Me.Height)
2  Dim shape As New GraphicsPath
3  shape.AddEllipse(rec)

4  Dim Brush As New PathGradientBrush(shape)
5  Brush.CenterColor = Color.Blue
6  Dim cc As Color = Color.DarkBlue
7  Dim c() As Color = {cc}
8  Brush.SurroundColors = c
9  e.Graphics.FillEllipse(Brush, rec)
10 Me.Region = New Region(shape)
```

عند تشغيل البرنامج ستجد الفورم تظهر كلاتي



في الكود الذي كتبنا ستجد اننا لم ندخل أي جديد ستجد نفس الكود الذي تعلمناه طوال رحلتنا في هذا الكتاب نستخدمه في تغيير شكل الفورم

ففي السطر الاول اعلنا عن متغير يقوم بانشاء مستطيل يبطول وعرص الفورم

من السطر الثاني الى السطر الثامن قمنا بانشاء فرشاه من نوع Path Gradiang Brush التي تهتم بالاشكال ففي هذه الاسطر قمنا باعداد الفرشاة لتكون شكل داائرة داخل الفورم

في السطر التاسع قمنا برسم شكل بيضاوي مظلل بالفرشاه داخل النمودج بحيث يكون طوله وعرضه نفس طول وعرص الفورم

في السطر الاخير استخدمنا خاصية Region التي تعتبر مبثابة مقص يقوم بقص الفورم على شكل الدائرة التي ادخلنها على المتغير Shape

المثال الثاني تغيير شكل الفورم الى نجمة

سوف نقوم فقط باضافة كود ثاني على الكود السابق الذي ادخلنا ولن نغير شيء الى سطر واحد سنوضحه بعد ان نظيف الكود الاتي للكود السابق

```
Dim p1 As New Point(0, Me.Height * 3 / 6)
Dim p2 As New Point(Me.Width * 1 / 6, Me.Height * 2 / 6)
Dim p3 As New Point(Me.Width * 1 / 6, Me.Height * 1 / 6)
Dim p4 As New Point(Me.Width * 2 / 6, Me.Height * 1 / 6)
Dim p5 As New Point(Me.Width * 3 / 6, 0)
Dim p6 As New Point(Me.Width * 4 / 6, Me.Height * 1 / 6)
Dim p7 As New Point(Me.Width * 5 / 6, Me.Height * 1 / 6)
Dim p8 As New Point(Me.Width * 5 / 6, Me.Height * 2 / 6)
Dim p9 As New Point((Me.Width * 6 / 6), Me.Height * 3 / 6)
Dim p10 As New Point(Me.Width * 5 / 6, Me.Height * 4 / 6)
Dim p11 As New Point(Me.Width * 5 / 6, Me.Height * 5 / 6)
Dim p12 As New Point(Me.Width * 4 / 6, Me.Height * 5 / 6)
Dim p13 As New Point(Me.Width * 3 / 6, (Me.Height - 6 / 6))
Dim p14 As New Point(Me.Width * 2 / 6, Me.Height * 5 / 6)
Dim p15 As New Point(Me.Width * 1 / 6, Me.Height * 5 / 6)
Dim p16 As New Point(Me.Width * 1 / 6, Me.Height * 4 / 6)
Dim pf As PointF() = {p1, p1, p2, p3, p4, p5, p6, p7, p8, p9, p10, p11, p12, p13, p14, p15, p16}
'رسم الشكل
```

1

```
Dim rec As New Rectangle(0, 0, Me.Width, Me.Height)
Dim shape As New GraphicsPath
shape.AddPolygon(pf)
```

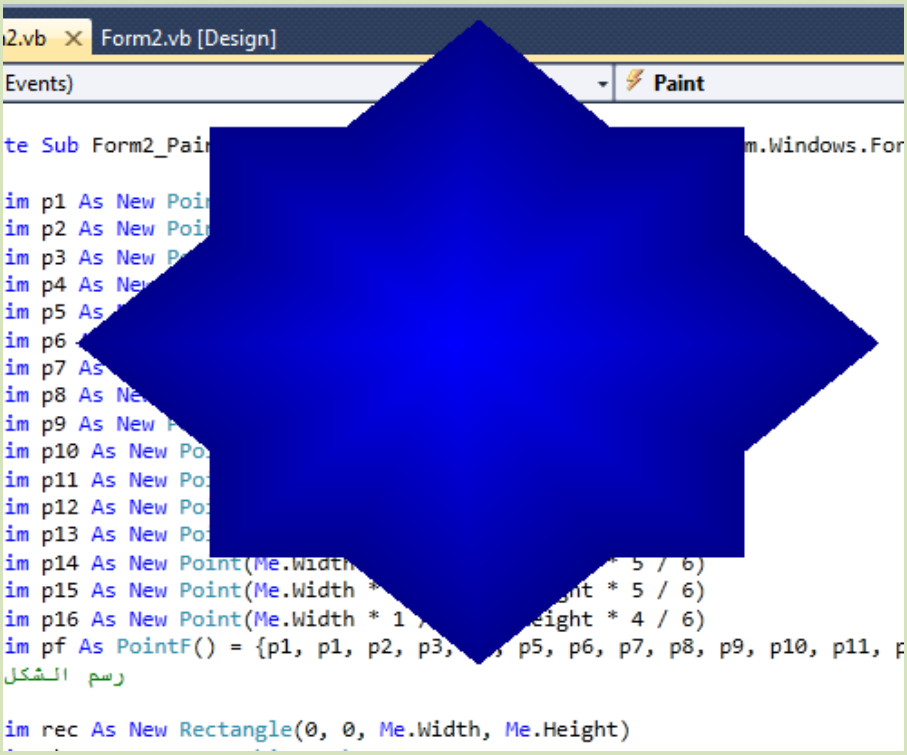
2

```
Dim Brush As New PathGradientBrush(shape)
Brush.CenterColor = Color.Blue
Dim cc As Color = Color.DarkBlue
Dim c() As Color = {cc}
Brush.SurroundColors = c
e.Graphics.FillEllipse(Brush, rec)
Me.Region = New Region(shape)
```

في المربع رقم 1 الكود المضاف وهو يمثل النقاط التي ترسم شكل المضلع الذي وشكل النجمة ثمانية الاضلاع

في المربع رقم 2 سنعدل الكود بحيث عدل الكود من اضافة بيضواي الى اضافة المضلع

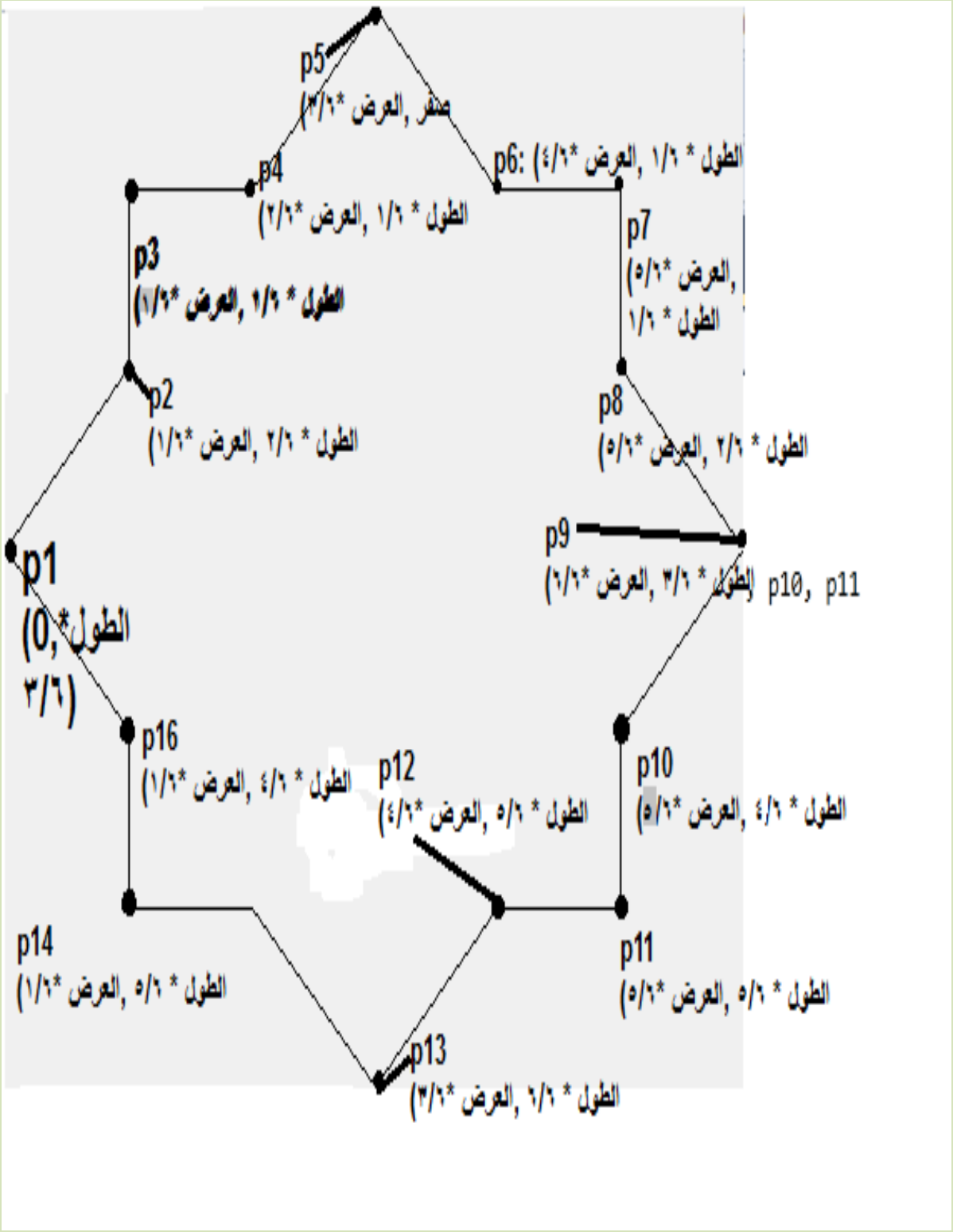
اذا شغلت البرنامج ستجد الفورم تظهر على الشكل الاتي



اذا اردت معرفة كيف تم توزيع النقاط على الفورم ستجد اننا نقسم الفورم الى ستة اجزاء بالطول و ستة اجزاء بالعرض ثم نحدد بعد ذلك الاحادث السيني و الصادي على الفورم

سننظر على الشكل الاتي ستلاحظ قد وضعنا النقاط على الفورم بحيث اننا نقسم طول الفورم ولعرضها على اجزاء من ستة ستلاحظ مثال ان مكان النقطة الرابعةp4 هو عرض الفورم مضرب على 6/2 فمثلا لو كان عرض النموذج 60 بكسل سنجد ان الاحدثي السيني للنقطة الرابعة هو 60مضروب في 6/2 فيكون الاحداثي السيني (X) يساوي 20 بكسل ويكون الاحداثي الصادي (y) هو طول ضروب في 6/1 فلو كان طول النموذج 60 يكون 10 بكسل وعلى هذه فقس كل النقاط

انظر للنموذج الاتي ستجد كيف وزعنا النقاط



تعديل ادوات التحكم

في هذا المثال سوف نقوم بتطوير شكل الزر بيحث يتغير لونه عندما يمر عليه الماوس ولكن لا يتغير بلون واحد سيكون لونين

```
If mousemve Then
    Dim gr As Graphics = e.Graphics
    Dim rec As New Rectangle(Button1.Left, Button1.Top, Button1.Width, Button1.Height)
    Dim shape As New GraphicsPath
    shape.AddRectangle(rec)
    Dim Brush As New PathGradientBrush(shape)
    Brush.CenterColor = Color.Blue
    Dim cc As Color = Color.DarkBlue
    Dim c() As Color = {cc}
    Brush.SurroundColors = c
    gr.FillRectangle(Brush, rec)
Else
    Dim gr As Graphics = e.Graphics
    Dim rec As New Rectangle(Button1.Left, Button1.Top, Button1.Width, Button1.Height)
    Dim shape As New GraphicsPath
    shape.AddRectangle(rec)
    Dim Brush As New PathGradientBrush(shape)
    Brush.CenterColor = Color.Blue
    Dim cc As Color = Color.LightBlue
    Dim c() As Color = {cc}
    Brush.SurroundColors = c
    gr.FillRectangle(Brush, rec)
End If
```

قم بفتح مشروع جديد واضف اليه زر ثم ضع الكود الاتي في الحدث Button1_paint

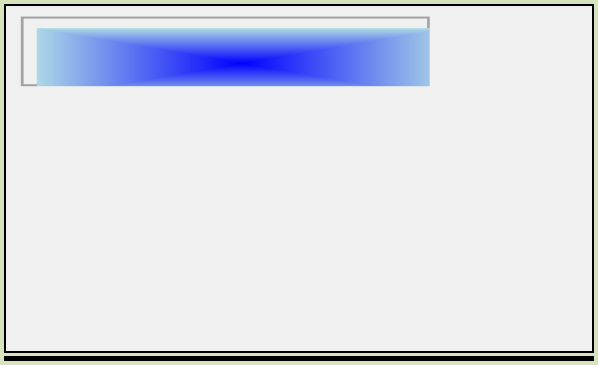
ستلاحظ ان الكود السابق هو نفسه الكود الذي استخدمنا عندما صممنا نموذج دائري ولكن ستجد هناك بعض الفروق وهي

- اننا لم نستخدم الخاصية Region في قص الزر ويمكن استخدامها ايضا اذا اردنا ان نجعل الزر دائره ..... **ساتركه تمرين لك تقوم انت به**
- اننا استخدمنا متغير اسمه MouseMove مهمته تحديد اذا مر الماوس على الزر ام لا فاذا مر الماوس يتم تنفيذ الامر الذي تحته وستجد ان الامر مكرر فيما عدا شيء واحد وهول الخاصية SurroundColor بحيث غيرنا لون عند مرور الماوس ما عدا ذكل ستجد نفسك تفهم هذا الكود

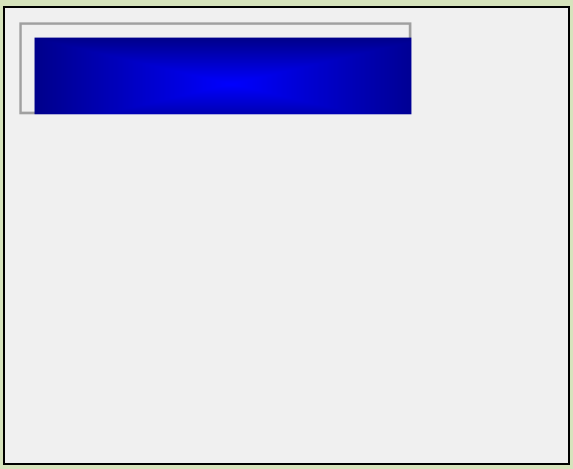
عند الحدث MouseMove للزر جعلنا المتغير MouseMove= true

وعند الحدث mouseLeave يكون المتغير mosuemove = false حتى يعود لون الزر كما كان قبل مرور الماوس

شغل البرنامج وستلاحظ شكل الزر كلاتي



قبل مرور الماوس



بعد مرور الماوس

ستلاحظ ان الزر لا يظهر النص المكتوب عليه ساتركها لك كتمرين فقد شرحتها بالتفصيل في الفصل الرابع

ملخص الفصل السابع

- تصميم شكل دائر للفورم
- تصميم شكل نجمة للفورم
- تصميم زر يتغير لونه عندما يمر عليه الماوس

تمرين للقارئ

قم بتصميم زر دائري وزر على شكل نجمة بحيث يمكن ان يمر عليه الماوس فيتغير لونه غيرنا لون النص الى اللون الابيض

خاتمة

وفي الختام اسئل الله العلي القدير ان اكون وفقت في شرح هذه الجزئية وهذه النقطة من بحر البرمجة وتكون قد استفدت كثير فهذا المجال اذا اتقنته سينفتح لك باب كبير في مجال برمجة الالعاب و تصميم النماذج وتستطيع اذا كنت متقننا التعامل مع قواعد البيانات ان تتعلم كيف تصمم تقرير خاص بك بدلا من اتسخدام البرامج الخارجية فانا اريد ان اساهم في ان نعتمد على برمجة ادواتنا بانفسنا ولا نعتمد على الادوات الخارجية لذا قدمت اليك جزء هام في تصميم وتجميل واجهتنا الرسومية لذا ارجو منك ان تعيد قرائة هذا الكتاب عدة مرات وقم تطبيقه حتى تتمكن منه

كما اطلب من الدعاء لي ان اوفق في تقديم كتب اخرى تساهم في تنمية مهارات شبابنا العربي و الاسلامي في هذا المجال

هذا لو كان فيه هذا الكتاب من احسان فمن الله وان كان فيه من تقصير فمن نفسي ومن الشيطان