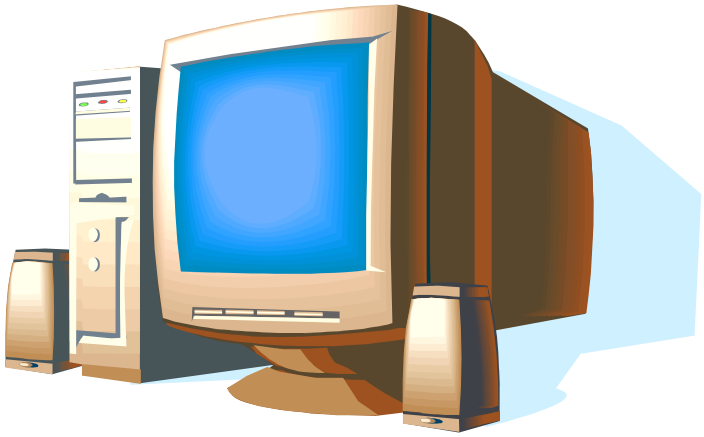
مشاريع بلغة

Visual Basic.NET

للصف الأول الثانوي



الفصل الدراسى الثانى

http://asmaagamal.weebly.com

اختبار في جدول الضرب

المطلوب عمل مشروع جديد باسم "TestInTable1" يمكن من خلاله اختبار المستخدم في جدول الضرب

طريقة التصميم :

نقوم بتصميم البرنامج حيث أننا نستخدم الكائنات الآتية من صندوق الأدوات Toolbox على نافذة النموذج (Form1) كما هو موضح بالشكل الآتي:

١- عدد (4 Labels)
٢- عدد (1 TextBox)
٣- عدد (2 Buttons)

ثم نقوم بتوصيف النموذج باستخدام خاصية (Text) لتغيير عنوان الكائنات كما هو موضح بالشكل الآتي:

ثم نقوم بالانتقال إلى نافذة البرمجة لكتابة الكود الآتي :

```
(General)                                   (Declarations)
Public Class Form1
    Dim A, B, M As Byte
    Dim R As Single

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e A
        Randomize()
        A = Fix(Rnd() * 13) : B = Fix(Rnd() * 13)
        Label1.Text = A : Label3.Text = B
        TextBox1.Clear() : TextBox1.Focus()
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e A
        R = Val(TextBox1.Text)
        M = A * B
        If R = M Then
            MsgBox("الحل صواب")
        Else
            MsgBox("الحل خطأ ... الصواب هو " & M)
        End If
    End Sub
End Class
```

برمجة زر "مسألة جديدة"

برمجة زر "اختبر الحل"

ملحوظة:

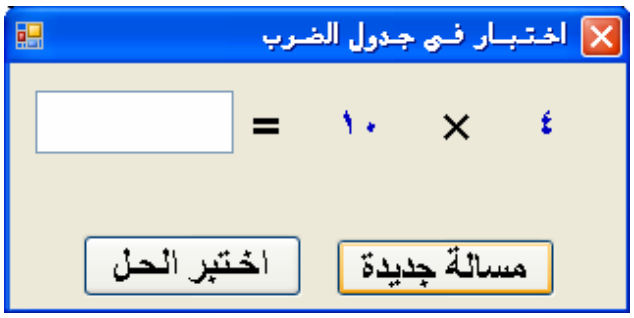
* المخزن (المتغير العددي) A يمثل العدد الأول الذي سيظهر داخل Label1 .
* المخزن (المتغير العددي) B يمثل العدد الثاني الذي سيظهر داخل Label3 .
* المخزن (المتغير العددي) M يمثل حاصل ضرب العددين .
* المخزن (المتغير العددي) R يمثل الحل الذي سوف يدخله المستخدم داخل TextBox1 .
* تم استخدام الدالة Rnd لإظهار الأعداد العشوائية والوسيلة Randomize لإظهار الأعداد بترتيب مختلف في كل مرة يتم فيها تشغيل البرنامج.
* تم استخدام الدالة Fix للحصول على الجزء الصحيح من القيمة الناتجة من ضرب ناتج الدالة Rnd في ١٣ لحساب العدد الأول الذي سيظهر داخل Label1 والعدد الثاني الذي سيظهر داخل Label3 بطريقة عشوائية.
* تم استخدام جملة If...Then...Else... لإظهار رسالة صواب أم رسالة خطأ.

تشغيل البرنامج :

١- يتم ضغط زر " مسالة جديدة"، بعد ذلك سيظهر عدد صحيح عشوائي قيمته من ٠ إلى ١٢ داخل كلٍ من Label1 و Label3 ثم يضع مؤشر الكتابة داخل صندوق النص TextBox1 .

٢- يقوم المستخدم بإدخال الحل داخل TextBox1.

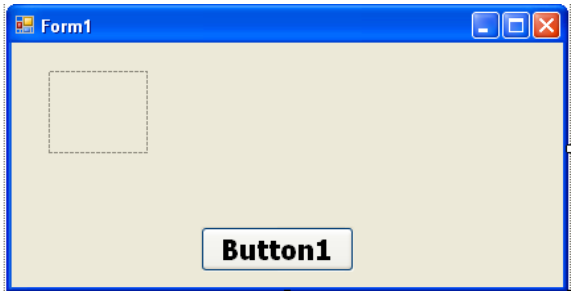
٣- يتم الضغط على زر "اختبر الحل" فإذا كان الحل صواب يظهر رسالة "الحل صواب" أما إذا كان خطأ سيظهر رسالة "الحل خطأ"، ثم يظهر ناتج الضرب الصحيح، كما بالشكل التالي.



ظهور صورة في أماكن عشوائية في النموذج

المطلوب عمل مشروع جديد باسم "RndPosition1" بعد التشغيل يمكن من خلاله ظهور صورة PictureBox1 في أماكن عشوائية داخل نافذة النموذج عند الضغط على زر أوامر Button1 .

طريقة التصميم :



نقوم بتصميم البرنامج حيث أننا نستخدم الكائنات الآتية من صندوق الأدوات Toolbox على نافذة النموذج (Form1) كما هو موضح بالشكل الآتي:

١- عدد (1 PictureBox)

٢- عدد (1 Button)

ثم نقوم بتوصيف النموذج باستخدام خاصية (Text) لتغيير عنوان الكائنات وخاصية (Image) لإدراج صورة داخل الأداة PictureBox1 كما هو موضح بالشكل الآتي:

ثم نقوم بالانتقال إلى نافذة البرمجة لكتابة الكود الآتي :

برمجة زر" أنقر هنا "

```
Button1                                    Click
Public Class Form1

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e A
        Dim X, Y, L, H As Single
        Randomize()
        X = Me.Width - PictureBox1.Width
        Y = Me.Height - PictureBox1.Height
        L = Fix(Rnd() * X)
        H = Fix(Rnd() * Y)
        PictureBox1.Left = L
        PictureBox1.Top = H
    End Sub
End Class
```

ملحوظة:

* المخزن (المتغير العددي) X يمثل فرق عرض نافذة النموذج Form1 و الأداة PictureBox1 .
* المخزن (المتغير العددي) Y يمثل فرق طول نافذة النموذج Form1 و الأداة PictureBox1.
* المخزن (المتغير العددي) L يمثل البعد الأفقي للأداة PictureBox1 عن أقصى يسار النموذج.
* المخزن (المتغير العددي) H يمثل البعد الرأسي للأداة PictureBox1 عن شريط عنوان النموذج.
* تم استخدام الدالة Rnd لإظهار الأعداد العشوائية والوسيلة Randomize لإظهار الأعداد بترتيب مختلف في كل مرة يتم فيها تشغيل البرنامج.
* تم استخدام الدالة Fix للحصول على الجزء الصحيح من القيمة الناتجة من ضرب ناتج الدالة Rnd في X ، Y لتحديد البعد الأفقي والبعد الرأسي للأداة PictureBox1 بطريقة عشوائية.

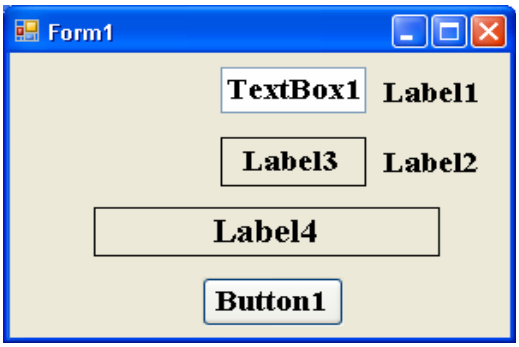
تشغيل البرنامج :

نقوم بالضغط على زر " أنقر هنا"، بعد ذلك نلاحظ انتقال الصورة في أماكن عشوائية داخل نافذة النموذج.

رقم الحظ هو رقم الكمبيوتر

المطلوب عمل مشروع جديد باسم "DiceGame1" بعد التشغيل يمكن من خلاله أن يلعب المستخدم مع الكمبيوتر من خلال زر أوامر Button1 بعد الضغط على هذا الزر يظهر رقم عشوائي للكمبيوتر من ١ إلى ٦ داخل Label3 ، ثم يقوم المستخدم بإدخال رقم داخل TextBox1 ثم يضغط زر Button1 فإذا تساوى رقمه مع رقم الكمبيوتر تظهر رسالة" تكسب "وخلاف ذلك تظهر رسالة "تخسر" وهكذا.

طريقة التصميم :



نقوم بتصميم البرنامج حيث أننا نستخدم الكائنات الآتية من صندوق الأدوات Toolbox على نافذة النموذج (Form1) كما هو موضح بالشكل الآتي:

١- عدد (4 Labels)

٢- عدد (1 TextBox)

٣- عدد (1 Button)

ثم نقوم بتوصيف النموذج باستخدام خاصية (Text) لتغيير عنوان الكائنات كما هو موضح بالشكل الآتي:

ثم نقوم بالانتقال إلى نافذة البرمجة لكتابة الكود الآتي :

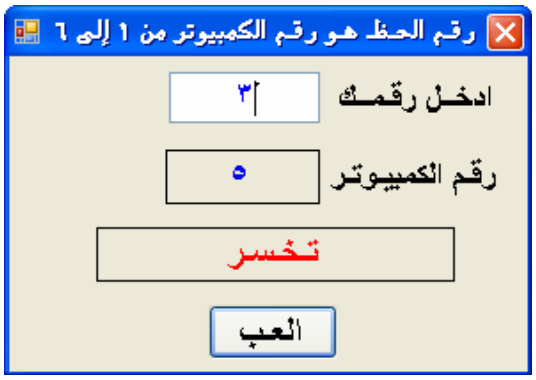
```
(General)                                    (Declarations)
Public Class Form1

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e A
        Dim N, C As Byte
        Randomize()
        N = Val(TextBox1.Text)
        C = 1 + Fix(Rnd() * 6)
        Label3.Text = C
        If N = C Then
            Label4.Text = "تكسب"
        Else
            Label4.Text = "تخسر"
        End If
        TextBox1.Focus()
    End Sub

    Private Sub TextBox1_TextChanged(ByVal sender As System.Object, By
        Label3.Text = "" : Label4.Text = ""
    End Sub
End Class
```

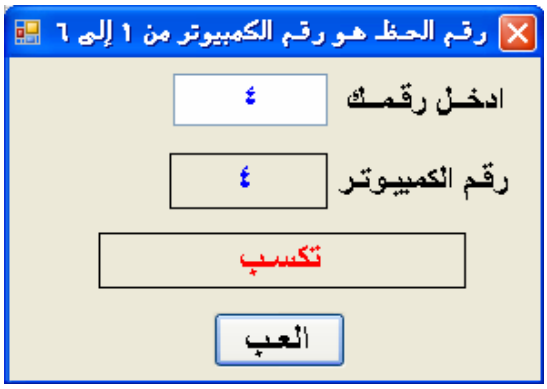
ملحوظة:

* المخزن (المتغير العددي) N يمثل الرقم الذي سوف يدخله المستخدم داخل TextBox1 .
* المخزن (المتغير العددي) C يمثل الرقم الذي سيظهر داخل Label3 .
* تم استخدام الدالة Rnd لإظهار الأعداد العشوائية والوسيلة Randomize لإظهار الأعداد بترتيب مختلف في كل مرة يتم فيها تشغيل البرنامج.
* تم استخدام الدالة Fix للحصول على الجزء الصحيح من القيمة الناتجة من ضرب ناتج الدالة Rnd في الرقم ٦ وذلك لحساب الرقم الذي سيظهر داخل Label3 بطريقة عشوائية.

تشغيل البرنامج :

نقوم بالضغط على زر " العب " عدة مرات بعد وضع أرقام مختلفة في TextBox1 في كل مرة . نلاحظ ظهور رسالة " تكسب " في حالة تساوي الرقمين ورسالة " تخسر " خلاف ذلك ، كما بالشكل التالي.

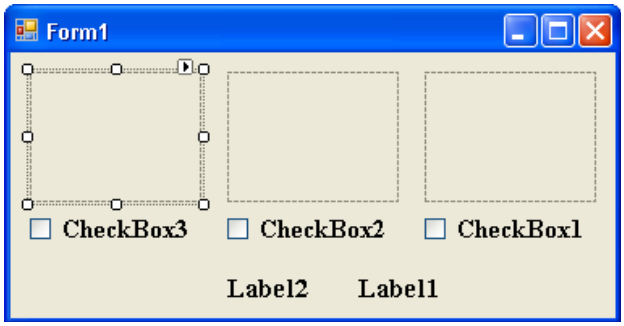




بيع ملحقات كمبيوتر

المطلوب عمل مشروع جديد باسم "ComputerShop1" على أن يمكنك بعد التشغيل حرية اختيار أحد ملحقات الكمبيوتر أو أكثر، ثم يتم حساب إجمالي السعر.

طريقة التصميم :



نقوم بتصميم البرنامج حيث أننا نستخدم الكائنات الآتية من صندوق الأدوات Toolbox على نافذة النموذج (Form1) كما هو موضح بالشكل الآتي:

١- عدد (3 PictureBoxes)

٢- عدد (3 CheckBoxes)

٣- عدد (2 Labels)

ثم نقوم بتوصيف النموذج باستخدام خاصية (Text) لتغيير عنوان الكائنات وخاصية (Image) لإدراج صور داخل الأدوات PictureBoxs كما هو موضح بالشكل الآتي:

ثم نقوم بالانتقال إلى نافذة البرمجة لكتابة الكود الآتي :

```
(General)                                    (Declarations)
Public Class Form1

    Dim A, B, C As Single

    Private Sub CheckBox1_CheckedChanged(ByVal sender As System.Object
        If CheckBox1.Checked Then A = 1200 Else A = 0
        Label2.Text = A + B + C
    End Sub

    Private Sub CheckBox2_CheckedChanged(ByVal sender As System.Object
        If CheckBox2.Checked Then B = 1500 Else B = 0
        Label2.Text = A + B + C
    End Sub

    Private Sub CheckBox3_CheckedChanged(ByVal sender As System.Object
        If CheckBox3.Checked Then C = 90 Else C = 0
        Label2.Text = A + B + C
    End Sub
End Class
```

ملحوظة:

* المخزن (المتغير العددي) A يمثل سعر أولى ملحقات الكمبيوتر .
* المخزن (المتغير العددي) B يمثل سعر ثاني ملحقات الكمبيوتر.
* المخزن (المتغير العددي) C يمثل سعر ثالث ملحقات الكمبيوتر.
* تم استخدام جملة If...Then...Else... لتحديد سعر إحدى ملحقات الكمبيوتر.

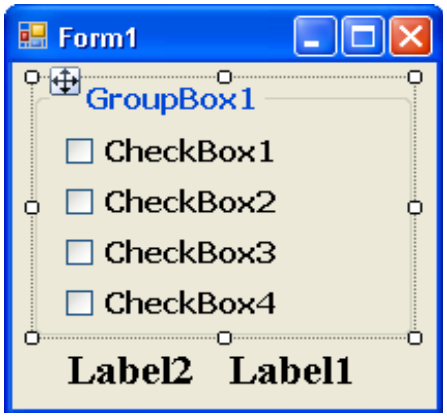
تشغيل البرنامج :

قم بعمل Start ، وحاول تحديد ملحقات الكمبيوتر المناسبة ، ولاحظ السعر الإجمالي لها.

مشروبات

المطلوب عمل مشروع جديد باسم "Drinks1" على أن يمكنك بعد التشغيل حرية اختيار مشروب أو أكثر من مجموعة من المشروبات، ثم يتم حساب إجمالي السعر.

طريقة التصميم :



نقوم بتصميم البرنامج حيث أننا نستخدم الكائنات الآتية من صندوق الأدوات Toolbox على نافذة النموذج (Form1) كما هو موضح بالشكل الآتي:

١- عدد (1 GroupBox)

٢- عدد (4 CheckBoxes)

٣- عدد (2 Labels)

ثم نقوم بتوصيف النموذج باستخدام خاصية (Text) لتغيير عنوان الكائنات كما هو موضح بالشكل الآتي:

ثم نقوم بالانتقال إلى نافذة البرمجة لكتابة الكود الآتي :

```
(General)                                    (Declarations)
Public Class Form1
    Dim Tea, Cof, Sah, Coc As Single

    Private Sub CheckBox1_CheckedChanged(ByVal sender As System.Object
        If CheckBox1.Checked Then Tea = 1 Else Tea = 0
        Label2.Text = Tea + Cof + Sah + Coc
    End Sub

    Private Sub CheckBox2_CheckedChanged(ByVal sender As System.Object
        If CheckBox2.Checked Then Cof = 2 Else Cof = 0
        Label2.Text = Tea + Cof + Sah + Coc
    End Sub

    Private Sub CheckBox3_CheckedChanged(ByVal sender As System.Object
        If CheckBox3.Checked Then Sah = 2 Else Sah = 0
        Label2.Text = Tea + Cof + Sah + COC
    End Sub

    Private Sub CheckBox4_CheckedChanged(ByVal sender As System.Object
        If CheckBox4.Checked Then Coc = 3 Else Coc = 0
        Label2.Text = Tea + Cof + Sah + COC
    End Sub
End Class
```

برمجة " شاي "

برمجة " قهوة "

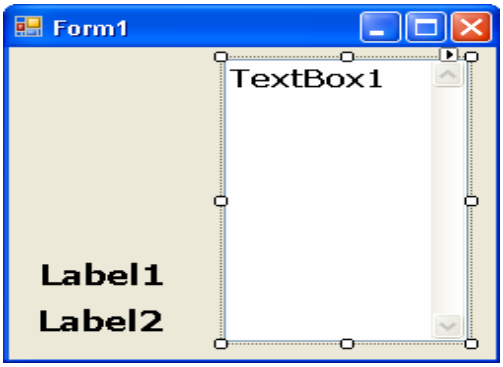
برمجة " سحلب "

برمجة " كاكاو "

إظهار الأعداد الصحيحة من ١ إلى ١٠ و مجموعها

المطلوب عمل مشروع جديد باسم "NumbersSum1to10" حيث بعد التشغيل مباشرً ة تظهر الأعداد الصحيحة من ١ إلى ١٠ داخل صندوق نص TextBox1 ، ويظهر مجموعها داخل Label2 .

طريقة التصميم :



نقوم بتصميم البرنامج حيث أننا نستخدم الكائنات الآتية من صندوق الأدوات Toolbox على نافذة النموذج (Form1) كما هو موضح بالشكل الآتي:

١- عدد (1 TextBox)

٢- عدد (2 Labels)

ثم نقوم بتوصيف النموذج باستخدام خاصية (Text) لتغيير عنوان الكائنات وجعل خاصية (MultiLine) لـ TextBox1 (True) وخاصية (ScrollBars) (Vertical) وخاصية (RightToLeft) (Yes) كما هو موضح بالشكل الآتي:

ثم نقوم بالانتقال إلى نافذة البرمجة لكتابة الكود الآتي :

```
(Form1 Events)                                  Load
Public Class Form1

    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As S
        Dim N As Byte
        Dim T As Integer
        TextBox1.Clear()
        T = 0
        For N = 1 To 10
            T = T + N
            TextBox1.Text = TextBox1.Text & N & vbCrLf
        Next
        Label2.Text = T
    End Sub
End Class
```

ملحوظة:

* المخزن (المتغير العددي) T يمثل مجموع الأعداد الصحيحة .
* المخزن (المتغير العددي) N يمثل الأعداد الصحيحة.
* تم استخدام جملة For.....Next لظهور الأعداد الصحيحة.
* تم استخدام الأمر TextBox1.Text = TextBox1.Text & N & vbCrLf لكتابة كل عدد من الأعداد الصحيحة في سطر منفصل داخل TextBox1 .

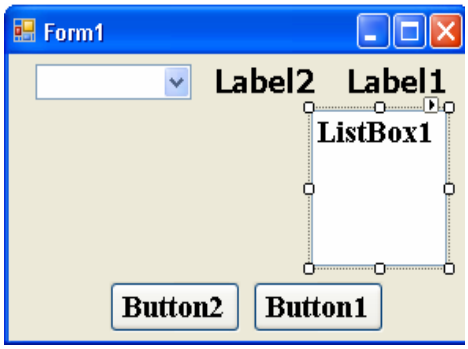
تشغيل البرنامج :

قم بتنفيذ الكود السابق ، ولاحظ ظهور الأعداد الصحيحة من ١ إلى ١٠ داخل صندوق النص TextBox1 وظهور مجموعها داخل Label2 كما بالشكل الآتي:

السؤال عن عاصمة دولة بطريقة عشوائية "دول و عواصم"

المطلوب عمل مشروع جديد باسم "Capitals2" حيث بعد التشغيل يمكن للمستخدم الضغط على زر" سؤال جديد " فيظهر " اسم دولة عشوائي " داخل Label2 وعلى المستخدم اختيار "العاصمة" المناسبة من صندوق السرد ComboBox1 ، كما يمكنه اختبار الإجابة بالضغط على زر "النتيجة" فيظهر له صندوق رسالة بالنتيجة "إجابة صحيحة" إذا كانت صحيحة و رسالة "إجابة خطأ" إذا كانت غير صحيحة.

طريقة التصميم :



نقوم بتصميم البرنامج حيث أننا نستخدم الكائنات الآتية من صندوق الأدوات Toolbox على نافذة النموذج (Form1) كما هو موضح بالشكل الآتي:

١- عدد (2 Labels)
٢- عدد (2 Buttons)
٣- عدد (1 ListBox)
٤- عدد (1 ComboBox)

ثم نقوم بتوصيف النموذج باستخدام خاصية (Text) لتغيير عنوان الكائنات واستخدام خاصية (Items) لـ ListBox1 لإدخال أسماء بعض الدول ولـ ComboBox1 لإدخال العواصم المناظرة للدول السابقة ، كما هو موضح بالشكل الآتي:

ثم نقوم بالانتقال إلى نافذة البرمجة لكتابة الكود الآتي :

```
(General)                                   (Declarations)
Public Class Form1
    Dim N, R As Byte

    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As S
        ListBox1.Hide()
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e A
        Randomize()
        N = Fix(ListBox1.Items.Count * Rnd())
        Label2.Text = ListBox1.Items(N)
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e A
        R = ComboBox1.SelectedIndex
        If N = R Then
            MsgBox("إجابة صحيحة")
        Else
            MsgBox("إجابة خطأ")
        End If
    End Sub
End Class
```

برمجة إخفاء صندوق القائمة "ListBox1" بعد التشغيل مباشرة

برمجة زر "سؤال جديد"

برمجة زر "النتيجة"

ملحوظة:

* المخزن (المتغير العددي) N يمثل رقم الفهرس للعنصر داخل ListBox1 .
* المخزن (المتغير العددي) R يمثل رقم الفهرس للعنصر المحدد داخل ComboBox1.
* تم استخدام الدالة Rnd لإظهار الأعداد العشوائية والوسيلة Randomize لإظهار الأعداد بترتيب مختلف في كل مرة يتم فيها تشغيل البرنامج.
* تم استخدام الدالة Fix للحصول على الجزء الصحيح من القيمة الناتجة من ضرب ناتج الدالة Rnd في عدد العناصر الموجودة داخل ListBox1.
* تم استخدام جملة If...Then...Else... لإظهار رسالة "إجابة صحيحة" أم رسالة "إجابة خطأ".

تشغيل البرنامج :

قم بتنفيذ الكود السابق ، وحاول الضغط على زر "سؤال جديد " ، سيظهر "اسم دولة " داخل Label2 ثم قم باختيار "اسم العاصمة المناسبة لهذه الدولة " من صندوق السرد ، ثم حاول الضغط على زر "النتيجة" لاختيار الإجابة، كما بالأشكال التالية :

* الشكل الأول : بعد التشغيل مباشرةً

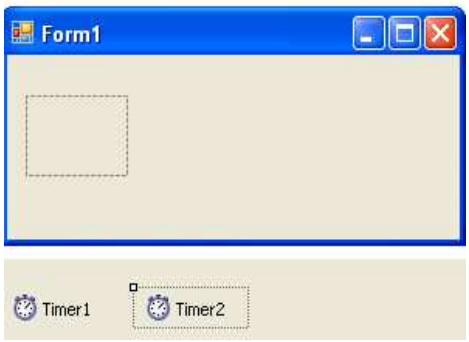
* الشكل الثاني : بعد التشغيل تم عمل الآتي:
  ١- ضغط على زر "سؤال جديد" فظهرت "مصر" في Label2.
  ٢- ضغط على صندوق السرد ComboBox1 لاختيار العاصمة "دمشق".
  ٣- ضغط زر "النتيجة " فظهر صندوق الرسالة "إجابة خطأ".

* الشكل الثالث : بعد التشغيل تم عمل الآتي:
  ١- ضغط على زر "سؤال جديد" فظهرت "مصر" في Label2 .
  ٢- ضغط على صندوق السرد ComboBox1 لاختيار العاصمة "القاهرة".
  ٣- ضغط زر "النتيجة " فظهر صندوق الرسالة " إجابة صحيحة ".

حركة أفقية لصورة بصفة مستمرة داخل نافذة النموذج

المطلوب عمل مشروع جديد باسم "PictureMoving1" حيث بعد التشغيل تتحرك الصورة حركة أفقية داخل نافذة النموذج بصفة مستمرة، ثم ترتد إذا وصلت إلى أي جانب من جوانب النافذة (جانب يسار أو جانب يمين).

طريقة التصميم :



نقوم بتصميم البرنامج حيث أننا نستخدم الكائنات الآتية من صندوق الأدوات Toolbox على نافذة النموذج (Form1) كما هو موضح بالشكل الآتي:

١- عدد (1 PictureBox)

٢- عدد (2 Timers)

ثم نقوم بتوصيف النموذج باستخدام خاصية (Text) لتغيير عنوان النموذج وخاصية (Image) لإدراج صورة داخل الأداة PictureBox1 ، كما هو موضح بالشكل الآتي:

ثم نقوم بالانتقال إلى نافذة البرمجة لكتابة الكود الآتي :

برمجة "Timer1"

```
(General)                    (Declarations)
Public Class Form1
    Dim X As Integer

    Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
        PictureBox1.Left = PictureBox1.Left + 5
        If PictureBox1.Left > X - 10 Then
            Timer1.Enabled = False : Timer2.Enabled = True
        End If
    End Sub

    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As S
        X = Me.Width - PictureBox1.Width
        Timer1.Enabled = True : Timer2.Enabled = False
    End Sub

    Private Sub Timer2_Tick(ByVal sender As System.Object, ByVal e As
        PictureBox1.Left = PictureBox1.Left - 5
        If PictureBox1.Left < 0 Then
            Timer1.Enabled = True : Timer2.Enabled = False
        End If
    End Sub
End Class
```

برمجة "Timer2"

ملحوظة:

* المخزن (المتغير العددي) X يمثل الفرق بين عرض نافذة النموذج Form1 وعرض صندوق الصور PictureBox1 .

* تم استخدام جملة If...Then... لجعل كل من Timer1 و Timer2 متاح للاستخدام أم لا.

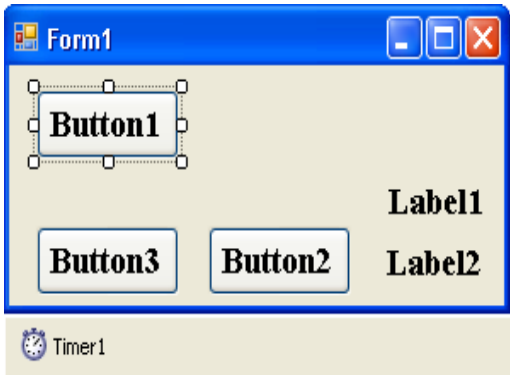
تشغيل البرنامج :

قم بتنفيذ الكود السابق ، ولاحظ تحرك الصورة أفقيًا داخل نافذة النموذج بصفة مستمرة.

وضع زر في أماكن عشوائية داخل النموذج ويتم نقره

المطلوب عمل مشروع جديد باسم "Game_ClickHere " حيث بعد التشغيل يظهر زر الأوامر Button1 في أماكن عشوائية داخل نافذة النموذج لمدة ٣٠ ثانية ، و يمكن للمستخدم محاولة النقر على هذا الزر ، وبعد النقر عليه يتم تسجيل عدد مرات النقر ، ثم بعد انتهاء الوقت المحدد يعطي رسالة بأن الوقت انتهى ، ويمكن للمستخدم تكرار اللعب مرات أخرى.

طريقة التصميم :



نقوم بتصميم البرنامج حيث أننا نستخدم الكائنات الآتية من صندوق الأدوات Toolbox على نافذة النموذج (Form1) كما هو موضح بالشكل الآتي:

١- عدد (3 Buttons)

٢- عدد (2 Labels)

٣- عدد (1 Timer)

ثم نقوم بتوصيف النموذج باستخدام خاصية (Text) لتغيير عنوان النموذج وخاصية (Image) لإدراج صورة داخل الأداة Button1 ، كما هو موضح بالشكل الآتي:

ثم نقوم بالانتقال إلى نافذة البرمجة لكتابة الكود الآتي :

```
(General)                                    (Declarations)
Public Class Form1
    Const K = " حاول النقر على الوجه"
    Dim N, T, X, Y As Integer

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e A
        N = 0 : T = 60
        Button1.Enabled = False : Button2.Enabled = True
        Button3.Enabled = False : Label2.Text = N
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e A
        Me.Text = K
        Timer1.Enabled = True : Button1.Enabled = True
        Button2.Enabled = False : Button3.Enabled = False
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e A
        N = N + 1 : Label2.Text = N
    End Sub
```

برمجة زر "جديد"

برمجة زر "العب"

برمجة زر "الوجه"

برمجة "Form1"

```
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As S
        Randomize()
        Button1.Enabled = False : Button2.Enabled = False
        X = Me.Width - Button1.Width
        Y = Me.Height - 2 * Button1.Height
        Timer1.Interval = 500
    End Sub
```

برمجة "Timer1"

```
    Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
        Button1.Left = X * Rnd() : Button1.Top = Y * Rnd()
        T = T - 1
        Me.Text = "الـوقــت المتبقــي: " & T
        If T = 0 Then
            Timer1.Enabled = False
            Button1.Enabled = False
            Button3.Enabled = True
            MsgBox("الـوقــت انـتهــى")
        End If
    End Sub
End Class
```

ملحوظة:

* المخزن الثابت K يمثل ما سيكتب على عنوان نافذة النموذج Form1.
* المخزن (المتغير العددي) N يمثل عدد النقرات.
* المخزن (المتغير العددي) T يمثل الوقت المتبقي.
* المخزن (المتغير العددي) X يمثل الفرق بين عرض نافذة النموذج Form1 وعرض زر الوجه Button1 .
* المخزن (المتغير العددي) Y يمثل الفرق بين ارتفاع نافذة النموذج Form1 وضعف ارتفاع زر الوجه Button1 .
* تم استخدام الدالة Rnd لإظهار الأعداد العشوائية والوسيلة Randomize لإظهار الأعداد بترتيب مختلف في كل مرة يتم فيها تشغيل البرنامج وذلك لتحديد المسافة التي سوف يتحرك بها زر الوجه Button1 أفقياً ورأسياً.

تشغيل البرنامج :

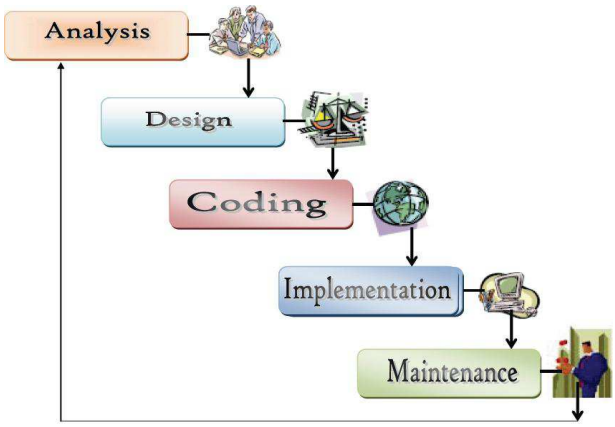
قم بتنفيذ الكود السابق ، وحاول الضغط على زر "جديد" ، ثم زر "العب" و بعد ذلك أنقر زر الوجه ، و لاحظ زيادة العدد في كل مرة.

دورة حياة تطوير النظام

SDLC

(System Development Life Cycle)

يحتاج أي نظام في الكون سوءًا كان ماديًا أم افتراضيًا إلى اتباع مراحل أساسية لإنشاء هذا النظام أو التحسين في أدائه، وتتلخص هذه المراحل في الآتي:



١. التحليل (Analysis)
٢. التصميم (Design)
٣. التكويد (Coding)
٤. التنفيذ (Implementation)
٥. الصيانة (Maintenance)

والجدول التالي يوضح هذه المراحل.

المراحل	الأنشطة المصاحبة
التحليل (Analysis)	تحليل المشكلة المطروحة ومعرفة متطلبات النظام
التصميم (Design)	تصميم ووضع طريقة الحل (Flowchart – Pseudo Code - ....etc)
التكويد (Coding)	كتابة كود البرنامج باستخدام لغة الكمبيوتر المناسبة
التنفيذ (Implementation)	تشغيل البرنامج وإدخال البيانات اللازمة ثم اختبار النتائج ثم بعد ذلك توثيق البرنامج
الصيانة (Maintenance)	إجراء التعديلات المطلوبة لملاحقة التطور وعمل نسخ احتياطية للبيانات