

* الفصل الأول * Data البيانات

مقدمة:

تعلمنا فيما سبق كيفية إعداد واجهة البرنامج باستخدام لغة Visual Basic .NET (وكيفية ضبط الخصائص) Properties في وضع التصميم من خلال نافذة الخصائص Properties ونعرفنا على نافذة الكود Code window وإجراءات الأحداث Event Procedure وكيفية تخصيص قيم برمجية لخصائص بعض الأدوات Controls ومن خلالها اتضح لنا أنه يمكن لمستخدم البرنامج أن يدخل قيم (بيانات) ذات أنواع مختلفة كما هو موضح

نلاحظ: أن (ياسمين تامر، ٧-١٢-٢٠٠٥، أنثى، ٥) عبارة عن قيم مختلفة في نوعها حيث:

اسم الحقل	القيمة (البيان)	نوع القيمة (نوع البيان)
الاسم	ياسمين تامر	نصية
تاريخ الميلاد	٢٠٠٥-١٢-٧	تاريخ
النوع	أنثى	منطقية
عدد أفراد الأسرة	٥	رقمية

ملاحظات هامة

- ١- كل هذه القيم يتم تخزينها في ذاكرة الكمبيوتر RAM بتمثيل معين يختلف حسب نوع هذه القيمة
- ٢- وكل نوع من هذه البيانات له طريقة تخزين مختلفة في ذاكرة الكمبيوتر وذلك يرجع إلى الاستخدام الأمثل لذاكرة الكمبيوتر من حيث المساحة التخزينية والعمليات الحسابية والمنطقية التي ستجرى على هذه البيانات

سؤال: ما هو سبب اختلاف طريقة تخزين أنواع البيانات في ذاكرة الكمبيوتر؟

- ١- حتى يتم استخدام ذاكرة الكمبيوتر الاستخدام الأمثل من حيث المساحة التخزينية المخصصة لكل نوع بيان
- ٢- حتى يتم استخدام ذاكرة الكمبيوتر الاستخدام الأمثل من حيث العمليات الحسابية والمنطقية التي ستجرى على هذه البيانات أو القيم

أنواع البيانات Data Type

توفر لغة Visual Basic .NET العديد من أنواع البيانات (Data Types) التي تستخدم في تخزين قيم البيانات التي يدخلها المستخدم أو الناتجة من تنفيذ الأوامر أو التعليمات الخاصة بالبرنامج

- ١- عندما يتم إدخال البيانات بواسطة وحدات الإدخال فإنها تخرن داخل الذاكرة الداخلية RAM
- ٢- وتتكون ذاكرة الحاسب من ملايين الخلايا (الأمكان) المتساوية وتسمى هذه الخلايا بالبايت (byte) وتحمل هذه الخلايا أرقاماً متسلسلة تبدأ من الصفر
- ٣- تمثل عدد هذه الخلايا حجم ذاكرة الحاسب ((أي أنه كلما زاد عدد هذه الخلايا (byte) كانت ذاكرة الجهاز أكبر وبالتالي فإن سعة تخزين البيانات تكون أكبر))

أهم أنواع البيانات Data Type

أولاً : البيانات الرقمية (Numeric Data Types) :

هي التي يمكن استخدامها في تخزين البيانات الرقمية وتنقسم إلى

١- البيانات الرقمية الصحيحة (Integral Numeric Types) : وهي التي يمكن أن تخزن بها أرقام (أعداد) صحيحة ومنها : (Long- Integer -Short - Byte)

٢- البيانات الرقمية الغير الصحيحة (No integral Numeric Types) : وهي التي يمكن أن تخزن بها أرقام بعلامات عشرية . ومنها : (Decimal - Double - Single)

ثانياً : البيانات الحرفية (Character Data Type) : وتستخدم في تخزين الحروف والكلمات ومنها : (String- Char)

ثالثاً : البيانات المتنوعة (Miscellaneous Data Types) : هي البيانات التي لا تندرج تحت تصنيف البيانات الرقمية أو الحرفية وتستخدم في تخزين بيانات منطقية أو تاريخ أو غيرها ومنها : (Boolean - Date - Object)

هام : كل نوع البيانات Data Types له

١. حجم المتغير (عدد الـ Bytes) أي حجم الذاكرة التي يحتاجها المتغير من النوع المصاحب له أو حيز التخزين في ذاكرة الكمبيوتر مثل Integer عند استخدامه يشغل (٤ bytes)
٢. المدى (Range) هو حدود القيم المتاحة لتخزين المتغير به مثل النوع Byte (يبدأ من صفر وينتهي عند ٢٥٥)
٣. الدقة (Precision) هي عدد الخانات التي يتم تمثيل المتغير بها (وهي خاصة بالمتغيرات الرقمية فقط) وتختلف درجة الدقة باختلاف نوع المتغير
٤. المتغيرات من النوع (short , integer , long) تستخدم لتخزين الأعداد الصحيحة فقط وليست الأعداد التي تحتوي على كسور
٥. أما المتغيرات من النوع (single, double, decimal) تستخدم لتخزين الأعداد التي تحتوي على كسور عشرية ولكن بدرجات دقة مختلفة تختلف من نوع إلى آخر

سؤال : ما الفرق بين البيانات الرقمية والبيانات الحرفية ؟

البيانات الرقمية

هي البيانات التي يتم إجراء العمليات الحسابية عليها

البيانات الحرفية

هي البيانات التي لا يجوز إجراء العمليات الحسابية عليها حتى لو كانت هذه البيانات أرقام مثل (رقم الهاتف - رقم المنزل - رقم البطاقة)

نوع البيان	نوع البيان	الحجم الذي يشغله في الذاكرة بالبايت	طبيعة البيان المخزن داخله	درجة الدقة (عدد الخانات)
أرقام صحيحة	Byte	١	من ٠ الى ٢٥٥	٣ أرقام
	Short	٢	من - ٣٢٧٦٨ الى ٣٢٧٦٧	٥ أرقام
	Integer	٤	من - ٢١٤٧٤٨٣٦٤٨ الى ٢١٤٧٤٨٣٦٤٧	١٠ أرقام
	Long	٨	أكثر من ٢١٤٧٤٨٣٦٤٧	١٩ رقم
أرقام عشرية	single	٤	من ٣٨١٠ الى ٣٨١٠	٧ أرقام
	double	٨	أكبر ٣٨١٠	١٥ أرقام
	decimal	١٦	مبلغ من المال	٢٩ رقم
حروف	char	٢	تخزين حرف واحد	لا يوجد
	string	العديد من الخلايا	تخزين سلسلة حرفية من ٠ الى ٢ بليون حرف	لا يوجد
قيمة منطقية	boolean	٢	yes or No او True or False	لا يوجد
كائن	Object	٤ بايت	كل الأنواع يمكن تخزينها هنا	لا يوجد
تاريخ	date	٨	تاريخ من ١/١/٠٠٠١ الى ٣١/١٢/٩٩٩٩	لا يوجد

مثال : لاحظ : هام

١- إذا أردنا تخزين العدد الصحيح ٢٠٠ فإننا يمكن تخزينه في كل من byte , short , integer , long ولكن الأصح تخزينه داخل متغير من نوع byte لأنه بذلك سيشتغل مساحة أقل في الذاكرة

مثال : توضيحي : فقط

٢- إذا لو أردنا تخزين العدد ١٢٣٤٥٦,٧٨٩١٢٣٤٥٦٧٨ داخل ثلاث متغيرات من نوع single و double و decimal نجد أن

القيمة التي تخزن داخل متغير من نوع single هي : ١٢٣٤٥٦,٧

القيمة التي تخزن داخل متغير من نوع double هي : ١٢٣٤٥٦,٧٨٩١٢٣٤٥٦

القيمة التي تخزن داخل متغير من نوع decimal هي : ١٢٣٤٥٦,٧٨٩١٢٣٤٥٦٧٨

حيث يتم تخزين الأعداد ولكن بدرجات دقة مختلفة تختلف من نوع إلى آخر

٣- إذا أردنا تخزين حرف واحد فقط فإننا نستخدم متغير من النوع char أما إذا أردنا تخزين مجموعة من الحروف نستخدم متغير من النوع string

هام إذا أردنا تخزين الرقم (٢٠٠) فإننا نحتاج إلى بايت واحد (١ byte) أى خلية واحدة لذلك نستخدم متغير من النوع (byte) أما إذا أردنا تخزين القيمة الحرفية " ٢٠٠ " فإننا نحتاج إلى ٣ بايت (٣ byte) لذلك يفضل أن نستخدم متغير من النوع (string)

تذكر أن

- يمكن ضبط الخصائص Properties من نافذة الخصائص Properties window أو برمجيا من نافذة الكود Code Window
- يقصد بضبط الخصائص Properties تحديد قيمة للخاصية Property وهذه القيمة لها نوع بيان محدد Data Type
- يمكن إظهار نافذة الخصائص من قائمة View ثم نختار Properties window
- يمكن فتح نافذة الكود من قائمة View الأمر Code

لاحظ

الخصائص Properties هي مكان تخزين البيانات ولها نوع (Type) أو نوع بيان (Data Type) .

- ١- وتختلف الخصائص حسب العنصر النشط فى شاشة الـ IDE
- ٢- يمكن إظهار نافذة الخصائص من خلال لوحة المفاتيح عن طريق الضغط على الزر F4
- ٣- الجزء الأيسر يحتوى على اسم الخاصية Text أما الجزء الأيمن يحتوى على قيمة الخاصية

معرفة نوع البيان (Data type) فى نافذة الكود

١. قف بمؤشر الفأرة على الخاصية Property بعد كتابة جملة التخصيص للخاصية Property فى نافذة الكود
٢. يظهر مستطيل أصفر به نوع البيان Data Type الخاص بهذه الخاصية Property ويظهر بعد كلمة AS

مثال

ضبط خصائص (Text , Auto Size , Width) الخاصة بإداه تحكم العنوان Label1

<pre>Label1.Text = "جمهورية مصر العربية"</pre> <pre>Public Overrides Property Text() As String</pre>	الخاصية (Text) نوعها (String)
<pre>Label1.AutoSize = False</pre> <pre>Public Overrides Property AutoSize() As Boolean</pre>	الخاصية AutoSize نوعها (Boolean)
<pre>Label1.Width = 120</pre> <pre>Public Property Width() As Integer</pre>	الخاصية Width نوعها (Integer)

لاحظ :

توفر لغة الفيجوال بيزيك أنواع أخرى (Types) غير (Data Types) .

تذكر ان

القيمة Value التي يتم تخصيصها للخاصية لها عدة أنواع أو أشكال منها:

١. القيمة المجردة مثال الخاصية Text
٢. القيمة المنطقية مثال الخاصية Visible
٣. القيمة من قائمة مثال الخاصية ForeColor
٤. القيمة بإنشاء كائن مثال الخاصية Font
٥. القيمة من ناتج تعبير حسابي
٦. القيمة من متغير Variable أو من خاصية Properties

صيغة ضبط الخصائص properties برمجيا

ControlName . Property = value

اسم أداة التحكم أو الكائن

الخاصية

القيمة

Label1.BackColor = Color.Red

Public Overridable Property BackColor() As System.Drawing.Color

المتغيرات

تعريفها

هي أماكن محجوزة في ذاكرة الكمبيوتر Ram ولها نوع بيان (Data Types) تتغير أثناء سير تنفيذ أوامر و تعليمات البرنامج

هام

- ١- يتم حجز الأماكن في ذاكرة الكمبيوتر بالإعلان عنها (أي إعطاء أسماء للخلايا التي تحمل قيم متغيرة)
- ٢- يتم تخصيص قيمة للمتغيرات على حسب نوع البيان Data Types
- ٣- هذه القيمة يمكن أن تتغير أثناء سير تنفيذ أوامر و تعليمات البرنامج .

لاحظ ان :-

١. قبل استخدام أي متغير داخل برنامج "Visual Basic" لابد من الإعلان عنه
٢. ويتم ذلك باستخدام صيغة معينة تسمى الإعلان عن المتغيرات وهذه الصيغة تبدأ بكلمة Dim

الصيغة العامة للإعلان عن المتغيرات في البرنامج

تستخدم الكلمة Dim ثم كتابة اسم المتغير ثم كلمة AS ثم نوع البيان Data Type كالتالي

نوع المتغير (نوع البيان Data Type) As (variable name) اسم المتغير Dim

الفصل الدراسي الثاني

كيفية اختيار نوع البيان Data Type المناسب لكل متغير

عند اختيار نوع البيان المناسب Data Type يراعى الآتي :

1. حجم البيانات التي سوف تخزن في المتغير Variable أي (حجم الذاكرة " عدد الـ Bytes " الذي يحتاجه البيان الذي سوف يخزن في المتغير)
2. تحديد نوع بيان القيم Data Type التي سوف يتم تخزينها بالمتغير Variable بناءً على العمليات الحسابية والمنطقية التي من الممكن إجراؤها على المتغير Variable .

النوع	المقصود به	طبيعة البيانات	السبب
Boolean	منطقي	قيمة true أو false	أحسن استغلال للذاكرة
Object	كائن او العام	كل الأنواع يمكن تخزينها هنا	لأنها أكثر مرونة و تستطيع ان تتكيف
String	حروف	مجموعة من الحروف أو أرقام لا تدخل في الحسابات (أرقام التليفون- الرقم القومي)	الاختيار الوحيد
Short	الأعداد الصحيحة	من -٣٢٧٦٧ الى ٣٢٧٦٧	استهلاك قليل للذاكرة كما انه سريع في الحسابات
Integer		من -٢١٤٧٤٨٣٦٤٨ الى ٢١٤٧٤٨٣٦٤٧+	يحتاج ذاكرة أقل من Long وأسرع في الحسابات
Long		أكبر من ٢١٤٧٤٨٣٦٤٧	الاختيار الوحيد
Single	أرقام تحتوي علي كسور عشرية	من -١٠ ^{٣٨} الى ١٠ ^{٣٨} أو سبع خانات دقة	أسرع في العمليات الحسابية من Double
Double		أكبر من ١٠ ^{٣٨} أو أكثر من سبع خانات دقة	الاختيار الوحيد
Decimal		مبلغ من المال	قدرة علي تخزين الجنيهات والقروش بدقة عالية وبدون أخطاء كما انه سريع في العمليات الحسابية

تدريب 1-1 (الإعلان عن المتغيرات) (Variables)

عزيزي الطالب قم بإعداد نافذة النموذج Form الآتية والتي من خلالها نستقبل مدخلات مستخدم

تسجيل مستخدم جديد

الاسم

تاريخ الميلاد

النوع ☐ ذكر ☐ أنثى

عدد أفراد الأسرة

تسجيل

القيم المتوقع إدخالها من مستخدم البرنامج ونوعها والتسمية المقترحة كما هو موضح بالجدول 1-1

القيم	نوع البيان Data Type	التسمية المقترحة
اسم المستخدم	نص String	U_Name
تاريخ الميلاد	تاريخ Date	U_B_D
النوع	منطقي Boolean	U_Gender
عدد أفراد الأسرة	رقم صحيح Integer	U_C_F

ثم اضغط على زر تسجيل مرتين متتاليتين لفتح نافذة الكود الخاصة بها ثم قم بكتابة معالج الحدث Event Handler والذي يسمى او يطلق عليه أيضا إجراء الحدث Event Procedure وقم بالإعلان عن المتغيرات كما بالشكل التالي :

```
Private Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs) Handles Button1.Click
    Dim U_Name As String
    Dim U_B_D As Date
    Dim U_Gender As Boolean
    Dim U_C_F As Integer
End Sub
```

لاحظ : معالج الحدث Event Handler يسمى او يطلق عليه أيضا إجراء الحدث Event Procedure

جملته التخصيص Assignment

- 1- ان أهم دور للمتغيرات في البرامج هو تخزين البيانات بها ولكي نخزن قيمة في متغير او خاصية نستخدم امر التخصيص
- 2- وهى جملة من طرفين بينهما علامة (=) وهذه العلامة لا يقصد بها علامة تساوي المعروفة في الحساب ولكن يطلق عليها معامل التخصيص
- 3- الطرف الأيسر لهذه العلامة هو المكان الذي توضع فيه القيمة أما الطرف الأيمن عبارة عن القيمة الصيغة العامة

الطرف الأيمن	علامة التخصيص	الطرف الأيسر
قيمة مجردة او قيمة من متغير Variable او خاصية Property او قيمة من تعبير Expression	=	متغير Variable او خاصية Property

لاحظ : جملة التخصيص Assignment :

هى طريقة لحجز أماكن فى ذاكرة الكمبيوتر RAM حتى يمكن لنا تخزين مدخلات مستخدم البرنامج فيها

يجب مراعاة الآتي:-

١. اسم المتغير الذى نريد التخزين فيه يكون على يسار " = "
٢. القيمة المراد تخزينها تكون يمين " = " وتكون من نوع ملائم لنوع المتغير
٣. ان قيمة المتغيرات التى على يمين " = " يتم تخزينها فى المتغير الموجود على يسارها
٤. ان قيمة المتغيرات التى على يمين " = لا تتغير وإنما قيمة المتغيرات الموجودة على شمال = هى التى تتغير وايضا ان القيمة الجديدة للمتغير تحل محل القيمة القديمة
٥. القيم المراد تخزينها يجب ان تكون من نوع ملائم للمتغير الذى سيتم التخزين به فالقيم الرقمية يجب ان تخرن فى متغيرات رقمية و القيم الحرفية يجب ان تخرن فى متغيرات حرفية وهكذا ..

أمثلة توضح التخصيص Assignment**قيمة مجردة**

Dim Number As Integer
Number = 5

تم الإعلان عن متغيرا اسمه Number من النوع Integer ثم تم تخصيص القيمة المجردة (5) للمتغير Number

قيمة متغير آخر

Dim Number1 As Integer
Dim Number 2 As Integer
Number1 = 5
Number2 = Number1

تم الإعلان عن متغيرين وهما Number1 و Number2 من النوع Integer ثم تم تخصيص القيمة المجردة (5) للمتغير Number1 و تم تخصيص قيمة المتغير Number1 الى المتغير Number2

قيمة من خاصية

Dim Name As String

Name = Label1.Text

تم الإعلان عن متغير Name من النوع String ثم تم تخصيص القيمة من الخاصية Text لأداة التحكم Label1

قيمة من تعبير حسابى

Dim Number As Single
Number = 7
Number = Number / 3

تم الإعلان عن متغير Number من النوع Single ثم تم تخصيص القيمة المجردة 7 للمتغير Number ثم إعادة عملية التخصيص للمتغير Number من خلال تعبير (Expression) وهو (Number/3)

لاحظ : يمكن الإعلان عن المتغير و تخصيص قيمة له أثناء الإعلان فى خطوه واحده

Dim Number As Single = 5.6

تم الإعلان عن متغير Number من النوع Single وتم تخصيص القيمة 5.6 أثناء الإعلان عن المتغير Number فى خطوة واحدة.

ملاحظات هامة :

- يمكن التخصيص للمتغيرات Variables أثناء الإعلان عنها أو في مرحلة لاحقة.
- يمكن إعادة التخصيص أي عدد من المرات ولذلك أطلق لفظ المتغيرات لأن قيمها يمكن أن تتغير أثناء سير عمليات البرنامج في أي وقت.
- التخصيص يمكن أن يكون بقيمة مجردة أو بمتغير أو بخاصية أو بتعبير.
- علامة = لا يقصد بها التساوي الحسابي وإنما يقصد بها علامة للتخصيص كما بالمثل $x = x + 1$
- حيث (x) الموجودة بالطرف الأيسر تمثل المخزن و (x) الموجودة بالطرف الأيمن تمثل القيمة حيث إذا افترضنا أن قيمة (x) تساوي (5) إذا بعد عملية التخصيص يصبح المتغير (x) يساوي (5+1) أي (6)

تخصيص قيم للمتغيرات Variables

سوف نقوم بتخصيص مدخلات مستخدم البرنامج للمتغيرات Variables التي سبق إعلانها في التدريب رقم ١ ص ٧ و ٨ وهذه المتغيرات هي

U_C_F	U_Gender	U_B_D	U_Name
-------	----------	-------	--------

في هذه الحالة سوف نستخدم خصائص Properties الأدوات Controls التي استقبلت مدخلات مستخدم البرنامج في عملية التخصيص حيث أننا لم نغير قيم خاصية Name لهذه الأدوات كما هو

الخاصية Property	اسماء أدوات التحكم Controls
Text	TextBox1
Text	TextBox2
Checked	RadioButton1
Text	TextBox3

جمل التخصيص

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim U_Name As String
    Dim U_B_D As Date
    Dim U_Gender As Boolean
    Dim U_C_F As Integer

    U_Name = TextBox1.Text
    U_B_D = TextBox2.Text
    If RadioButton1.Checked = True Then
        U_Gender = True
    End If
    If RadioButton1.Checked = False Then
        U_Gender = False
    End If
    U_C_F = TextBox3.Text
End Sub
```

لاحظ

تم اختبار الخاصية Checked لأداة التحكم RadioButton1 باستخدام جملة If وسوف يتم شرحها لاحقا ، فإذا كانت الخاصية Checked قيمتها True يخصص للمتغير U_Gender القيمة True أما إذا كانت العكس يخصص للمتغير U_Gender القيمة False

لاحظ

يوجد اختلاف بين نوع بيان Data Type و الخصائص Properties عن المتغيرات المستخدمة Variables كما هو موضح بالجدول التالي:

الخاصية	نوع بيان الخاصية	المتغير	نوع بيان المتغير
TextBox1.Text	String	U_Name	String
TextBox2.Text	String	U_B_D	Date
TextBox3.Text	String	U_C_F	Integer

هام:

- لا يجوز أن يختلف نوع بيان الخصائص عن نوع بيان المتغيرات المستخدمة Variables
- لذا فتتيح لغة الفيجوال بيزيك إمكانية تحويل القيم إلى نوع بيان متوافق مع المتغير Variable أو الخاصية Property المخصص لها القيمة وهو ما يسمى تحويل ضمني

سؤال : ما المقصود بالتحويل الضمني ؟

هو إمكانية توفرها لغة الفيجوال بيزيك لتحويل القيم إلى نوع بيان Data Type متوافق مع المتغير Variable أو الخاصية Property المخصص لها القيمة

استخدام المتغيرات Variables

إذا وجد المتغير في الطرف الأيسر من جملة التخصيص فهذا يعني أنه المكان الذي سوف تخزن به القيمة أما إذا وجد في الطرف الأيمن من جملة التخصيص فهذا يعني أنه يعبر عن القيمة الموجودة لديه.

- يمكن استعمال متغير على يمين علامة (=) ومتغير على يسارها في نفس الأمر
- حيث المتغير الموجود على يسار العلامة (=) هو مكان تخزين القيمة
- بينما المتغير الموجود على يمينها هو القيمة

Price = 50

Price = Price + 10

تدريب على استخدام المتغيرات Variables

الخاصية	القيمة
AutoSize	False
Size, Location	باستخدام مؤشر الفأرة
BorderStyle	FixedSingle

قم بإضافة أداة تحكم عنوان Label بعد ضبط خصائصها كما هو موضح بالجدول

```

Private Sub Button1_Click(ByVal
    Dim U_Name As String
    Dim U_B_D As Date
    Dim U_Gender As Boolean
    Dim U_C_F As Integer

    U_Name = TextBox1.Text
    U_B_D = TextBox2.Text
    If RadioButton1.Checked = True Then
        U_Gender = True
    End If
    If RadioButton1.Checked = False Then
        U_Gender = False
    End If
    U_C_F = TextBox3.Text

    Me.Label5.Text = U_Name & vbCrLf & U_B_D & vbCrLf & _
    U_Gender & vbCrLf & U_C_F

End Sub

```

الإعلان عن المتغيرات

تخصيص قيم للمتغيرات

استخدام المتغيرات

قم بفتح إجراء الحدث (Click) الخاص بالزر " تسجيل " ثم أضف الكود (Code) الآتي

```

Me.Label5.Text = U_Name & vbCrLf & U_B_D & vbCrLf & U_gender & _
vbCrLf & U_C_F

```

الاحظ:

- ١- الكود السابق عبارة عن أمر تخصيص مجموعة متغيرات بقيمة للخاصية Text لأداة التحكم Label 5 حيث توجد المتغيرات على يمين علامة (=) إذن فهي القيمة
- ٢- يفصل بين كل متغير والآخر (& vbCrLf &)
- ٣- علامة (معامل الربط) (&) تستخدم للربط بين النصوص.
- ٤- الكلمة المحجوزة vbCrLf عبارة عن ثابت حرفي يستخدم لإضافة رمز مفتاح الإدخال وإنشاء سطر جديد
- ٥- تستخدم علامة (_) حتى يمكن كتابة سطر الكود على أكثر من سطر في حالة إذا كان سطر الكود Code طويل بعض الشيء وذلك للتنظيم وتسهيل عملية قراءة الكود Code
- ٦- كلمة Me من الكلمات المحجوزة تعبر عن نافذة النموذج Form الحالية.

بعد الانتهاء من كتابة الكود (Code) نقوم بتشغيل البرنامج أو اختبار البرنامج

١- من قائمة (Debug) اختر (Start Debugging) .

٢- الضغط على أيقونة (Start icon) من شريط الأدوات .

٣- اضغط على (F5) من لوحة المفاتيح (لتشغيله من فيجوال استوديو) .

ونملاً بيانات استمارة التسجيل ثم نقوم بالنقر على مفتاح تسجيل كما هو موضح :

لاحظ :

ظهور محتوى أدوات
التحكم على أداة
العنوان Label 5

كتابة الملاحظات

١- توفر لنا لغة Visual Basic .NET إمكانية كتابة ملاحظات أو تعليقات حيث تساعد القارئ على فهم

وتوضيح الأكواد المكتوبة في نافذة الكود Code Window

٢- يتم كتابة الملاحظات في نافذة الكود Code Window باستخدام الكلمة REM أو العلامة (')

٣- كل ما يكتب بعد هـ لا يعتبر أكواد ويهمل عند ترجمة الأكواد Code بواسطة مترجم اللغة Compiler

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As Sy
    REM الإعلان عن المتغيرات
    Dim UserName As String
    Dim UserBirthDate As Date
    Dim UserGender As Boolean
    Dim UserNoFamily As Integer

    REM تخصيص قيم لهذه المتغيرات
    UserName = Me.TextBox1.Text
    UserBirthDate = Me.TextBox2.Text
    If RadioButton1.Checked = True Then
        UserGender = True
    Else
        UserGender = False
    End If
    UserNoFamily = Me.TextBox3.Text

    ' استخدام هذه المتغيرات
    Me.Label5.Text = UserName & vbCrLf & UserBirthDate & vbCrLf & _
        UserGender & vbCrLf & UserNoFamily
End Sub
```


الثوابت Constants

تعريفها

هي عبارة عن أماكن محجوزة بذاكرة الكمبيوتر RAM ولها نوع بيان Data Type ولها قيمة ثابتة وهذه القيمة لا يمكن أن تتغير أثناء سير تنفيذ أوامر وتعليمات البرنامج

هام

١- يتم الإعلان عن الثوابت ثم تخصيص قيمة لها حسب نوع بيان Data Type أثناء الإعلان عنها فقط

٢- قيمة الثابت Constant (ثابتة) لا يمكن أن تتغير أثناء سير تنفيذ أوامر وتعليمات البرنامج

تعريف آخر

الثوابت : هي مخزن في ذاكرة الكمبيوتر يحمل القيم التي لا تتغير أثناء تشغيل البرنامج ويجب الإعلان عنها داخل البرنامج واعطاء قيمة لها أثناء الإعلان عنها

الصيغة العامة للإعلان عن الثوابت في البرنامج

تستخدم جملة Const ثم كتابة اسم للثابت Constant name ثم كلمة As ثم نوع البيان Data Type ثم علامة = ثم القيمة كما هو مبين كالآتي

القيمة Value = نوع الثابت (نوع البيان Data Type) As (constant name) اسم الثابت Const

أمثلة

١- الإعلان عن ثابت C_Name له نوع بيان String وتم تخصيص القيمة النصية " جمهورية مصر العربية " أثناء الإعلان

Const C_Name As String = " جمهورية مصر العربية "

٢- الإعلان عن ثابت pi له نوع بيان Single وتم تخصيص القيمة الرقمية 22 / 7 أثناء الإعلان.

Const pi As Single = 22 / 7

٣- الإعلان عن ثابت BirthDate له نوع بيان Date وتم تخصيص قيمة التاريخ #1/25/2011# أثناء الإعلان.

Const BirthDate As Date = #1/25/2011#

لاحظ

تستخدم علامتي " " في حالة إذا ما أردنا كتابة نص
تستخدم علامتي # # في حالة إذا ما أردنا كتابة تاريخ أو وقت

تذكر فتح نافذة الكود

١- اضغط على مفتاح F7 من لوحة المفاتيح

٢- قم بتنشيط الأداة المراد فتح نافذة الكود لها ثم نختار من قائمة View الأمر Code

٣- قم بالضغط مرتين متتاليتين D- Click على الأداة المراد فتح نافذة الكود لها

٤- اضغط بزر الفأرة الأيمن ثم اختر من القائمة المختصرة View Code

تدريب: الإعلان عن الثوابت Constants

١- سوف نقوم بإعداد نافذة نموذج

Form كما هو موضح

٢- ثم نقوم بفتح نافذة الكود Code window بالضغط على مفتاح F7 ثم

إضافة إجراء الحدث Click لكل من زري الأمر Button1, Button2

```
Public Class Form3
    Private Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs)
        ' Code for Button1_Click
    End Sub
    Private Sub Button2_Click(ByVal sender As Object, ByVal e As EventArgs)
        ' Code for Button2_Click
    End Sub
End Class
```

علما بأن :

مساحة الدائرة πR^2 محيط الدائرة $2\pi R$

حيث R تمثل نصف القطر

$$\pi = 22/7$$

لاحظ :

سوف نكتب الكود Code الخاص بحساب مساحة الدائرة باستخدام إجراء الحدث Button1_Click ومحيط الدائرة باستخدام إجراء الحدث Button2_Click وتظهر النتيجة داخل أداة التحكم عنوان Label2 عند النقر على أحدهما.

في نطاق إجراء الحدث Button1_Click نكتب الكود Code التالي :

```
Private Sub Button1_Click(ByVal sender As
```

شرح الكود :

```
Dim Radius As Single
```

١. تم الإعلان عن متغير Radius من النوع Single

```
Const pi As Single = 22 / 7
```

٢. تم الإعلان عن ثابت pi من النوع Single وتم وضع القيمة 22/7

```
Radius = TextBox1.Text
```

٣. تم تخصيص القيمة من الخاصية TextBox1.Text للمتغير Radius

```
Label2.Text = pi * Radius ^ 2
```

٤. تم تخصيص التعبير $\pi * \text{Radius}^2$ للخاصية Label2.Text

```
End Sub
```

تطبيقا للقانون الرياضي الخاص بمساحة الدائرة

بعد الانتهاء من كتابة الكود Code نقوم بالضغط

على مفتاح F5 لعمل Start Debugging

ونكتب قيمة نصف القطر ثم نقوم بالنقر على زر

"مساحة الدائرة"

تدريب (5) تحديد نطاق إعلان المتغيرات (Variables) والثوابت (Constants)

يقصد بنطاق المتغيرات أو الثوابت موضع الإجراء الذي يتم فيه الإعلان عن الثابت أو المتغير واستخدامه.

قم بإضافة الكود Code الخاص بحساب محيط الدائرة وذلك في نطاق إجراء الحدث Button2_Click نكتب الكود Code الآتي:

Radius = TextBox1.Text

Label2.Text = pi * Radius * 2

ليصبح شكل الكود كما يلي:

```
Private Sub Button1_Click(ByVal sender As Object)
    Dim Radius As Single
    Const pi As Single = 22 / 7
    Radius = TextBox1.Text
    Label2.Text = pi * Radius ^ 2
End Sub
```

```
Private Sub Button2_Click(ByVal sender As Object)
    Name 'Radius' is not declared.
    Radius = TextBox1.Text
    Label2.Text = pi * Radius * 2
End Sub
```

كما يظهر المستطيل الأصفر عند توجيه مؤشر الفأرة إلى المتغير pi في السطر الثاني كما هو موضح

```
Private Sub Button2_Click(ByVal sender As Object)
    Radius = TextBox1.Text
    Name 'pi' is not declared.
    Label2.Text = pi * Radius * 2
End Sub
```

لاحظ :

- 1- أننا لم نعلن عن المتغير والثابت في نطاق إجراء الزر Button 2
 - 2- تم الاعتماد على أن المتغير Radius والثابت pi سبق إعلانهما في نطاق إجراء الحدث Button1_Click
 - 3- تظهر رسائل داخل مستطيل أصفر تنبه أن المتغير Radius والثابت pi لم يتم الإعلان عنهما في نطاق إجراء الحدث Button2_Click رغم أنهما معلنين في نطاق إجراء الحدث Button1_Click
- استنتاج هام : أن المتغيرات والثوابت لا نستطيع استخدامها إلا في نطاق إعلانها

ويمكن حل هذه المشكلة بطريقتين (الطريقة الأولى)

يمكن أن نعيد إعلان المتغير Radius والثابت pi مرة أخرى في نطاق إجراء حدث الزر Button2_Click لأنه لا يمكن استخدامهما إلا في حدود نطاق الإجراء فقط (المتغيرات المحلية Local Variables)

```

Private Sub Button1_Click(ByVal sender As Object,
    Dim Radius As Single
    Const pi As Single = 22 / 7
    Radius = TextBox1.Text
    Label2.Text = pi * Radius ^ 2
End Sub

Private Sub Button2_Click(ByVal sender As Object,
    Dim Radius As Single
    Const pi As Single = 22 / 7
    Radius = TextBox1.Text
    Label2.Text = pi * Radius * 2
End Sub

```

(الطريقة الثانية)

كما يمكننا أن نعلن عن المتغيرات Variable والثوابت Constant على مستوى التصنيف Class ومنها لا نحتاج عملية الإعلان عند كل نطاق إجراء حدث (Event Procedure) (المتغيرات العامة Global Variables)

```
Public Class Form3
```

```
    Dim Radius As Single
    Const pi As Single = 22 / 7
```

```
    Private Sub Button1_Click(ByVal sender As
        Radius = TextBox1.Text
        Label2.Text = pi * Radius ^ 2
    End Sub
```

```
    Private Sub Button2_Click(ByVal sender As
        Radius = TextBox1.Text
        Label2.Text = pi * Radius * 2
    End Sub
```

```
End Class
```

الشرح

حيث تم الإعلان عن المتغير Radius والثابت pi في نطاق التصنيف Form3 فاستطعنا استخدامهما داخل نطاق إجراء أحداث Button1_Click و Button2_Click لأن إجراء أحداث Button1_Click و Button2_Click معلنه في نطاق التصنيف Form3 أيضا

نطاق المتغيرات :

يحدد نطاق " مجال " المتغير اى الأماكن المسموح بها استعمال المتغير داخل البرنامج وتنقسم المتغيرات من حيث النطاقات او المجالات الى نوعان

١- المتغيرات المحلية (Local Variables) :

وهى متغيرات يتم الاعلان عنه داخل إجراء حدث معين مثل الحدث Button1_Click وينفذ داخل هذا الإجراء فقط ويفقد قيمته اذا ما تم استعماله خارج هذا الإجراء مثل الحدث Button2_Click

٢- المتغيرات العامة " الشاملة " (Global Variables) :

هى متغيرات يتم الإعلان عنه خارج جميع الإجراءات فى نطاق التصنيف Form3 وهى يمكن استخدامها فى كل إجراءات البرنامج و لا يفقد قيمتها عند انتهاء الإجراء المستخدمة به

الأخطاء Errors

لا بد من الإلتزام بقواعد لغة البرمجة المستخدمة حيث أنه عند كتابة الكود Code يحدث عدة أنواع من الأخطاء منها

أنواع الأخطاء		
الإملانية والنحوية Syntax Errors	المنطقية Logic Error	عند التشغيل Runtime Error

١- الأخطاء والنحوية الإملانية Syntax Errors

هى الأخطاء التي تحدث عند كتابة الكود Code بصورة غير سليمة كما هو موضح بالأمثلة الآتية:

امثلة

Dim x As Single

تم الإعلان عن المتغير x ولكن كتبت كلمة Dim خطأ

Const x As Single

تم الإعلان عن الثابت x ولكن لم تخصص له قيمة أثناء الإعلان

كيفية التغلب على الأخطاء الإملانية والنحوية

للتغلب على هذا النوع من الأخطاء يساعدنا IDE بيئة التطوير المتكاملة بانها لا تسمح بوجود أي خطأ من هذا النوع. وتظهر خط متعرج تحت الخطأ

٢- الأخطاء المنطقية Logic Error

هى الأخطاء التي تؤدي إلى نتائج غير سليمة عند استخدام البرنامج وتحدث إذا استخدمت تعبيرات Expressions بناؤها غير سليم في جمل التخصيص غالباً

عند حساب مساحة الدائرة استخدمنا الكود الآتي:

امثلة

Dim Radius As Single
Const x As Single = 22 / 7
Radius = TextBox1.Text
Label2.Text = x * Radius ^ 2

Dim Radius As Single

Const x As Single = 22 / 7

Radius = TextBox1.Text

Label2.Text = x + Radius ^ 2



عند تنفيذ الكود Code سوف يعمل دون أن يعطي أي رسائل خطأ ولكن النتيجة سوف تصبح مغايرة للواقع لأنه سوف يتم الجمع بدلا من الضرب.

كيفية التغلب على الأخطاء المنطقية Logic Error

١- يجب مراعاة الكود المكتوب جيدا

٢- اختبار البرنامج على بيانات سبق التأكد من صحتها حتى يتم التأكد من عدم وجود أخطاء من النوع Logic Error

٢- الأخطاء عند التشغيل Runtime Error

هي الأخطاء التي تظهر أثناء التشغيل ونجدها غالبا في الأكواد Code التي يتم فيها جملة تقصيص Assignment

متى تحدث :

١. تحدث عند تقصيص قيمة أكبر من مدى نوع البيان المستخدم

٢. تحدث عند تقصيص قيمة نوعها يختلف عن نوع بيان المتغير أو الثابت أو الخاصية ولا يستطيع مترجم Compiler اللغة تحويلها لهذا النوع Type

أمثلة على الأخطاء التي تظهر أثناء التشغيل

٢- أكتب الكود Code الآتي لدى إجراء

الحدث Click للزر "مجموع"

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    'الإعلان عن متغير لكل مادة'
    Dim arabic As Byte
    Dim computer As Byte

    'تخصيص مدخلات المستخدم لكل متغير'
    arabic = TextBox1.Text
    computer = TextBox2.Text

    'تخصيص ناتج مجموع المتغيرين'
    Label3.Text = arabic + computer
End Sub
```

١- قم بإعداد نافذة النموذج كما هو موضح بشكل

لاحظ:

تم الإعلان عن متغيرين arabic (و) computer لهما نوع البيان byte والقيم التي يمكن أن يستقبلها تبدأ من صفر وتنتهي عند 255 وهذا يعني أنه يجب على مستخدم البرنامج أن يلتزم بحدود تلك القيم وفي حالة مخالفتها يحدث خطأ أثناء التشغيل Runtime عند إجراء التخصيص لهذه المتغيرات Variables كما يظهر في الحالات الآتية:

الحالة الأولى:

١- قم بعمل Start Debugging وذلك بالضغط على مفتاح F5

ثم قم بالنقر على مفتاح "مجموع" دون إدخال أي بيانات

لصناديق النص TextBox

٢- تظهر لنا رسالة الخطأ الآتية أنه لا يجوز تحويل القيمة النصية الفارغة "" إلى النوع Byte وحدث ذلك بسبب أننا لم ندخل درجة اللغة العربية

رسالة خطأ أثناء وقت التشغيل بسبب عدم ادخال أي قيمة

تخصيص مدخلات المستخدم لكل متغير

```
arabic = TextBox1.Text
computer = TextBox2.Text
```

InvalidCastException was unhandled

Conversion from string "" to type 'Byte' is not valid.

يُصنّف ناتج مجموع المتغيرين

```
Label3.Text = arabic + computer
```

الحالة الثانية

قم بإيقاف التشغيل Stop Debugging وذلك بالضغط على أيقونة

وذلك بالضغط على مفتاح F5 ثم قم بالنقر على مفتاح مجموع بعد إدخال قيمة نصية (خمس عشرة درجة) لصندوق

النص TextBox

كمبيوتر 1 خمسة عشر

تظهر لنا رسالة الخطأ الآتية أنه لا يجوز تحويل القيمة النصية "خمس عشرة درجة" إلى النوع Byte حدث ذلك بسبب إدخال

قيمة نصية بدلا من قيمة رقمية

رسالة خطأ أثناء وقت التشغيل بسبب إدخال نص.

تخصيص مدخلات المستخدم لكل متغير

```
arabic = TextBox1.Text
computer = TextBox2.Text
```

InvalidCastException was unhandled

Conversion from string "خمس عشرة درجة" to type 'Byte' is not valid.

يُصنّف ناتج مجموع المتغيرين

```
Label3.Text = arabic + computer
```

الحالة الثالثة

قم بإيقاف التشغيل Stop Debugging وذلك بالضغط على أيقونة

وذلك بالضغط على مفتاح F5 ثم قم بالنقر على مفتاح مجموع بعد إدخال قيمة أكبر من 255 لصندوق النص TextBox

رسالة الخطأ الآتية نتيجة إدخال قيمة أكبر من الحد المسموح وحدث ذلك بسبب إدخال درجة الكمبيوتر أكبر من 255 أي

القيم خارج الحدود المسموحة

تخصيص مدخلات المستخدم لكل متغير

```
arabic = TextBox1.Text
computer = TextBox2.Text
```

كمبيوتر 300

OverflowException was unhandled

Arithmetic operation resulted in an overflow.

يُصنّف ناتج مجموع المتغيرين

```
Label3.Text = arabic + computer
```

516

لاحظ:

الشكل البيضاوي المرسوم على كل رسالة خطأ يشير إلى نوع الخطأ الذي لم يتم معالجته.

كيفية التغلب على أخطاء التشغيل Runtime Error

الطريقة الأولى

توفر لغة Visual Basic .NET إمكانية معالجة الأخطاء المحتمل وقوعها من خلال جملة Try/ Catch

مجموعة أوامر Try / Catch

تستخدم لاكتشاف الأخطاء التي تحدث أثناء تشغيل البرنامج والتعامل معها

١. يقوم VB .net باكتشاف الأخطاء ويقوم بعرض رسائل تفيد بحدوث خطأ

٢. لكن المبرمجين يفضلون استعمال Try / Catch

□ لأنها تتيح لهم التعامل مع الأخطاء بأنفسهم

□ عرض رسائل أوضح لمستخدم البرنامج يقوم المبرمج بكتابتها

الصيغة العامة لها :-

Try

الكود Code المحتمل أن يظهر منه خطأ أثناء التشغيل

Catch ex As Exception

كود Code الخاص بمعالجة الخطأ إذا وقع وغالبا يكون رسالة

EndTry

بقية أوامر البرنامج

شرح الأمر

١. اذ قمنا بتنفيذ الأوامر بعد Try بدون أخطاء ينتقل البرنامج الى الأوامر بعد End Try مباشرة ويتم تجاهل

الأوامر التي تلي Catch.

٢. أما إذا تم تنفيذ الأوامر بعد Try بوجود أخطاء ينتقل البرنامج الى الأوامر بعد Catch لعرض الرسالة الموجودة

بها ولا يستكمل تنفيذ ما بقي من الأوامر التي تلي Try الإعلان عن متغير لكل مادة

```
Dim arabic As Byte
Dim computer As Byte
```

Try

تخصيص مدخلات المستخدم لكل متغير

```
arabic = TextBox1.Text
computer = TextBox2.Text
```

Catch ex As Exception

```
Label3.Text = "من فضلك أدخل الدرجات بصورة صحيحة"
```

Exit Sub

End Try

تخصيص ناتج مجموع المتغيرين

```
Label3.Text = arabic + computer
```

الطريقة الثانية

التحكم في مدخلات المستخدم بحيث نوفر للمستخدم الأدوات Controls مع ضبط خصائصها Properties التي من خلالها يقوم المستخدم بإدخال البيانات الصحيحة دون غيرها (أى إلزام المستخدم على الاختيار من ضمن البيانات الصحيحة المعروضة فقط دون كتابة أى بيانات أخرى)

لاحظ :

في نافذة النموذج السابقة لا يستطيع المستخدم اختيار دولة ليست موجودة في القائمة.

لاحظ :

وهذه أدوات تقدمها لغة البرمجة للمبرمج ليفرض بها اختيارات محددة للمستخدم حتى يضمن تقليص عدد الأخطاء إلى أدنى حد

الجدول الآتي يوضح العمليات الحسابية التي يمكن إجراؤها في vb.net

العمليات الحسابية

العملية	جمع	طرح	ضرب	قسمة عادية	باقي القسمة	أسس
المعامل المؤثر	+	-	*	/	mod	^

أولويات تنفيذ العمليات الحسابية

جميع لغات البرمجة بما فيها vb.net قد وضعت قاعدة تسمى قاعدة أولويات تنفيذ العمليات الحسابية لتحديد أولوية تنفيذ العملية الحسابية وفي هذه القاعدة يتم ترتيب المعاملات المختلفة ترتيباً تنازلياً حسب أولوية تنفيذها عند اجتماعها في عملية حسابية واحدة

فمننا بإجراء بعض عمليات التخصيص ومنها تخصيص تعبير حسابي مثل

Label2.Text = x * Radius * 2

في حالة إذا ما وجد أكثر من عملية حسابية مثل

5+3*2

أيهما ينفذ أولاً عملية الضرب أم عملية الجمع ؟

يتم تنفيذ عملية الضرب أولاً ثم يجمع الناتج بحيث تصبح النتيجة (11) ولكن إذا وضعنا أقواس بهذه الطريقة (5+3) * 2 ينفذ ما بداخل القوس أولاً فتصبح النتيجة (16)

تنفذ العمليات الحسابية وفقا للترتيب الآتي

١. العمليات التي بين الأقواس (الداخلية أولا ثم الخارجية)
٢. عمليات الأس
٣. عمليات الضرب و القسمة (الأولوية من اليسار ثم اليمين)
٤. عمليات الجمع و الطرح (الأولوية من اليسار إلي اليمين)

ملحوظة هامة جدا

في حالة وجود أكثر من عملية لها نفس الأولوية كالضرب والقسمة او الجمع والطرح فان ترتيب تنفيذها يكون من اليسار الى اليمين ، أما إذا تداخلت الأقواس (كأن يكون هناك أقواس داخل أقواس) فان الأقواس الداخلية يتم تنفيذ ما بينها أولا

خلى بالك : في حالة وجود عمليات حساب باقى القسمة فانها تأتى بعد الضرب و القسمة و قبل الجمع و الطرح

ملحوظة هامة جدا

من الضروري مراعاة أولويات تنفيذ العمليات الحسابية عند إنشاء تعبيرات Expression تستخدم في جملة التخصيص Assignment حتى لا تقع أخطاء من النوع Logic Errors لأن النتيجة ستتغير في هذه الحالة دون ظهور أي رسائل خطأ

مثال :-

ما هي نتيجة تنفيذ العمليات الآتية بافتراض القيم التالية للمتغيرات :-

$$X = 2 \quad Y = 3 \quad Z = 4$$

A) $X * (Y + 1)$
B) $Y \wedge X * X + Z * 3$

(A الحل)

١- نستبدل كل متغير بقيمته

$$2 * (3 + 1)$$

٢- حساب ما بداخل الأقواس أولا

$$2 * (4)$$

٣- تنفيذ عملية الضرب فيكون الناتج (8)

(B

١. نستبدل كل متغير بقيمته $3 \wedge 2 * 2 + 4 * 3$

٢. يتم حساب الاس أولا $9 * 2 + 4 * 3$

٣. تنفيذ عملية الضرب التي في اليسار $18 + 4 * 3$

٤. تنفيذ عملية الضرب التي في اليمين $18 + 12$

٥. تنفيذ الجمع ليكون الناتج 30

أسئلة وتدريبات

السؤال الأول:

في نافذة النموذج الآتية:

تسجيل مستخدم جديد

الاسم: مروة ماهر

تاريخ الميلاد: 20 يونيو , 1998

النوع: ☒ انثى ☐ ذكر

الجنسية: مصر

تسجيل

إذا أردنا أن نخزن مدخلات المستخدم في متغيرات حدد نوع البيان المناسب لكل مدخل من مدخلات مستخدم البرنامج:

- ١
- ٢
- ٣
- ٤

السؤال الثاني:

في الكود (Code) الآتي:

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim Radius As Single
    Const x As Single = 22 / 7
    Radius = TextBox1.Text
    Label2.Text = x * Radius ^ 2
End Sub
```

استخرج أسماء المتغيرات والثوابت وأنواعها:

السؤال الثالث:

في الأكواد الآتية حدد سبب الخطأ

تخصيص مدخلات المستخدم لكل متغير

arabic = TextBox1.Text

computer = TextBox2.Text

يتم ناتج مجموع المتغيرين

Label3.Text = arabic + computer

InvalidCastException was unhandled

Conversion from string "خمس عشرة درجة" to type 'Byte' is not valid.

السبب:

تخصيص مدخلات المستخدم لكل متغير

arabic = TextBox1.Text

computer = TextBox2.Text

تخصيص ناتج مجموع المتغيرين

Label3.Text = arabic + computer

Sub

OverflowException was unhandled

Arithmetic operation resulted in an overflow.

السبب:

السؤال الرابع:

حدد ماهو الكود (Code) المطلوب كتابته حسب ما تشير اليه الأسهم:

Try



Catch ex As Exception



EndTry

السؤال الخامس:

حدد نوع الخطأ في الكود (Code) الآتي ثم صوبه

الكود (Code)	نوع الخطأ	التصويب
Din x As Single		
Const x As Single		

السؤال السادس:

عند كتابة الكود يحتاج المبرمج لوضع تعليق معين لا يتم تنفيذه، لابد أن يسبقه :

-٢

أو

-١

السؤال السابع:

في الكود (Code) الآتي:

```
Private Sub Button1 Click(ByVal sender As System.Object, ByVal e As S
Dim UserName As String
Dim UserBirthDate As Date
Dim UserGender As Boolean
Dim UserTelNo As Integer

UserName = TextBox1.Text
UserBirthDate = TextBox2.Text
UserGender = CheckBox1.CheckState
UserTelNo = TextBox3.Text
```

١

٢

فسر الكود المشار إليه بالأرقام على الشكل

- ١

- ٢

السؤال الثامن:

رتب السطور الآتية وفقاً لأولويات تنفيذ العمليات الحسابية

العملية	الترتبة
الضرب والقسمة من اليسار إلى اليمين.	()
الأقواس من الداخل إلى الخارج.	()
الجمع والطرح من اليسار إلى اليمين.	()
الأسس.	()

السؤال التاسع:

حدد ناتج تنفيذ العمليات الحسابية الآتية:

أ - $(5+3)*2$

النتيجة:

ب - $5+3*2$

النتيجة:

السؤال العاشر:

ضع علامة (✓) أمام العبارة الصحيحة، وعلامة (✗) أمام العبارة الخاطئة:

(١)	أحد قواعد تسمية المتغيرات أو الثوابت في البرنامج أن يبدأ اسم المتغير بحرف أو رقم. (...)
(٢)	يتم الإعلان عن المتغيرات باستخدام جملة Dim. (...)
(٣)	المتغير من النوع Double يأخذ القيمة True أو False. (...)
(٤)	المتغيرات من أنواع (Integer & Long & Double) تستخدم لتخزين الأعداد الصحيحة فقط. (...)
(٥)	يتم استقبال مدخلات المستخدم من خلال العديد من الأدوات منها TextBox. (...)

تدريب: أوجد ناتج تنفيذ العمليات التالية داخل VB.Net :

- $8 + 5 * 3$
- $(8 + 5) * 3$
- $16 / 4 + 8 + 6 * 8$
- $X = 10 + 3 / 3 * 2 ^ 2 \bmod 4$
- $Y = 16 ^ 0.5 \bmod 3$
- $Z = 5 + 5 - 10 * 3 / 6$
- $X = (3 + 7) / 2 + 3 / 6 * 2$
- $Y = 1 + (2 + 2) ^ 2 / 4 * 2$
- $Z = 10 + 3 * 5 / 3$
- $X = 9 + 9 / 3 / (9 \bmod 4)$
- $Y = 16 + 5 - 10 / (2 + 3)$
- $Z = 18 / 9 ^ 0.5 / 3$
- $X = 13 - 3 / 3 + 2 * 5$

إذا علمت أن $A = 1$ و $B = 2$ و $C = 3$ أوجد ناتج كل من العمليات التالية :

- $A ^ 2 - B * C / (A + 2)$
- $(A + 2B) / (C + B)$
- $A + B / B + C$
- $(A + 3 * B / C) ^ 2 - 2 + 3 * B$
- $(5 * 1 + A) / C \bmod B$
- $10 / 2 * B * C / 2 * B / A$
- $A + B - 8 + 9 / 3 \bmod 4 ^ 0.5$
- $A - B / -1$
- $- A - C / -2$
- $- 2 * - A / 2 + 3 * C \bmod 2$

** الفصل الثاني **

التفرع Branching

مقدمة

- درست في ان التفرع في خرائط التدفق هو اتخاذ قرار أو اختيار لتنفيذ مجموعة خطوات معينة أو خطوات أخرى بناء على سؤال معين في حالة ان السؤال له احتمالين أو أكثر
- وسوف تجد أن كتابة الكود Code الخاص بعملية التفرع ماهي إلا خطوات الحل الخوارزمية Algorithm مع الالتزام بقواعد اللغة المستخدمة وطريقة صياغتها
- وللتعبير عن التفرع برمجيا نستخدم جمل معينة في لغة البرمجة مثل جملة If...Then و Select...Case وغيرها..

التفرع باستخدام جملة If...Then

الصيغة العامة لهذه الجملة:

If conditional Expression (شرط أو تعبير شرطي) Then

مجموعة الأوامر التي يتم تنفيذها عندما يكون ناتج الشرط (True) صواب

End if

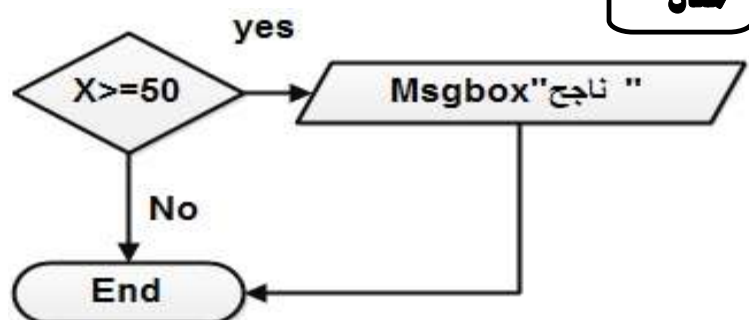
تكملة البرنامج

شرح الصيغة العامة

- ١- يبدأ التفرع باستخدام جملة If...Then بكلمة If وما بعدها هو الشرط أو التعبير الشرطي Conditional Expression ثم كلمة Then على نفس السطر و تقابل كلمة اذن في خطوات الحل او Yes في خرائط التدفق
- ٢- ينتهي التعبير الشرطي بكلمة End if
- ٣- في حالة ما يكون ناتج التعبير الشرطي صواب True يتم تنفيذ الأوامر التي تلي Then حتي الوصول إلى End If
- ٤- وفي حالة ناتج التعبير الشرطي خطأ False يتم تنفيذ الأوامر التي تلي End If

مثال

```
If X >= 50 Then
Msgbox(" ناجح ")
End if
```



مربع الرسالة (MsgBox)

هي اختصار (Message Box) أي مربع الرسالة ، ويستخدم عند الحاجة لعرض رسائل أو إعطاء نصائح
 لمستخدم البرنامج أثناء تشغيل البرنامج ويكتب بالقاعدة التالية (الرسالة ") Msgbox(")

التعبيرات الشرطية Conditional Expressions

تعريف : التعبير الشرطي

هو جزء من كود Code البرمجة يكون ناتجه إما صواب (صحيح) True أو خطأ False وذلك بناء على قيمة خاصية Property أو متغير Variable أو بيان آخر بالبرنامج

للتعبير الشرطي قيمة تتوقف على صحة العلاقة أو عدم صحتها .

✗ فإذا كانت العلاقة غير صحيحة فتكون قيمة التعبير الشرطي False .

✗ والعكس إذا كانت العلاقة صحيحة فتكون قيمة التعبير الشرطي True

Degree >= 50

مثال : التعبير الشرطي

□ عبارة عن صياغة للعلاقة بين المتغير Degree و الرقم 50

□ العلاقة بينهما هي ان المتغير Degree أكبر من او تساوى 50

□ فإذا كانت العلاقة صحيحة فان قيمة التعبير الشرطي تكون صحيحة (True)

□ أما إذا كانت العلاقة غير صحيحة فان قيمة التعبير الشرطي تكون خاطئة (False)

لصياغة التعبيرات الشرطية فإننا نقوم باستعمال بعض المعاملات تسمى معاملات المقارنة

A = 50 B = 60

مثال ::

النتيجة	مثال	المعنى	المعامل
False	$A > B$	أكبر من	$>$
True	$A < B$	اصغر من	$<$
False	$A = B$	يساوي	$=$
True	$A < > B$	لا يساوي	$< >$
False	$A >= B$	أكبر من او تساوى	$>=$
True	$A <= B$	اصغر من او تساوى	$<=$

مثال آخر

التعبير الشرطي	الناتج	الشرح
$100 < > 100$	FALSE	حيث أن القيمتين متساويتين
$300 < > 100$	TRUE	حيث أن القيمتين غير متساويتين
DEGREE=Textbox1.Text	TRUE	إذا كانت قيمة الخاصية Text للأداة Textbox1 تساوى محتوى المتغير DEGREE
	FALSE	إذا كانت قيمة الخاصية Text للأداة Textbox1 لا تساوى محتوى المتغير DEGREE

تدريب: المطلوب عمل برنامج لإدخال الدرجة وفي حالة ما تكون الدرجة اكبر من أو تساوى 50 يعطى صندوق رسالة به كلمة ناجح

١- قم بإنشاء مشروع يحتوى على نموذج Form كما هو موضح و ضع الأدوات المناسبة

المطلوب

- ١- عند الضغط على زر "نتيجة" يعطى صندوق رسالة MessageBox به كلمة ناجح إذا كانت الدرجة أكبر من أو تساوى 50
- ٢- وفي حالة إدخال درجة أقل من 50 لا يظهر أي شيء .
- ٣- سوف نكتب الكود Code بالاعتماد على خريطة التدفق كما هو مبين

الكود	خريطة التدفق
<p>الكود Code يكتب في اجراء الحدث Click الخاص بزر الأمر Button</p> <pre>Private Sub Button1_click Dim X As Single X = Me.Textbox1.Text IF X >= 50 Then MsgBox ("ناجح") End If End Sub</pre>	<pre>graph TD Start([Start]) --> EnterX[/Enter X/] EnterX --> Decision{X >= 50} Decision -- YES --> Print[/Print "ناجح"/] Print --> End([End]) Decision -- NO --> End</pre>

لاحظ:

- ١- تم استخدام جملة IF والتي تعبر عن الشرط هل قيمة X أكبر من أو تساوى 50
- ٢- تم كتابة كود إظهار مربع الرسالة ("ناجح") MsgBox في السطر التالي لكلمة Then
- و تم انهاء الكود بكلمة End If

٣- يمكن كتابة كود Code البرنامج بالاعتماد على خطوات الحل Algorithm

- ١- بداية
- ٢- ادخل X
- ٣- إذا كان $X \geq 50$ اذن
- ١-٣ طباعة "ناجح"
- ٤- نهاية

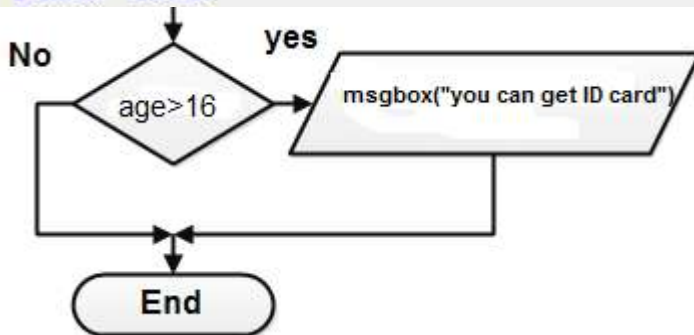
□ قم بتشغيل البرنامج.

□ أدخل قيم مثل 20 و 50 و 75 والنقر على زر "نتيجة" في كل مرة.

و لاحظ ما سيحدث ؟

- لاحظ :** ١- عدم ظهور صندوق الرسالة MessageBox عند إدخال القيمة ٢٠ أى أقل من 50 طبقا للشرط فيكون ناتج الشرط False فينفذ ما بعد End if وهو End Sub أي إنهاء الإجراء Procedures
- ٢- أما عند إدخال القيم ٥٠ و ٧٠ فيتم عرض صندوق رسالة ناجح حيث تحقق الشرط $X \geq 50$ و قيمته True خلى بالك : - يمكن كتابة نفس جملة If السابقة في سطر واحد ولا يتم وضع End If

```
Private Sub Button1_Click(ByVal send
    Dim x As Single
    x = Me.TextBox1.Text
    If x >= 50 Then MsgBox("ناجح")
End Sub
```



مثال آخر

```
If Age > 16 Then
MsgBox("You can get an ID card")
End If
```

هام :: جملة If Then

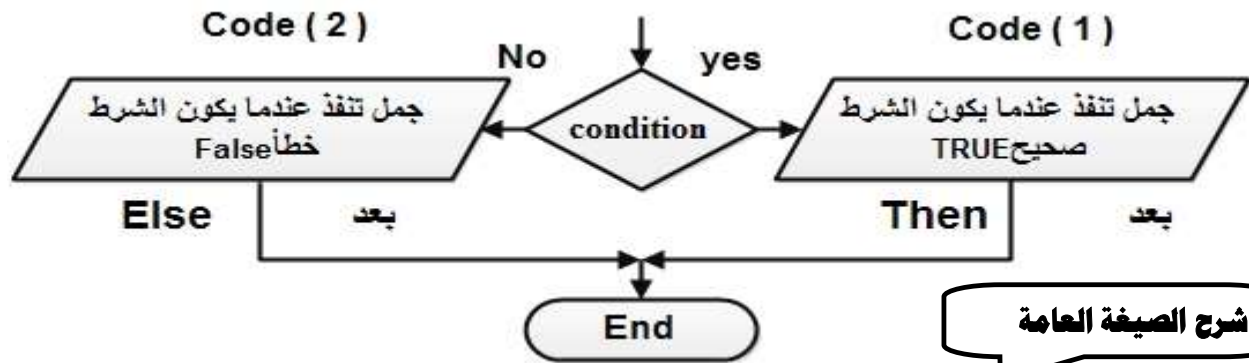
يتم استخدامها في حاله ما إذا كان هناك بديل واحد فقط (أ) بمعنى انه عند قيمة معينة للشرط يتم تنفيذ البديل (أ) أما عند القيمة الأخرى فلا يتم عمل شئ و يذهب الى النهاية

التفرع باستخدام جملة If...Then..Else

تختلف جملة If...Then..Else عن السابقة حيث يتم استخدام هذه الجملة في حالة ما إذا كان هناك سؤال او شرط معين له بديلين (أ ، ب) وكنا نريد أن يتم تنفيذ احد البديلين إذا كانت قيمة الشرط true او yes ويتم تنفيذ البديل الآخر إذا كانت قيمة الشرط false او No

الصيغة العامة لهذه الجملة:

```
If conditional Expression ( شرط او تعبير شرطي ) Then
    Code(1) مجموعة الاوامر التي يتم تنفيذها عندما يكون ناتج الشرط ( True ) صواب
Else
    Code(2) مجموعة الاوامر التي يتم تنفيذها عندما يكون ناتج الشرط ( False ) خطأ
End if
تكملة البرنامج
```

- ١- يبدأ التفرع باستخدام جملة If...Then..Else بكلمة If وما بعدها هو الشرط أو التعبير الشرطي Conditional Expression ثم كلمة Then على نفس السطر و تقابل كلمة اذن في خطوات الحل او Yes في خرائط التدفق
- ٢- في حالة ما يكون ناتج التعبير الشرطي صواب True يتم تنفيذ الأوامر التي تلي Then أي Code(1) ويهمل مجموعة الأوامر التي تلي Else أي Code(2) ثم يذهب لتنفيذ مجموعة الأوامر التي تلي End If
- ٣- كلمة Else على سطر بمفردها ومعناها (و"لا" و تقابل كلمة غير ذلك في خطوات الحل او No في خرائط التدفق)
- ٤- وفي حالة ناتج التعبير الشرطي خطأ False يتم تنفيذ الأوامر التي تلي Else أي Code(2) ويهمل مجموعة الأوامر التي تلي Then أي Code(1) ثم يذهب لتنفيذ مجموعة الأوامر التي تلي End If
- ٥- ينتهي التعبير الشرطي بكلمة End if

مثال

تعديل الكود Code السابق ليعطي صندوق رسالة MessageBox به كلمة راسب في حالة الدرجة أقل من 50

المطلوب

- ١- عند الضغط على زر "نتيجة" يعطى صندوق رسالة MessageBox به كلمة ناجح إذا كانت الدرجة أكبر من أو تساوى 50
- ٢- ويعطى صندوق رسالة MessageBox به كلمة راسب في حالة الدرجة أقل من 50
- ٣- سوف نكتب الكود Code بالاعتماد على خريطة التدفق كما هو مبين

الكود	خريطة التدفق
<p>الكود Code يكتب في اجراء الحدث Click الخاص بزر الأمر Button</p> <pre> Private Sub Button1_click Dim X As Single X =Me. Textbox1.Text IF X >=50 Then MsgBox ("ناجح") Else Msgbox("راسب") End If End Sub </pre>	

لاحظ

١- أننا استخدمنا Else والتي تعني تنفيذ ما يليها في حالة ناتج الشرط False
خلى بالك : يمكن كتابة نفس جملة If السابقة في سطر واحد ولا يتم وضع End If

```
Dim x As Single
x = Me.TextBox1.Text
If x >= 50 Then MsgBox("ناجح") Else MsgBox("راسب")
```

مثال

بنفس الأسلوب السابق قم بتنفيذ مشروع Project يحتوي على نموذج Form يستقبل قيمة في المتغير N من خلال صندوق نص TextBox ثم اظهر عبارة "الرقم زوجي" أو "الرقم فردي"

المطلوب

- ١- عند الضغط على زر "نوع الرقم" يعطى صندوق رسالة MessageBox به كلمة الرقم زوجي إذا كانت قيمة N أي العدد المدخل يقبل القسمة على 2 بدون باقي (الباقى = صفر)
- ٢- ويعطى صندوق رسالة MessageBox به كلمة الرقم فردي إذا كانت قيمة N أي العدد المدخل يقبل القسمة على 2 مع وجود باقي (الباقى = واحد)

الكود	خريطة التدفق
<p>الكود Code يكتب في اجراء الحدث Click الخاص بجزر الأمر Button</p> <pre>Private Sub Button1_click Dim N As Long N = Me. Textbox1.Text IF N Mod 2=0 Then MsgBox ("الرقم زوجي") Else Msgbox("الرقم فردي") End If End Sub</pre>	<pre> graph TD Start([start]) --> GetN[/Get N/] GetN --> Decision{N mod 2=0} Decision -- Yes --> PrintYes[/Print "الرقم زوجي"/] Decision -- No --> PrintNo[/Print "الرقم فردي"/] PrintYes --> End([End]) PrintNo --> End </pre>

لاحظ

- ١- أننا اختبرنا قيمة N إذا كانت تقبل القسمة على 2 بدون باقي من خلال التعبير الشرطي $N \text{ Mod } 2$
- ٢- المعامل Mod هو معامل حسابي يعطى باقي القسمة وفي حالة ان يكون باقي القسمة على 2 يساوى صفر فهذا يعنى عدم وجود باقي قسمة وبالتالي يكون العدد زوجي

التفرع باستخدام جملة If...Then..ElseIf

لاحظ

- ١- جملة If...Then...ElseIf تستخدم عند وجود اختيارين او بديلين أحدهما في حالة الصواب True و الآخر في حالة الخطأ False
- ٢- أما جملة If...Then...Elseif فتستخدم عند وجود أكثر من تعبير شرطي

الصيغة العامة لهذه الجملة:

If conditional Expression (التعبير الشرطي الاول) Then

Code(1) مجموعة الاوامر التي يتم تنفيذها عند تحقق الشرط الاول (True) صواب

ElseIf conditional Expression (التعبير الشرطي الثانى) Then

Code(2) مجموعة الاوامر التي يتم تنفيذها عند تحقق الشرط الثانى (True) صواب

ElseIf conditional Expression (التعبير الشرطي الثالث) Then

Code(3) مجموعة الاوامر التي يتم تنفيذها عند تحقق الشرط الثالث (True) صواب

Else

Code(4) مجموعة الاوامر التي يتم تنفيذها عند عدم تحقق كل الشروط السابقة

End if

تكملة البرنامج

شرح الصيغة العامة

- ١- يبدأ التفرع باستخدام كلمة If و ما بعدها هو الشرط أو التعبير الشرطي الاول Conditional Expression ثم كلمة Then على نفس السطر ثم يقوم باختبار الشرط الاول فإذا كان الشرط صحيحا ينفذ مجموعة الاوامر الخاصة بالكود الاول Code(1) اما اذا كان الشرط غير صحيح يذهب الى الشرط الثانى بعد كلمة ElseIf
- ٢- يقوم البرنامج باختبار الشرط الثانى فإذا كان الشرط صحيحا ينفذ مجموعة الاوامر الخاصة بالكود الثانى Code(2) اما اذا كان الشرط غير صحيح يذهب الى الشرط الثالث بعد كلمة ElseIf
- ٣- يقوم البرنامج باختبار الشرط الثالث فإذا كان الشرط صحيحا ينفذ مجموعة الاوامر الخاصة بالكود الثالث Code(3) اما اذا كان الشرط غير صحيح يذهب الى الشرط الرابع اى الكود الرابع Code(4) بعد كلمة Else فقط
- ٤- ينتهي التعبير الشرطي بكلمة End if

مثال

تم بعمل مشروع يحتوى على نموذج Form و أداة مربع النص Textbox1 وأداتين عنوان Label و زر Button و النص المعروض عليه كلمة اختبار

المطلوب

- ١- إدخال درجة الحرارة (D) او Degree من خلال ثم النقر على الزر "اختبار" يعرض في Label2 عبارة " فوق الصفر " أو عبارة " تساوى صفر " أو عبارة " تحت الصفر " .
- ٢- سوف نكتب الكود Code بالاعتماد على خريطة التدفق كما هو مبين

الكود	خريطة التدفق
<p>الكود Code يكتب في اجراء الحدث Click الخاص بزر الأمر Button</p> <pre> Private Sub Button1_click Dim Degree as Single Degree = Me. Textbox1.Text IF Degree = 0 Then Label2.text = " درجة الحرارة تساوي الصفر " Else If Degree < 0 Then Label2.text = " درجة الحرارة تحت الصفر " Else Label2.text = " درجة الحرارة فوق الصفر " End If End Sub </pre>	<pre> graph TD Start([Start]) --> GetD[/Get D/] GetD --> D0{D=0} D0 -- True --> Print0[/Print "تساوى صفر"/] D0 -- False --> Dlt0{D<0} Dlt0 -- True --> PrintBelow[/Print "تحت صفر"/] Dlt0 -- False --> PrintAbove[/Print "فوق صفر"/] Print0 --> End([End]) PrintBelow --> End PrintAbove --> End </pre>

لاحظ:

- ١- تم الإعلان عن المتغير Degree من النوع single لاحتمال إدخال رقم عشري
- ٢- أن جملة If بدأت بالتعبير الشرطي Degree=0 فإذا كان ناتج الشرط True يطبع درجة الحرارة تساوي الصفر
- ٣- وفي حالة ناتجه يساوى False يتم اختبار تعبير شرطي آخر وهو Degree<0
- ٤- وفي حالة ناتجه يساوى False يطبع درجة الحرارة فوق الصفر يبقى احتمال آخر وهو Degree>0 لذلك لم يكتب صراحة إنما تم التعبير عنه ب Else لأنه هو الاحتمال الوحيد الباقي وهو طباعة درجة الحرارة تحت الصفر

قم بتشغيل البرنامج بالضغط على F5 لعمل Start debugging

١- ادخل أرقام مثل 30 و 0 و 10 مع النقر على الزر كل مرة ماذا تلاحظ؟

٢- انقر الزر مع بقاء مربع النص TextBox خالياً أو محتوياً نص؟ ماذا تلاحظ؟

ج : سوف نلاحظ توقف البرنامج وظهور خطأ من النوع Runtime

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) H
    Dim degree As Single
    degree = Me.TextBox1.Text
    If degree = 0 Then
        Label2.Text = "ساوي صفر"
    ElseIf degree < 0 Then
        Label2.Text = "من الصفر"
    Else
        Label2.Text = "من الصفر"
    End If
End
```

سطر الكود الذي حدث به الخطأ

InvalidCastException was unhandled

رسالة تبين نوع الخطأ

Troubleshooting tips:

When casting from a number, the value must be a number less than infinity.

Make sure the source type is convertible to the destination type.

لاحظ:

١- أن الخطأ حدث عند السطر Degree = Me.Textbox1.Text

٢- أن سبب الخطأ هو أن صندوق النص TextBox لا يحتوى على رقم لكي يتم تحويله ضمناً إلى النوع Single كما حدث من قبل عند ادخال القيم ٣٠ و ٠ و ١٠ وهو نوع بيانات المتغير Degree وإنما حدثت مشكلة نتيجة عدم إمكانية التحويل من النوع حرفي (حيث تم ترك مربع النص TextBox خالياً بدون قيم) إلى النوع Single

طرق منع حدوث الأخطاء عند التشغيل Runtime error

الطريقة الأولى

إعطاء رسالة للمستخدم عن طريق استخدام جملة If إضافية مع دالة Isnumeric كما يلي

جملة If الإضافية

الكود Code الأصلي
ينفذ في حالة إدخال
قيم عددية

الكود Code الذي ينفذ
في حالة إدخال قيم غير
عددية

```
Dim Degree as Single
If Isnumeric (Me. Textbox1.Text) Then
    Degree =Me. Textbox1.Text
    IF Degree =0 Then
        Label2.text =" درجة الحرارة تساوي الصفر "
    Else If Degree < 0 Then
        Label2.text =" درجة الحرارة تحت الصفر "
    Else
        Label2.text =" درجة الحرارة فوق الصفر "
    End If
Else
    MsgBox( " ادخل عدد " )
    Me. Textbox1. Focus ( )
    Me. Textbox1.Text = ""
End If
End Sub
```


لاحظ:

وجود جملة If الإضافية لاختبار قيمة الخاصية Text لصندوق النص TextBox هل هي قيمة عددية أم لا

١- وفي حالة الناتج True يتم تنفيذ الكود المطلوب

٢- أما في حالة الناتج False أي أن القيمة غير عددية يتم تنفيذ الكود Code البديل الذي يلي Else

٣- استخدام التعبير الشرطي $\text{Isnumeric}(\text{Textbox1.Text}) = 0$ في جملة IF الإضافية لمعرفة هل هو عددي أم حرفيما المقصود بالدالة Isnumeric

١- هي أحد الدوال المعرفة Predefined Functions التي تستخدم لمعرفة محتوى متغير Variable أو محتوى

صندوق نص TextBox هل هو عددي أم حرفي

٢- الدالة Isnumeric لها مدخل أو وسيط أو معامل Argument بين القوسين

٣- الدالة Isnumeric ولها ناتج منطقي يكون True أو False

أمثلة على الدالة Isnumeric

الامر	الناتج	السبب
MsgBox IsNumeric("youssif")	False	حيث أن القيمة داخل القوسين غير عددية
MsgBox(IsNumeric(10))	True	حيث أن القيمة داخل القوسين عددية
MsgBox IsNumeric(a)	True	إذا كانت قيمة المتغير a عددية
	False	إذا كانت قيمة المتغير a غير عددية
MsgBox IsNumeric(textbox3.text)	True	إذا كان صندوق النص TextBox3 يحتوى على قيمة عددية
	False	إذا كان صندوق النص TextBox3 يحتوى على قيمة غير عددية

الطريقة الثانية

يمكن استخدام الجملة Try..catch كما يلي

بدا جملة Try

الكود Code الأصلي

ينفذ في حالة إدخال قيم عددية

الكود Code الذي ينفذ

في حالة إدخال قيم غير عددية

Dim Degree as Single

Try

Degree = Me. Textbox1.Text

IF Degree = 0 Then

Label2.text = "درجة الحرارة تساوي الصفر"

Else If Degree < 0 Then

Label2.text = "درجة الحرارة تحت الصفر"

Else

Label2.text = "درجة الحرارة فوق الصفر"

End If

Catch ex As Exception

Msgbox(" ادخل عدد ")

Me. Textbox1. Focus ()

Me. Textbox1.Text = ""

End Try

لاحظ:

الهدف من الكود Code المكتوب عمل التالي عند حدوث الخطأ

١- عرض صندوق رسالة MessageBox يطالب المستخدم بإدخال عدد بدلا من قيمة حرفية أو قيمة خالية

٢- الامر Focus

هو (وسيلة وليس خاصية) معناه حيازة الانتباه أي أن الأداة قيد الاستخدام أي هي المستخدمة الآن
و هو خاص بصندوق النص TextBox أي تقبل لغة VB.Net بنقل التركيز اليه ووضع المؤشر بداخل صندوق النص

٣- الخاصية Text الخاصة بالأداة TextBox إذا أخذت القيمة " "

يعنى جعل خاصية Text تساوى سلسلة حرفية خالية empty string وهذا يلغى المدخلات السابقة للأداة TextBox

استخدام المعاملات المنطقية Logical Operators مع الجملة الشرطية If

١. كنا نتخذ القرار فى جملة If...Then...Else بناء على شرط واحد او تعبير شرطى واحد
٢. لكن قد نحتاج إلى اختبار أكثر من تعبير شرطى معا في نفس جملة If وفى الحالة نربط بين التعبيرات الشرطية باستخدام المعاملات المنطقية Logical Operators

فائدة المعاملات المنطقية: هى ربط واختبار أكثر من تعبير شرطى معا فى نفس جملة IF الواحدة

أمثلة على المعاملات المنطقية Logical Operators

المعامل المنطقي Or

المعامل المنطقي And

لاحظ:

المعامل And يعطى ناتج نهائى True إذا كان ناتج كافة (كل) الشروط True

المعامل or يعطى ناتج نهائى True إذا كان ناتج أحد الشروط True

أمثلة لاستخدام المعامل And بفرض أن المتغير A والمتغير B متغيرات منطقية

جدول AND

إذا كان كل من الشرطين صحيح True تكون قيمة التعبير الشرطي صحيحة True و إذا كان إحداهما او كلاهما خطأ False تكون قيمة التعبير الشرطي خطأ False

السبب	النتيجة	التعبير الشرطي
تم تحقق الشرطين كلاهما معا	True(✓)	A= TRUE(✓) AND B= TRUE(✓)
تم تحقق احد الشرطين فقط	False(✗)	A=TRUE(✓) AND B=FALSE(✗)
تم تحقق احد الشرطين فقط	False(✗)	A= FALSE(✗) AND B= TRUE(✓)
لم يتحقق اى من الشرطين	False(✗)	A= FALSE(✗) AND B= FALSE(✗)

أمثلة لاستخدام المعامل Or بفرض أن المتغير A والمتغير B متغيرات منطقية**جدول OR**

إذا كان كل من الشرطين صحيح True أو إحداهما صحيح True تكون قيمة التعبير الشرطي **صحيحة** وإذا كان كلاهما خطأ False تكون قيمة التعبير الشرطي **خطأ** False

السبب	النتيجة	التعبير الشرطي
تم تحقق الشرطين كلاهما معا	True(✓)	A= TRUE(✓) OR B= TRUE(✓)
تم تحقق الشرط الأول	True(✓)	A=TRUE(✓) OR B=FALSE(✗)
تم تحقق الشرط الثانى	True(✓)	A= FALSE(✗) OR B= TRUE(✓)
لم يتحقق أى من الشرطين	False(✗)	A= FALSE(✗) OR B= FALSE(✗)

مثال

تم بعمل مشروع يحتوى على نموذج Form وأداتين مربع النص Textbox1 لإدخال درجة التحريرى Textbox2 لإدخال درجة العملى وأربع أدوات عنوان Label و زر Button المعروض عليه كلمة **عرض النتيجة**

المطلوب

- عند النقر على الزر "عرض النتيجة" يعرض فى Label4 ناجح أو راسب
- يعرض ناجح في حالة كل من الدرجتان أكبر من أو تساوى 50
- يعرض راسب في حالة ما إذا كانت أي من هاتين الدرجتين أقل من 50

الكود Code يكتب في اجراء الحدث Click الخاص بزر الأمر Button (عرض النتيجة)

```
Private Sub Button1_click
Dim deg1, deg2 as Single
deg1=Me. Textbox1.Text
deg2=Me. Textbox2.Text
```

```
IF deg1 >=50 And deg2 >=50 Then
```

```
Me. Label4.text = "ناجح"
```

```
Else
```

```
Me. Label4.text = "راسب"
```

```
End If
```

```
End Sub
```

درجة التحريرى

درجة العملى

المعامل المنطقى And

مثال

استخدم نفس التدريب السابق مع تغير الشرط ليصبح ناجح في حالة إى درجة من الدرجتين أكبر من أو تساوى 50 و راسب إذا كانت كل من الدرجتان أقل من 50

المطلوب

- ١- عند النقر على الزر " عرض النتيجة " يعرض فى Label4 ناجح أو راسب
- ٢- يعرض ناجح في حالة ما إذا كانت أي من هاتين الدرجتين أكبر من أو تساوى 50
- ٣- يعرض راسب في حالة كل من الدرجتين أقل من 50

الكود Code يكتب في إجراء الحدث Click الخاص بزر الأمر Button (عرض النتيجة)

```
Private Sub Button1_click
Dim deg1, deg2 as Single
deg1=Me. Textbox1.Text
deg2=Me. Textbox2.Text
IF deg1 >=50 Or deg2 >=50 Then
Me. Label4.text = " ناجح "
Else
Me. Label4.text = " راسب "
End If
End Sub
```

درجة التحريرى

درجة العملى

Or المعامل المنطقى

لاحظ

١- استخدام And في التعبير الشرطي (المثال الاول) والذي يعنى وجوب تحقق جميع الشروط حتى يكون الناتج

الكل للتعبير الشرطي True

٢- استخدام Or في التعبير الشرطي (المثال الثانى) والذي يعنى عدم وجوب تحقق جميع الشروط ولكن يكفي

بان يكون احدى الشرطين True حتى يكون الناتج الكل للتعبير الشرطي True

٣- أن ناتج ما بين If و Then لابد أن يكون ناتج منطقي True أو False

٤- لاحظ أن الفارق الوحيد فى المثالين السابقين هو استخدام Or بدلا من And

التفرع باستخدام Select...Case

لاحظ

١- جملة Select...Case تشبه جملة If...Then...Elseif غير أنها فعالة بشكل أكثر

٢- يفضل استخدام Select...Case عندما يكون التفرع معتمدا على قيمة متغير Variable واحد

٣- تستخدم فى حالة وجود بدائل واحتمالات متعددة

٤- كما أنها تجعل الكود Code مفهوم بشكل أكثر

الصيغة العامة لهذه الجملة:

Select Case Variable اسم المتغير

Case value1 الحالة الأولى

Code1 مجموعة الأوامر التي يتم تنفيذها عندما تكون قيمة المتغير مساوية للحالة الأولى

Case value2 الحالة الثانية

Code2 مجموعة الأوامر التي يتم تنفيذها عندما تكون قيمة المتغير مساوية للحالة الثانية

Case value3 الحالة الثالثة

Code3 مجموعة الأوامر التي يتم تنفيذها عندما تكون قيمة المتغير مساوية للحالة الثالثة

Case else الحالة الأخيرة

Code مجموعة الأوامر التي يتم تنفيذها في حالة لم تكن قيمة المتغير مساوية لأي من الحالات السابقة

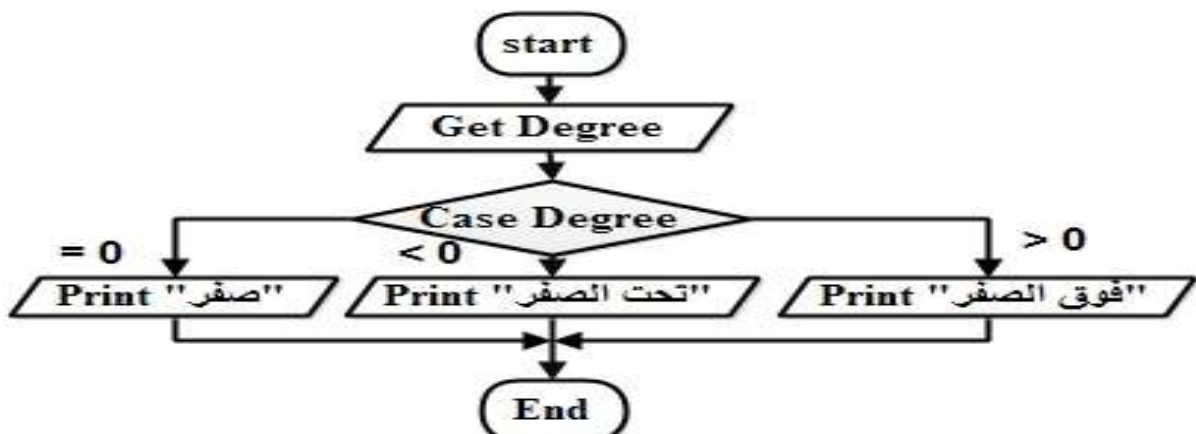
End Select

شرح الصيغة العامة

- ١- يبدأ التفرع باستخدام كلمة Select Case و ما بعدها هو اسم المتغير Variable
- ٢- ثم كلمة Case على السطر التالي وهي تدل على المتغير عائدة على المتغير و يليها الشرط او الحالة الأولى
- ٣- يأتي بعد Case مجموعة الأوامر الأولى (Code1) التي يتم تنفيذها عندما تكون قيمة المتغير مساوية للحالة الأولى
- ٤- وفي حال لم تكن الحالة الأولى مساوية ل قيمة المتغير يتم الانتقال الى الحالة الثانية وتنفيذ مجموعة الأوامر الثانية (Code2) وهكذا
- ٥- ويمكن ان يتم تكرار كلمة Case يليها الشرط اي عدد من المرات او الحالات حسب متطلبات البرنامج
- ٦- تنتهي الحالات بالحالة الأخيرة Case else ويتم تنفيذ الأوامر التالية لها في حالة لم تكن قيمة المتغير مساوية لأي من الحالات السابقة في البرنامج
- ٧- تنتهي جملة Select...Case بكلمة End Select
- ٨- يمكن الاستغناء عن Case else طالما لا حاجة لها

تعديل الكود Code في التدريب السابق باستخدام Select...Case بدلاً من If...Then...Elseif

لو حاولنا رسم خريطة التدفق المتوافقة مع Select ...Case سنجد أنها كما هو موضح



من شكل السابق يتضح إمكانية التفرع إلى أكثر من فرعين من رمز اتخاذ القرار حيث يحتوي رمز اتخاذ القرار على سؤال

عن قيمة المتغير Degree وحسب قيمته نتفرع إلى كودات مختلفة

```
Dim Degree as Single
```

```
Try
```

```
Degree = Me.Textbox1.Text
```

بدا جملة Select Case

```
Select Case Degree
```

```
Case 0
```

```
Me.Label2.text = "الصفحة 0"
```

الكود الأصلي ينفذ في

حالة إدخال قيم عددية

```
Case Is < 0
```

```
Me.Label2.text = "تحت الصفحة 0"
```

```
Case Is > 0
```

```
Me.Label2.text = "فوق الصفحة 0"
```

نهاية جملة Select Case

```
End Select
```

```
Catch ex As Exception
```

```
Msgbox("ادخل عدد")
```

```
Me.Textbox1.Focus()
```

```
Me.Textbox1.Text = ""
```

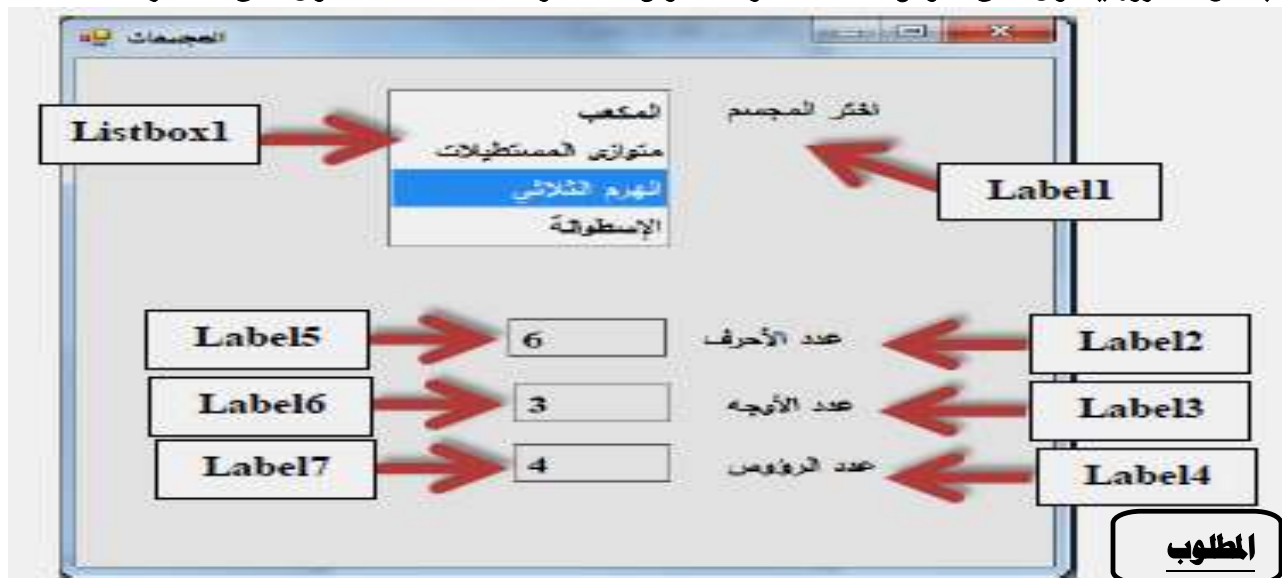
الكود الذي ينفذ في

حالة إدخال قيم غير عددية

```
End Try
```

تدريب قم بإنشاء مشروع جديد New Project يحتوي على نموذج Form

قم بعمل مشروع يحتوي على نموذج Form ٧ أدوات عنوان Label وأداة Listbox تحتوي على عناصر



عند النقر على أي عنصر من عناصر الأداة Listbox1 يعرض عدد الأحرف sides وعدد الأوجه faces وعدد الرؤوس vertices للمجسم المحدد

لاحظ :

يكتب الكود في إجراء الحدث SelectedIndexChanged الخاص بالأداة Listbox1 وعند الانتهاء يتم تشغيل

البرنامج بالضغط على مفتاح F5

ملاحظات هامة

الأداة Listbox

- ١- تعد الأداة Listbox واحدة من الأدوات التي تربط غالبا بال تكرار وتتيح لك هذه الأداة أن تنشئ أو تعرض قائمة من العناصر وان تختار منها
- ٢- تظهر الأداة على هيئة مستطيل يعرض بداخله صفوفًا من الكلام
- ٣- كل صف من هذه الصفوف يعد عنصرا يمكن لمستخدم البرنامج أن يختاره
- ٤- Index يقصد به الفهرس أى هو رقم العنصر في القائمة (ترتيبه) ويأخذ الأرقام من صفر إلى 1 - count حيث count هو عدد العناصر في الأداة listbox
- ٥- يتم إضافة عناصر Items للأداة Listbox عن طريق الخاصية Items في نمط التصميم
- ٦- يبدأ الترقيم داخل الأداة listbox من الصفر { أى ان العنصر الأول يأخذ الرقم (صفر) والثانى رقم (واحد) والثالث رقم (اثنين) وهكذا ... } ولو ان عدد عناصر القائمة ٤ فان العنصر الرابع ترتيبه فى القائمة هو ٣
- ٧- يكتب الكود Code في إجراء الحدث SelectedIndexChanged الذى يعنى عند تغير فهرس Index أي ترتيب العنصر المحدد وهو الحدث الافتراضي للأداة Listbox1 ويتحقق هذا الحدث عند اختيار عنصر من عناصر الأداة Listbox1
- ٨- لذلك يتم معرفة العنصر المحدد (رقم العنصر في القائمة) عن طريق الخاصية Selected Index للأداة Listbox1
- ٩- عندما تكون قيمة هذه الخاصية 1 مثلا فهذا يعنى تحديد العنصر الثانى وهكذا

```
Private Sub Listbox1_SelectedIndexChanged
```

```
Dim X as Byte
```

```
X = Listbox1.SelectedIndex
```

تخصيص قيمة فهرس العنصر المحدد فى المتغير X

```
Select Case X
```

```
Case 0
```

```
Label5.Text = "12"
```

```
Label6.Text = " 6 "
```

```
Label7.Text = " 8 "
```

الكود فى حالة اختيار العنصر الذى ترتيبه صفر (المكعب)

```
Case 1
```

```
Label5.Text = "12"
```

```
Label6.Text = " 6 "
```

```
Label7.Text = " 8 "
```

الكود فى حالة اختيار العنصر الذى ترتيبه (١) (متوازي المستطيلات)

```
Case 2
```

```
Label5.Text = " 6 "
```

```
Label6.Text = " 3 "
```

```
Label7.Text = " 4 "
```

الكود فى حالة اختيار العنصر الذى ترتيبه (٢) (الهرم الثلاثى)

```
Case 3
```

```
Label5.Text = " لا يوجد "
```

```
Label6.Text = " لا يوجد "
```

```
Label7.Text = " لا يوجد "
```

الكود فى حالة اختيار العنصر الذى ترتيبه (٣) (الاسطوانة)

```
End Select
```

```
End Sub
```

تدريبات عملية

من د ا رستك مادة العلوم تعلمت أن لكل عنصر رمز وعدد ذري وعدد كتلي , قم بإنشاء مشروع Project يحتوي على

نموذج Form كما هو موضح بشكل

المطلوب

عند اختيار أحد العناصر ثم النقر على الزر "عرض" يظهر الرمز والعدد الذري والعدد الكتلي لهذا العنصر.
نكتب الكود Code في إجراء الحدث Click الخاص بالزر "عرض" كما يلي

```
Private Sub Button1_Click
```

```
If RadioButton1.Checked = True Then
```

```
Label4.Text = "O"
```

```
Label5.Text = "8"
```

```
Label6.Text = "16"
```

```
ElseIf RadioButton2.Checked = True Then
```

```
Label4.Text = "Na"
```

```
Label5.Text = "11"
```

```
Label6.Text = "23"
```

```
ElseIf RadioButton3.Checked = True Then
```

```
Label4.Text = "Mg"
```

```
Label5.Text = "12"
```

```
Label6.Text = "24"
```

```
ElseIf RadioButton4.Checked = True Then
```

```
Label4.Text = "Al"
```

```
Label5.Text = "13"
```

```
Label6.Text = "27"
```

```
Else
```

```
MsgBox("يرجى اختيار أحد العناصر")
```

```
End If
```

```
End Sub
```

عرض الرمز و العدد الذري و الكتلي للأكسجين

عرض الرمز و العدد الذري و الكتلي للصوديوم

عرض الرمز و العدد الذري و الكتلي لماغنسيوم

عرض الرمز و العدد الذري و الكتلي للومنيوم

عرض مربع رسالة في حالة عدم اختيار أي عنصر

من دارستك للجغرافيا تعلمت الكثير من المفاهيم مثل المجرة والكوكب والنجم ... الخ قم بإنشاء مشروع Project يحتوي

على نموذج Form

المطلوب

عند اختيار أحد عناصر الأداة
ComboBox يعرض في
مربع النص TextBox تعريفه

Private Sub ComboBox1_SelectedIndexChanged

Select Case ComboBox1.SelectedIndex

Case 0

TextBox1.Text = "تجمعات كبيرة من النجوم المختلفة الأحجام والأشكال والأنواع"

Case 1

TextBox1.Text = "جسم معتم لا يشع ضوءاً ولا حرارة ويعتمد ضوءه وحرارته من نجم قريب منه"

Case 2

TextBox1.Text = "جسم معتم أصغر حجماً من الكوكب يرتبط به بفعل الجاذبية ويعكس ضوء الشمس الواقع عليه"

Case 3

TextBox1.Text = "جسم مضئ ملتهب يشع ضوءاً وحرارة"

Case 4

TextBox1.Text = "جسم صلب يحترق جزئياً عند احتكاكه بالغلاف الجوي"

Case 5

TextBox1.Text = "جسم سماوي صلب يسبح في الفضاء حول الشمس ويجرى في السماء على هيئة سهام ضوئية"

Case Else

MsgBox ("يرجى اختيار أحد المفاهيم")

End Select

End Sub

إرشادات

- ١- اجعل الخاصية Multiline لمربع النص TextBox تساوي true حتى تكون متعددة الأسطر
 - ٢- أضف العناصر مثل مجرة ، كوكب ، قمر ، نجم ، نيزك ، شهاب للأداة ComboBox من خلال الخاصية items
 - ٣- يكتب الكود في إجراء الحدث SelectedIndexChanged الخاص بالأداة ComboBox
 - ٤- لقد اعتمدنا في كتابة الكود Code على ترتيب العناصر داخل الأداة ComboBox حيث أن العنصر الأول له ترتيب في index صفر والعنصر الثاني ترتيب في index (١) والعنصر الثالث ترتيب في index (٢) وهكذا
- هام : إجراء الحدث SelectedIndexChanged خاص بالأداة ComboBox والأداة listbox

أسئلة

السؤال الأول : ضع علامة (✓) أمام العبارة الصحيحة وعلامة (x) أمام العبارة الخطأ:

()	(١) عدد اختيارات التفرع الممكنة مع استخدام جملة (If..Then..else) هو ٢
()	(٢) ينفذ الكود الذي يلي (Else) في جملة (IF) ، عندما يكون ناتج التعبير الشرطي (True).
()	(٣) المعامل & هو أحد معاملات المقارنة المنطقية.
()	(٤) باستخدام معاملي المقارنة (Or) يجب أن يكون كافة نواتج الشروط المستخدمة (False) لكي يكون الناتج النهائي (False).
()	(٥) تستخدم جملة (Select Case) في حالة اختبار أكثر من تعبير شرطي بجملة شرطية واحدة.
()	(٦) إذا كانت قيمة المتغير (x) تساوي 15 والمتغير (Y) تساوي 350 فإن ناتج التعبير الشرطي (x>10 And y<500) هو (False).
()	(٧) استخدام (Else) اختياري في جملة (If).
()	(٨) المعامل (and) يتطلب ان يكون الشرطان ناتجهما (true) لكي يكون الناتج النهائي (True).
()	(٩) الخاصية التي تشير الى العنصر المحدد بالأداة (ListBox) هي (SelectedIndex).
()	(١٠) تستخدم (Select..Case) في حالة وجود أكثر من احتمالين للتفرع.

السؤال الثاني:

يستخدم الكود (Code) التالي لحساب مساحة دائرة نصف قطرها (R) قم بإعادة كتابة الكود في العمود الأيسر بحيث يعطى رسالة "غير مسموح" عند ادخال عدد سالب في مربع النص (Textbox1) الذي خصص قيمته للمتغير (R).

```
Dim r, area As Single
Const pi As Single = 22 / 7

r = Me.TextBox1.Text

area = 2 * pi * r ^ 2

MsgBox("area of circle= " & area)
```


السؤال الثالث: قم بدراسة الكود (Code) التالي دراسة جيدة والذي يحدد التقدير المناسب لكل درجة مدخلة في مربع النص (Textbox1)

```
Dim deg As Single
deg = Me.TextBox1.Text
```

```
Select Case deg
```

```
Case 85 To 100
    Label12.Text = "ممتاز"
```

```
Case 75 To 84
    Label12.Text = "جيد جداً"
```

```
Case 65 To 74
    Label12.Text = "جيد"
```

```
Case 50 To 64
    Label12.Text = "مقبول"
```

```
Case 0 To 50
    Label12.Text = "ضعيف"
```

```
Case Else
    Label12.Text = "يرجى ادخال درجة بين صفر و 100"
```

```
End Select
```

يكافئ التعبير

≥ 85 and ≤ 100

في هذه الحالة قيمة deg تكون محصورة ما بين ٨٥ الى ١٠٠

حدد ما يتم عرضه في أداة العنوان (Label1) عند ادخال الدرجات التالية في مربع النص (Textbox1):

الدرجة المدخلة	الناتج
30	
84	
64.4	
-33	
صفر	
114.3	

السؤال الرابع: يستخدم الكود التالي لحل معادلة من الدرجة الأولى ($y=3x+2$) ويعطى نتائج صحيحة عند ادخال قيم عددية في مربع النص (Textbox) ولكن حدث خطأ أثناء التشغيل (Runtime Error) عند ادخال قيم حرفية في مربع النص (Textbox) قم بإعادة كتابة الكود (Code) بعد حل هذه المشكلة باستخدام جملة (If) إضافية بحيث يعطى رسالة في أداة العنوان وهي "يجب إدخال قيمة عددية"

```
Dim x, y As Single
x = Me.TextBox1.Text
y = 3 * x + 2
Label11.Text = y
```

السؤال الخامس: اختر ناتج تنفيذ كل من الأكواد التالية:

م	الكود	الناتج
١	<pre>Dim Num As Integer = 10 If Num >= 10 Then Label1.Text = "One" ElseIf Num = 5 Then Label1.Text = "Two" Else Label1.Text = "Three" End If</pre>	<p>أ. One</p> <p>ب. Two</p> <p>ج. three</p>
٢	<pre>Dim A As Integer = 15 If A Mod 3 = 0 Then Label1.Text = "True" Else Label1.Text = "False" End If</pre>	<p>أ. True</p> <p>ب. False</p> <p>ج. 15</p>
٣	<pre>Dim strName As String = "Ahmed" strName = " " & strName strName = "Youssif" & strName Label1.Text = strName</pre>	<p>أ. Youssif Ahmed</p> <p>ب. Ahmed Youssif</p> <p>ج. Ahmed</p>
٤	<pre>Dim A As Integer = 7 Dim B As Integer = 9 MsgBox(A & "X" & B & "=" & " " & A * B)</pre>	<p>أ. AXB=63</p> <p>ب. 7X9=63</p> <p>ج. 7X9=A*B</p>
٥	<pre>Dim x As Integer = 2, y As Integer = 3, z As Integer = 4 If x = y And x <> z Then Label1.Text = "First" ElseIf x = y Or x <> z Then Label1.Text = "Second" ElseIf x <> y And x <> z Then Label1.Text = "Third" Else Label1.Text = "Forth" End If</pre>	<p>أ. First</p> <p>ب. Second</p> <p>ج. Forth</p>

** الفصل الثالث ** Loops and Timers الحلقات التكرارية والمؤقتات

مقدمة

درسنا في الفصل السابق كيفية تنفيذ كود Code محدد بناءً على ناتج تعبير شرطي

تعريف الحلقات التكرارية Loops : هي تكرار كود (Code) محدد أو معين لعدد من المرات

أمثلة على الحلقات التكرارية :

٢- جملة Do While ...Loop

١- جملة For...Next

لاحظ : يمكن تنفيذ كود Code محدد كل فترة زمنية محددة باستخدام الأداة Timer

استخدام جملة For...Next

الصيغة العامة لهذه الجملة:

For Variable = Start To End
Code
Next

بداية التكرار

نهاية التكرار

For العداد = البداية To النهاية
العمل المراد تكرارها
Next

بالعربي

حيث

□ تبدأ الحلقة التكرارية بكلمة For (**بداية التكرار**)

□ وتنتهي الحلقة التكرارية بكلمة Next (**نهاية التكرار**)

□ يتم تحديد اسم المتغير Variable بعد كلمة For ويطلق عليه متغير العداد Counter

□ العداد Counter هو متغير رقمي يستعمله المبرمج لحساب عدد مرات التكرار المحددة مسبقاً

□ المتغير العداد Counter له قيمة بداية Start وقيمة نهاية End ويتم تكرار الكود Code داخل الحلقة حتى

الوصول إلى قيمة النهاية

□ الكلمات (End- Start) هي قيم رقمية يتولى المبرمج تحديدها

شرح الصيغة العامة

القيمة الابتدائية للعداد
For counter = 5 To 50
العداد القيمة النهائية للعداد

١. تخزين القيمة start داخل المتغير counter

٢. مقارنة القيمة المخزنة في counter بالقيمة end

١- إذا كانت القيمة المخزنة في المتغير (العداد Counter) أكبر من القيمة النهائية (End) للتكرار ينتهي

التكرار ويتم تنفيذ الأوامر الموجودة بعد next

ب- إذا كانت القيمة المخزنة في المتغير (العداد Counter) أقل أو تساوي من القيمة النهائية (End) للتكرار يتم

تنفيذ الأوامر المراد تكرارها بين (for , next) من أعلى إلى أسفل

٣. عندما يصل البرنامج إلى next

أ) نيتم زيادة القيمة المخزنة في العداد counter بمقدار واحد (القيمة الافتراضية)

ب) يعود البرنامج إلى بداية التكرار

مثال

قم بإنشاء مشروع Project يحتوي على نموذج Form
و زر Button و النص المعروض عليه كلمة
عرض الأعداد من ١ الى ٣

المطلوب

١- عند الضغط على زر "عرض الأعداد من ١ الى ٣" يعطى صندوق رسالة MessageBox يعرض
بداخله الأعداد من ١ الى ٣ - سوف نكتب الكود Code بالاعتماد على خريطة التدفق كما هو مبين

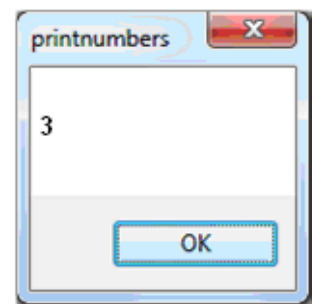
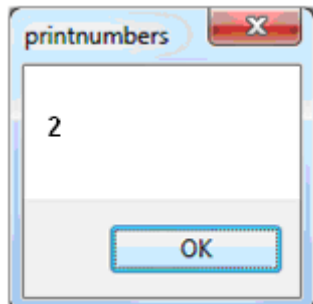
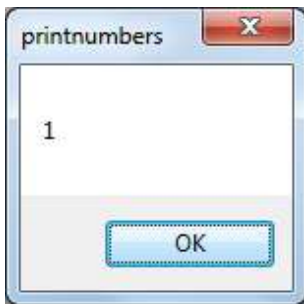
الكود	خريطة التدفق
<p>الكود Code يكتب في اجراء الحدث Click الخاص بزر الأمر Button</p> <pre> Private Sub Button1_click Dim M As Integer For M = 1 To 3 MsgBox (M) Next End Sub </pre>	<pre> graph TD Start([start]) --> M1[M=1] M1 --> Decision{M<=3} Decision -- true --> MsgBox[Msgbox M] MsgBox --> Mplus[M=M+1] Mplus --> Decision Decision -- false --> End([End]) </pre>

ويمكن تتبع سير الكود Code كما هو موضح بالجدول

الامر	قيمة المتغير M	النتائج
Dim M As Integer	0	-
For M=1 to 3	1	قيمة البداية
Msgbox M	1	١
Next	زيادة M بواحد M=1+1	-
Next	هل قيمة M أكبر من قيمة النهاية (النتيجة False)	-
Msgbox M	2	٢
Next	زيادة M بواحد M=2+1	-
Next	هل قيمة M أكبر من قيمة النهاية (النتيجة False)	-
Msgbox M	3	٣
Next	زيادة M بواحد M=3+1	-
Next	هل قيمة M أكبر من قيمة النهاية (النتيجة True)	-

لاحظ

- ١- عندما يصل البرنامج إلى next يقوم بزيادة قيمة المتغير M بمقدار واحد
- ٢- يتم مقارنة القيمة الجديدة بقيمة النهاية للحلقة التكرارية وفي حالة زيادتها عن قيمة النهاية يتم الخروج من الحلقة بالقيمة (٤)
- ٣- عند تشغيل البرنامج ثم النقر على زر الأمر " عرض الأعداد من ١ الى ٣ " نلاحظ ظهور صندوق رسالة msgbox يعرض الرقم 1 وعند النقر على زر OK يظهر صندوق رسالة MessageBox آخر يعرض الرقم 2 وهكذا حتى تتعدى القيمة ٣



المطلوب:

- ١- تعديل التدريب السابق لعرض الأعداد من خلال صندوق نص (Textbox)
 - ٢- اضافة صندوق نص الى النموذج باسم (Textbox1)
- لاحظ : سيتم التعديل في الكود Code بحيث تطبع الأرقام داخل صندوق النص TextBox

الكود	شكل النموذج
<p>الكود Code يكتب في اجراء الحدث Click الخاص بزر الأمر Button</p> <pre>Private Sub Button1_click Dim M As Integer القائمة القديمة (الموجودة مسبقا) For M = 1 To 3 Me.TextBox1.Text = Me.TextBox1.Text & M Next M End Sub</pre>	

لاحظ

- ١- الأمر داخل الحلقة التكرارية Me.TextBox1.Text & M والذي يعنى وضع قيمة المتغير M بجانب ماهو موجود بداخل صندوق النص القيمة القديمة (الموجودة مسبقا) (TextBox) وتم ذلك بكتابة معامل الربط & وظيفة معامل الربط : يستخدم في وصل سلسلتين حرفيتين ببعضهما.
- ٢- في حالة كتابة السطر المشار إليه بهذه الطريقة me.textbox1.text=m لن نلاحظ تغير قيمة المتغير (M) أثناء التنفيذ وإنما سيظهر آخر قيمة فقط وهى ٣ . سؤال



١- قم بتشغيل البرنامج

٢- ثم انقر الزر يظهر صندوق النص (TextBox) بالنموذج (Form)

كما يلي :

٣- انقر الزر (Button1) مرة أخرى

لاحظ :

سوف تجد اضافة الأعداد من 1 إلى 3 مرة أخرى الى صندوق النص

كما يلي :

123123

و للتغلب على هذه المشكلة :

١- لذا يتم إضافة الأمر (Me.TextBox1.Text="") قبل الحلقة التكرارية لمسح محتويات صندوق النص (TextBox)

٢- لعرض الأرقام داخل صندوق النص (TextBox) بحيث يكون كل رقم في سطر جديد نقوم بتغيير خاصية (Multiline) الى (True) للسماح بعرض أكثر من سطر في وضع تصميم النموذج.

٣- ما المقصود ب vbCrLf : هو ثابت حرفي يستخدم لإضافة رمز مفتاح الإدخال وإنشاء سطر جديد وهي اختصار لـ vbCrLf=Visual Basic Carriage Return Line Feed

الكود Code يكتب في اجراء الحدث Click الخاص بزر الأمر Button

Private Sub Button1_click

Dim M As Integer

Me.TextBox1.Text= ""

مسح محتويات صندوق النص

For M = 1 To 3

Me.TextBox1.Text = Me.TextBox1.Text & M & vbCrLf

ثابت حرفي يستخدم لإضافة رمز مفتاح الإدخال و سطر جديد

Next M

End Sub



قم بتشغيل البرنامج ثم انقر الزر "عرض الأعداد

من 1 الى 3" أكثر من مرة

ماذا تلاحظ :

نلاحظ مسح محتوى صندوق النص وإعادة كتابة الأرقام مرة أخرى.

قم بالاستفادة من التدريبين السابقين لطباعة جدول ضرب 3 بصندوق النص (TextBox):

الكود Code يكتب في إجراء الحدث Click الخاص بزر الأمر Button

```
Private Sub Button1_click
Dim M , Product As Integer
```

```
Dim Str As String
```

الاعلان عن المتغيرات

```
Me.TextBox1.Text = ""
```

مسح محتويات صندوق النص

```
For M = 1 To 12
```

```
Str = 3 & " X " & M & " = "
```

لاحظ : وضع X و = بين علامتى تنصيص لتطبع كما هي و لم يوضع المتغير M ليتم التعويض عنه بقيمة فى المعادلة

```
Product = 3 * M
```

```
Me.TextBox1.Text = Me.TextBox1.Text & Str & Product & vbCrLf
```

```
Next M
```

ثابت حرفي يستخدم لإضافة رمز مفتاح الإدخال وسطر جديد

```
End Sub
```

لاحظ:

١- تم التعريف او الاعلان عن متغير رقمى صحيح باسم (product) و ذلك لتخزين حاصل الضرب به مع كل تغيير يحدث على قيمة المتغير (M) (زيادة قيمتها)

٢- تم التعريف او الاعلان عن متغير باسم (str) كمتغير حرفى String و ذلك لتخزين شكل جملة حاصل الضرب $Str = 3 \& " X " \& M \& " = "$ حيث تم وضع علامتى X و = بين علامتى تنصيص و ذلك لتطبع كما هي بنفس الشكل كسلسلة نصية مع كل تغير فى قيمة المتغير (M) و لم يوضع المتغير M بين علامتى تنصيص ليتم التعويض عنه بقيمة الرقمية التى تتغير بمقدار الزيادة فى المعادلة كالتالى $3 \times 1 =$ (و $3 \times 2 =$ وهكذا)

٣- تم الربط بين محتوى صندوق النص TextBox بقيمة المتغير str بقيمة المتغير product بمعامل الربط & .

٤- سؤال : لماذا تم استخدام المتغيرات و الاعلان عنها فى البرنامج ؟

و ذلك للتبسيط و لتسهيل تتبع كود Code البرنامج أثناء التشغيل

يمكن كتابة الكود Code بدون استخدام هذه المتغيرات Variables : تابع

```
Private Sub Button1_click
```

```
Dim M As Integer
```

```
Me.TextBox1.Text = ""
```

```
For M = 1 To 3
```

```
Me.TextBox1.Text = Me.TextBox1.Text & 3 & "X" & M & " = " & 3 * M & vbCrLf
```

```
Next M
```

القيمة الموجودة مسبقاً

شكل حاصل الضرب

ناتج حاصل الضرب

انشاء سطر جديد



قم بتطوير البرنامج بالتدريب السابق لعرض جدول الضرب الخاص بأي رقم يتم ادخاله من خلال مربع نص (TextBox)

كما يظهر في الشكل التالي

قم بإنشاء مشروع Project يحتوي على

نموذج Form و اداتى مربع نص TextBox

و اداة عنوان Label و زر Button

و النص المعروض عليه كلمة

عرض جدول الضرب

الكود Code يكتب في اجراء الحدث Click الخاص بزر الأمر Button

Private Sub Button1_click

Dim M , Product, NUM As Integer

الاعلان عن المتغيرات

Dim Str As String

NUM = TextBox2.Text

جعل المتغير NUM قيمته تساوى العدد المدخل فى صندوق

Me.TextBox1.Text = ""

مسح محتويات صندوق النص

For M = 1 To 12

Str = NUM & " X " & M & " = "

لاحظ : وضع X و = بين علامتى تنصيص لتطبع كما هى و لم يوضع المتغير M او المتغير NUM ليتم التعويض عنهما بقيمهم فى المعادلة

Product = NUM * M

Me.TextBox1.Text = Me.TextBox1.Text & Str & Product & vbCrLf

Next M

ثابت حرفي يستخدم لإضافة رمز مفتاح الإدخال وسطر جديد

End Sub

لاحظ:

١- تم التعريف او الاعلان عن متغير باسم (NUM) كمتغير رقمى صحيح و ذلك لتخزين محتويات صندوق النص

TextBox2 كما بالشكل NUM = TextBox2.Text

٢- تم اعلان المتغير NUM مكان القيمة الرقمية ٢ فى عملية الضرب و فى سطر المخرجات فى TextBox1 و لم توضع بين علامتى تنصيص حتى تاخذ القيمة الرقمية المدخلة بواسطة المستخدم

التحكم في البداية والنهاية ومقدار الزيادة في جملة For.. Next

لاحظ:

١- في الأمثلة السابقة كانت قيمة البداية دائماً أصغر من قيمة النهاية ومقدار زيادة المتغير Variable دائماً

تساوى ١

٢- يمكننا إضافة كلمة (Step) والتي تعنى الخطوة أو مقدار الزيادة فيمكن جعل قيمة البداية أكبر من قيمة النهاية

وفى هذه الحالة يجب أن تكون قيمة الزيادة (step) سالبة

٣- يمكن لكل من قيمة البداية أو النهاية أو الزيادة أن تكون أرقام صحيحة أو عشرية أو متغيرات :

استخدام جملة For...Next

الصيغة العامة لهذه الجملة:

For Variable = Start To End [Step N]
Code
Next [Variable]

بداية التكرار

نهاية التكرار

قيمة الزيادة Step النهاية To البداية = العداد
الجميل المراد تكرارها
Next

بالعربي

لاحظ:

١- فى الصيغة العامة تم وضع بعض الكلمات بين قوسين مربعين مثل [Step N] و [Variable] بهذا الشكل [] وهذا للدلالة على أنها اختيارية أى يمكن عدم كتابتها

٢- و فى حالة عدم كتابتها تأخذ فى هذا الحالة قيمة افتراضية

٣- فى حالة عدم استخدام أو كتابة مقدار الزيادة (Step) فان قيمة العداد تزيد بمقدار 1 بعد كل تكرار

٤- وفى حالة عدم كتابة اسم المتغير (Variable) مع (Next) تكون بنفس اسم المتغير المحدد مع (For) .

٥- تستخدم جملة For...Next فى تكرار كود Code محدد لعدد من المرات معروف مسبقاً

أمثلة بسيطة وسهلة و متنوعة

مثال ١ :

أكتب كود برنامج يقوم بتغيير قيمة العداد Counter من 5 إلى 50 بزيادة 2 في كل مرة باستخدام أمر ForNext
For Counter = 5 To 50 Step 2

مثال ٢ :

أكتب كود برنامج يقوم بتغيير قيمة العداد i من 1 إلى 100 بزيادة 1 في كل مرة باستخدام أمر For...Next
For i = 1 To 100 Step 1
Or
For i = 1 To 100

مثال ٣ :

أكتب كود برنامج يقوم بتغيير قيمة العداد i من 10 إلى 2 بإضافة 2- في كل مرة باستخدام أمر For...Next
For i = 10 To 2 Step -2

مثال ٤ :

أكتب كود برنامج يقوم بتغيير قيمة العداد i كالتالي (11 , 13 , 15 , 17 , 19) باستخدام أمر For...Next .
For i = 11 To 19 Step 2

مثال ٥ :

أكتب كود برنامج يقوم بتغيير قيمة العداد i كالتالي (49 , 42 , 35 , 28 , 21) باستخدام أمر For...Next .
For i = 49 To 21 Step -7

أمثلة :

م	المثال	الكود
١	لعرض الأعداد الفردية من ١ إلى ١٠	Dim I As Integer For I = 1 to 10 Step 2 Me.TextBox1.Text = Me.TextBox1.Text & I & vbCrLf Next
٢	لعرض الأعداد الزوجية من ٢ إلى ١٠	Dim I As Integer For I = 2 to 10 Step 2 Me.TextBox1.Text = Me.TextBox1.Text & I & vbCrLf Next
٣	لعرض الأعداد التي تقبل القسمة على ٣ من ٣ إلى ٢٠	Dim I As Integer For I = 3 to 20 Step 3 Me.TextBox1.Text = Me.TextBox1.Text & I & vbCrLf Next
٤	لعرض الأعداد الزوجية مرتبة تنازليا من ١٠ إلى ١	Dim I As Integer For I = 10 to 1 Step -2 Me.TextBox1.Text = Me.TextBox1.Text & I & vbCrLf Next
٥	لعرض الأعداد من 1.50 إلى 0.5 بتناقص 0.05 كل مرة.	Dim I As Single For I = 1.5 to 0.5 Step -0.05 Me.TextBox1.Text = Me.TextBox1.Text & I & vbCrLf Next
٦	لعرض الأعداد من 1 إلى قيمة B بمعدل زيادة قيمة C	Dim I , B , C As Integer For I = 1 to B Step C Me.TextBox1.Text = Me.TextBox1.Text & I & vbCrLf Next

نستنتج من الأمثلة بالجدول السابق أن:

١. يمكن تحديد معدل الزيادة للمتغير بكلمة (Step) ثم كتابة الرقم.
٢. معدل الزيادة يجب أن يكون سالبا في حالة وجود قيمة البداية أكبر من قيمة النهاية (مثال ٤ ، ٥)
٣. يمكن لقيمة البداية أو النهاية أو معدل الزيادة أن يكون عدد عشري وفي هذه الحالة يجب تعريف متغير الحلقة من نوع يقبل الكسور العشرية مثل النوع (Single) (مثال ٥) .
٤. يمكن لأي من قيمة البداية أو النهاية أو معدل الزيادة أن يكون متغير (Variable) (مثال ٦) .

استخدام جملة Do while

مقدمة :

- ١- لقد تعلمت تكرار كود Code محدد لعدد من المرات معروف مسبقاً باستخدام جملة For..next
- ٢- جملة Do while... loop تستخدم لتكرار كود Code محدد لعدد من المرات غير معروف مسبقاً بناءً على شرط معين لذلك هي مفيدة في حالة عدم معرفة عدد مرات التكرار بشكل قاطع على سبيل تكرار استقبال أسماء في مربع النص TextBox والخروج من الحلقة التكرارية عند إدخال عبارة معينة.

وظيفة جملة Do while... loop

تستخدم هذه الجملة في تكرار كود Code محدد لعدد من المرات غير معروف مسبقاً بناءً على شرط معين

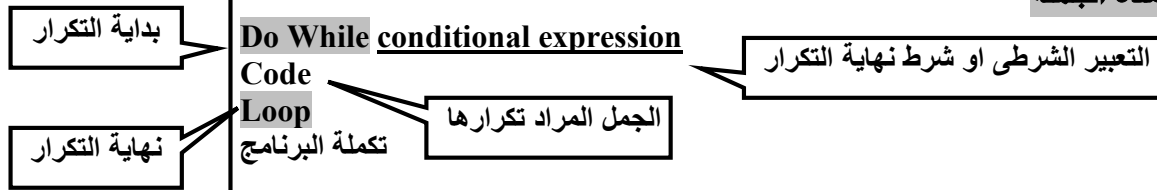
فائدتها

هي مفيدة في حالة عدم معرفة عدد مرات التكرار بشكل قاطع ويتم تنفيذها بناءً على تحقق شرط معين

مثال :

- ١- تكرار استقبال أسماء طلاب المدرسة في مربع النص TextBox والخروج من الحلقة التكرارية عند إدخال عبارة معينة
- ٢- تكرار الذهاب إلى المدرسة يومياً بناءً على شرط معين ألا وهو استمرار الدراسة (طالما الدراسة مستمرة)

الصيغة العامة لهذه الجملة



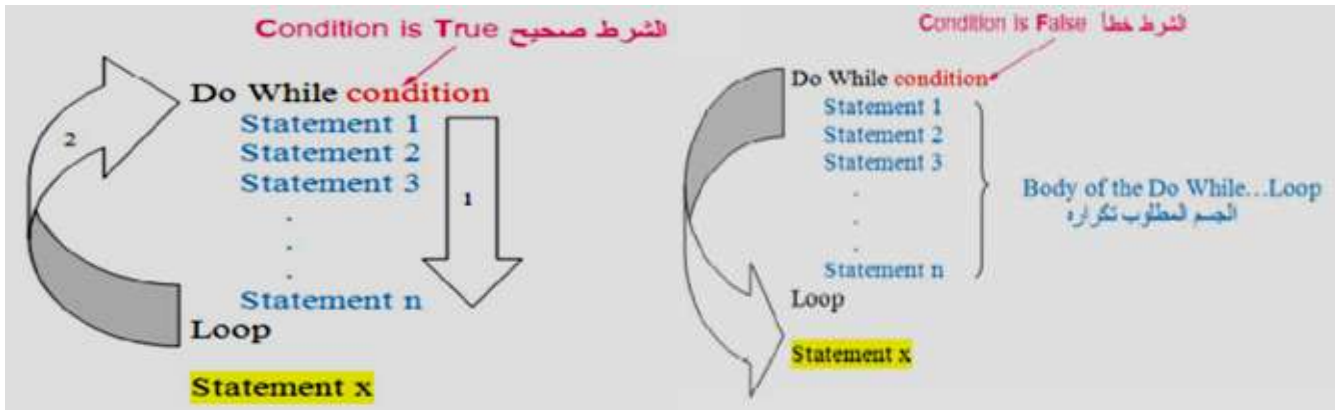
لاحظ:

- ١- في هذا الأمر لا نعرف عدد المرات التي سيتم تكرارها لأن عدد مرات التكرار هنا يعتمد على شرط معين
- ٢- المقصود بـ (do while) تنفيذ الكود (Code) طالما التعبير الشرطي صحيح (True) .
- ٣- يتم تنفيذ الأمر Do While conditional expression طالما الشرط أو التعبير الشرطي متحقق (صحيح True) وعندما لا يتحقق الشرط أو التعبير الشرطي (خطأ False) يتم الخروج من الدورة Loop وتنفيذ ما بعدها مباشرة
- ٤- في حالة عدم تحقق الشرط False لا يتم تنفيذ أي أوامر داخل الدورة ويتم الخروج منها وتنفيذ ما بعدها Loop

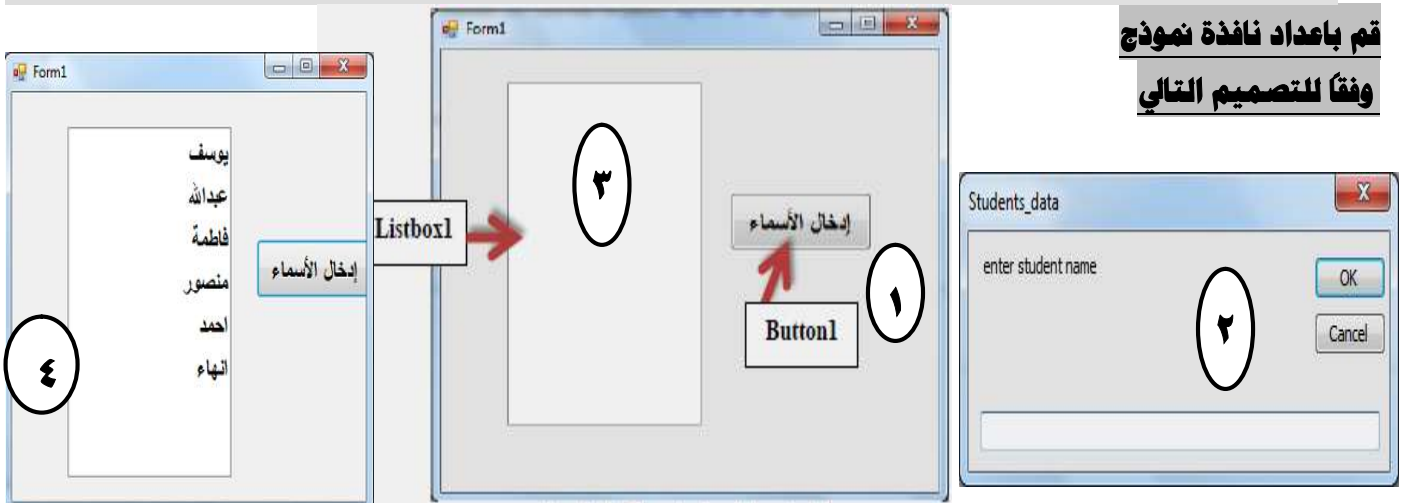
شرح الصيغة العامة

في كل مرة يصل فيها البرنامج إلى Do While يقوم باختبار التعبير الشرطي conditional expression

- أ- فإذا كان التعبير الشرطي خطأ False يتم تنفيذ أول أمر بعد كلمة Loop مباشرة
- ب- أما إذا كان التعبير الشرطي صحيح يتم تنفيذ مجموعة الأوامر التي تلي التعبير الشرطي وحتى آخر أمر قبل كلمة Loop عندما يصل البرنامج إلى كلمة Loop فإنه يعود مرة أخرى إلى Do While conditional expression لاختبار التعبير الشرطي وهكذا وطبقاً لقيمة الشرط فإذا أن تكون True فيستمر التكرار أو تكون False فينتهي التكرار كما في الشكل.



قم باعداد نافذة نموذج
وفقاً للتصميم التالي



المطلوب:

- ١- عند النقر على الزر Button يظهر صندوق ادخال InputBox لإدخال الاسم الأول
- ٢- بعد كتابة الاسم الأول ونقر زر ok أو ضغط مفتاح الإدخال يضاف إلى ListBox1 و يظهر صندوق الإدخال InputBox مرة أخرى لإدخال الاسم الثاني وهكذا
- ٣- عند ادخال كلمة إنهاء يتم إنهاء إدخال الأسماء.
- ٤- يلاحظ في هذا التدريب وجود حلقة تكرارية غير معلوم عدد التكرارات بها حيث ينتهي التكرار عند إدخال قيمة معينة وهي كلمة " إنهاء " .

يكتب الكود في إجراء الحدث Click الخاص بالأداة Button1 كما يلي

```
Private Sub Button1_Click
```

```
Dim X As String
```

```
Do While X <> "إنهاء"
```

```
X = InputBox ( " enter student name " )
```

```
M . ListBox . Items.Add ( X )
```

```
Loop
```

```
End Sub
```

بداية الحلقة التكرارية

دالة InputBox لاستقبال قيمة من
مستخدم البرنامج و تخزينها في المتغير X

لإضافة العنصر الذي كتبه المستخدم الى الاداة
List Box عن طريق الوسيلة Items.Add (X)

نهاية الحلقة التكرارية

شرح الكود السابق

- ١- قم بتشغيل البرنامج ثم انقر الزر " أدخل الأسماء " يظهر صندوق إدخال InputBox لإدخال الاسم الأول
- ٢- يقوم البرنامج باختبار الشرط فإذا كانت قيمة الشرط صحيحة أي لا تساوى كلمة " انتهاء " ينفذ الكود الذى يلي Do While وهو اضافة العنصر الذى كتبه المستخدم الى الاداه List Box عن طريق الوسيلة (X) Items.Add
- ٣- عندما يصل البرنامج الى كلمة Loop فانه يعود مرة اخرى الى Do While conditional expression لاختبار التعبير الشرطى وهكذا وطبقا لقيمة الشرط فيستمر التكرار طالما تكون True
- ٤- عندما يدخل المستخدم كلمة انتهاء فيختبر البرنامج قيمة الشرط فتكون False فينتهي التكرار

لاحظ:

الدالة Input Box

وظيفتها :

استقبال قيمة من مستخدم البرنامج و ارجاع القيمة المدخلة فى متغير من النوع (String)

حيث :

١. تعرض نافذة تحتوي علي إشارة إدخال ومربع كتابة لمستخدم البرنامج لكي يدخل ما يريد (شكل ٢)
٢. كتابة المستخدم لإشارة الإدخال هي معامل الدالة (Arguments) و المعلومات التى يدخلها المستخدم فى مربع الكتابة تعتبر ايضا ارجاع القيمة (Return Value)
٣. البيانات التى يدخلها مستخدم البرنامج تكون مدخلات حرفية من النوع (String) حتى لو أدخل المستخدم أرقاما والمخرجات Return Value تكون ايضا من النوع " String " و على المبرمج ان يقوم بتخزينها أو طباعتها

لاحظ

بعد إدخال كلمة انتهاء سوف تلاحظ الخروج من الحلقة ولكن بعد اضافة كلمة انتهاء فى ListBox رغم أنها شرط الخروج

من الحلقة (شكل ٤)

السبب : حيث أن أمر الإضافة قد حدث قبل الوصول إلى أول الحلقة الذى يتم فيه اختبار التعبير الشرطى

و للتغلب على هذه المشكلة : يتم إضافة شرط بعد عملية إدخال الاسم وقبل عملية الإضافة كما بالكود

```
Private Sub Button1_Click
```

```
Dim X As String
```

بداية الحلقة التكرارية

```
Do While X <> " انتهاء "
```

دالة InputBox لاستقبال قيمة من مستخدم البرنامج و تخزينها فى المتغير X

```
X = InputBox ( " enter student name " )
```

```
If X = " انتهاء " Then Exit Do
```

شرط اختبار قيمة المتغير X هل تساوى انتهاء ام لا

```
M . ListBox . Items.Add ( X )
```

لاضافة العنصر الذى كتبه المستخدم الى الاداه List Box عن طريق الوسيلة (X) Items.Add

```
Loop
```

نهاية الحلقة التكرارية

```
End Sub
```

خلى بالك :

- ١- معنى الشرط هو عندما تصبح قيمة المتغير (x) تساوى كلمة "إنهاء" ينفذ الأمر (Exit Do) أى الخروج مباشرة بدون انتظار اختبار التعبير الشرطى المكتوب مع (Do While) .
- ٢- يجب كتابة كلمة إنهاء في صندوق الإدخال بنفس التهجى المكتوب في الكود وفي حالة كتابة إنهاء أى بهمزة على الألف لن يخرج من الحلقة التكرارية .

للتغلب على هذه المشكلة

ويمكن تفادى ذلك بكتابة الشرط بهذه الطريقة

"إنهاء" And X<>"إنهاء" Do While X<>

كما يمكنك إضافة أي كلمات أخرى تريد استخدامها للخروج من الحلقة بنفس الأسلوب كمثال

"نهاية" And X<>"نهاية" And X<>"إنهاء" And X<>"إنهاء" Do While X<>

واستخدام المعامل المنطقي And هنا يعنى تنفيذ ما بداخل الحلقة طالما قيمة X لا تساوى كلمة "إنهاء" و لا تساوى كلمة "إنهاء" و لا تساوى كلمة "نهاية" و لا تساوى كلمة "نهاية"

الكائن Timer

تعريفه :

هو ساعة توقيت غير مرئية تتيح لك التعامل مع ساعة الكمبيوتر (Computer Clock) من داخل البرنامج التي تقوم بتنفيذه

استخدام الأداة Timer

أداة المؤقت Timer لتكرار تنفيذ كود Code كل فترة زمنية محددة أو لتنفيذ كود Code لوقت محدد ولذلك هي مفيدة جدا عندما يكون تكرار الكود Code مرتبط بالوقت

استعمال الكائن timer

عندما يكون لديك timer على النموذج فانه ينشط علي فترات زمنية متساوية . وهذه الفترات تسمى (Interval) ونتيجة نشاط Timer يقع الحدث Tick ويمكن الاستفادة من الحدث Tick عن طريق كتابة كود Code وربطها بالحدث Tick

وعند إضافة الأداة Timer إلى نموذج Form لابد من تحديد قيم خصائص Properties هامة له

خصائص الكائن Timer

الخاصية (فعال) Enabled

☐ تسمى خاصية التمكين وتأخذ القيم اما (True او False) فى نافذة الخصائص و الكود

☐ القيمة الافتراضية للخاصية Enabled هي False

☐ تستخدم لكى ينشط timer بعد فترة زمنية محددة اذا اخذت الخاصية Enabled القيمة True
Timer1.Enabled = True

☐ تستخدم لكى لإيقاف timer بعد فترة زمنية محددة اذا اخذت الخاصية Enabled القيمة False
Timer1.Enabled = False

الخاصية (الفترة الزمنية) Interval

يتم وضع المدة التي نريدها ان تمر في هذه الخاصية قبل ان ينطلق حدث التنبيه Tick

- ☐ هي فترات زمنية متساوية تقاس بالمللي ثانية و تتراوح قيمتها بين (٠ و ٦٥٥٣٥) مللي ثانية
- ☐ قيمتها الافتراضية هي ١٠٠ مللي ثانية
- ☐ الثانية الواحدة = ١٠٠٠ مللي ثانية
- ☐ غير مسموح بجعل قيمة هذه الخاصية صفرا
- ☐ يمكنك ان تغير هذه المدة اثناء تشغيل البرنامج
- ☐ كلما مرت هذه المدة أعاد الميقاتي Timer إطلاق الحدث Tick

أحداث الكائن Timer

حدث التنبيه (التكة) Tick

☐ هو الحدث المرتبط بالكائن Timer (الحدث الافتراضي عند النقر المزدوج على الأداة)

☐ يقع او ينطلق كلما مرت الفترة الزمنية المحددة في الخاصية Interval بدون تدخل المستخدم

سؤال: متى يقع الحدث Tick ؟ يقع الحدث Tick نتيجة نشاط Timer و بعد مرور الفترة الزمنية المحددة بـ Interval

ملاحظات هامة

١. يمكن تغير قيمة الخاصية Enabled الى True اثناء التصميم او اثناء التشغيل

٢. يمكن إيقاف Timer او منع وقوع الحدث Tick باعطاء الخاصية Enabled القيمة False

٣. الكائن Timer يعمل في الخلفية فلا يراه او يشعر به المستخدم

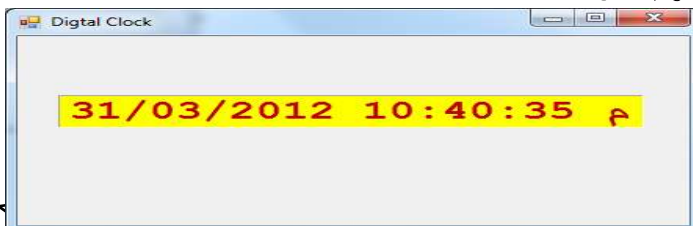
٤. الكائن Timer لا يظهر الا في نمط التصميم

٥. الكائن Timer لا يوضع على النموذج و لكن يوضع اسفل النموذج

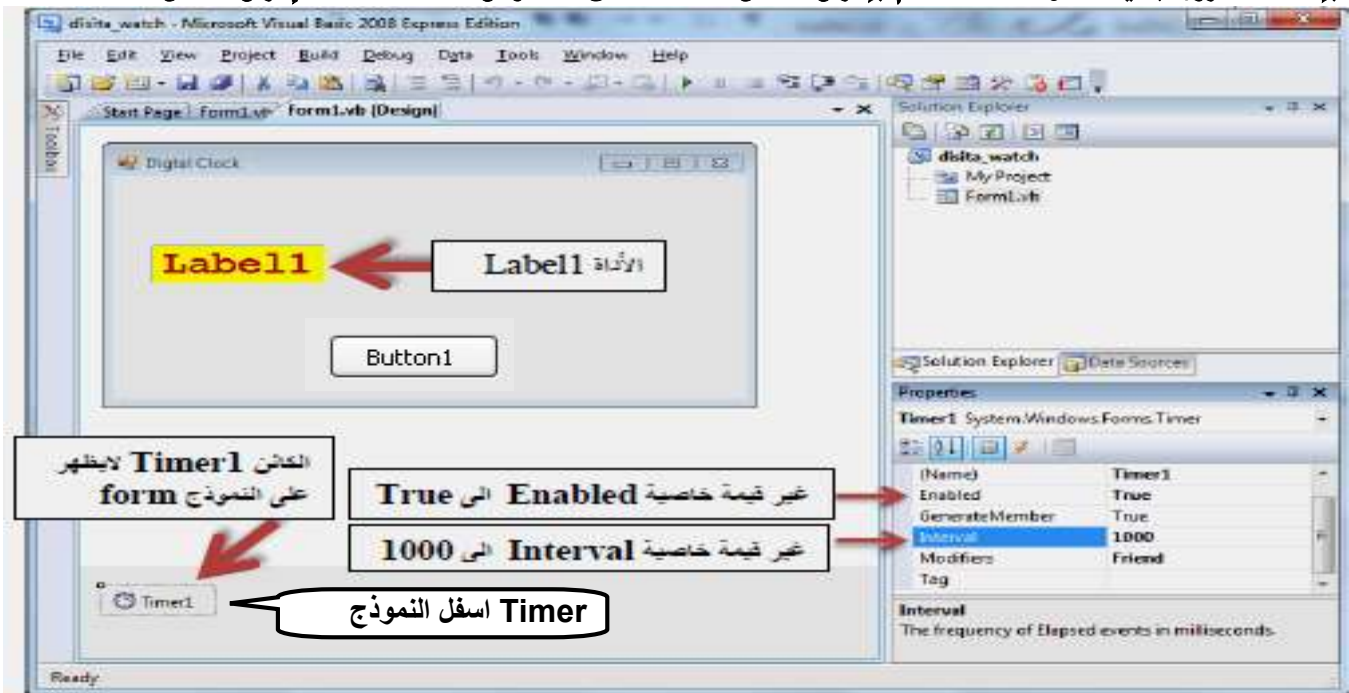
أمثلة على قيم الخاصية Interval و ما يعادلها بالثانية

قيم الخاصية Interval بالمللي ثانية	ما يعادلها بالثانية
١٠٠ مللي ثانية	٠.١ ثانية
٥٠٠ مللي ثانية	٠.٥ ثانية
١٠٠٠ مللي ثانية	١ ثانية
١٥٠٠ مللي ثانية	١.٥ ثانية
٢٠٠٠ مللي ثانية	٢ ثانية
٥٠٠٠ مللي ثانية	٥ ثانية
١٠٠٠٠ مللي ثانية	١٠ ثانية

قم بتصميم نموذج Form يعرض ساعة رقمية كما هو موضح بشكل



قم بإنشاء مشروع جديد New Project قم بإدراج الكائن Timer على النموذج Form1 كذلك إدراج الكائن Label1 كما هو



المطلوب:

- 1- قم بتغيير قيم الخصائص Properties من على الرسم كما يمكنك تغيير خصائص الكائن Label لكي يظهر بشكل منسق وجذاب
- 2- انقر نقر مزدوج على الأداة Timer1 للوصول لتنفيذ الكود Code وللمحدث Tick الافتراضي واكتب الكود Code المطلوب كما بالشكل

يكتب الكود في إجراء الحدث Tick الخاص بالأداة Timer1 كما يلي

```
Private Sub Timer1_Tick
```

الحدث Tick الخاص بالأداة Timer1

```
Label1.Text = Now
```

دالة Now لعرض الوقت و التاريخ الخاص بجهاز الكمبيوتر و هي محتوى الخاصية Text

```
End Sub
```

الدالة Now

هي دالة من الدوال المعرفة التي لاتأخذ وسائط ونتاجها هو تاريخ ووقت جهاز الكمبيوتر الحالي لديك

تم تخصيص ناتج الدالة Now للخاصية Text الخاصة بأداة العنوان Label1

خلى بالك :

بعد تشغيل البرنامج ستجد ظهور التاريخ والوقت بعد مرور ١٠٠٠ ميلي ثانية ثم يتم تخصيص القيمة مرة أخرى بعد مرور الثانية الثانية.

يقوم الكود Code المكتوب بالحدث Tick كل ثانية بعرض التاريخ والوقت لذلك يظهر التغير في الثواني وتبدو كأنها ساعة رقميه.

الدالة TimeOfDay

هي دالة من الدوال المعرفة التي لاتأخذ وسائط ونتاجها هو الوقت الحالي فقط لجهاز الكمبيوتر لديك

جعل الكائن Label يعرض الوقت فقط يمكن استبدال الدالة Now بالدالة TimeOfDay

يكتب الكود في إجراء الحدث Tick الخاص بالأداة Timer1 كما يلي

Private Sub Timer1_Tick

الحدث Tick الخاص بالأداة Timer1

Label1.text = Time of Day()

End Sub

دالة Time of Day لعرض الوقت الخاص بجهاز الكمبيوتر و هي محتوى الخاصية Text

يمكن التحكم برمجيا في إيقاف او تشغيل المؤقت Timer بتغيير الخاصية Enabled

كمثال: النقر على الزر Button1 يؤدي إلى إيقاف المؤقت Timer1

Private Sub Button1_Click

Timer1.Enabled = False

كود لإيقاف المؤقت

كمثال: النقر على الزر Button1 يؤدي إلى تشغيل المؤقت Timer1

Timer1.Enabled = True

كود لتشغيل المؤقت

End Sub

تدريبات عملية

قم بإنشاء مشروع Project يحتوي على نموذج Form به أداة عنوان Label تحتوي على نص "انتبه" وعند تشغيل المشروع تظهر أداة العنوان Label لمدة نصف ثانية ثم تختفي لمدة نصف ثانية وهكذا عند النقر على زر إيقاف تظهر أداة العنوان Label بشكل نهائى بدون أن تختفي عند النقر على زر تشغيل تعود للظهور والاختفاء مرة أخرى

المطلوب:

١- اضافة الأداة Timer الى النموذج مع ضبط الخصائص

(الخاصية Enabled و جعل قيمتها True و ضبط الخاصية Interval و جعل قيمتها تساوى ٥٠٠ مللى ثانية)

٢- كتابة الكود داخل إجراء الحدث Tick للأداة Timer1 ويمكن تحقيق الظهور والاختفاء عن طريق الكود بطريقتين

الطريقة الثانية باستخدام جملة If	الطريقة الأولى باستخدام المعامل Not
<pre>If Label1.Visible = True Then Label1.Visible = False Else Label1.Visible = True End If</pre>	<pre>Label1.Visible = Not (Label1.Visible)</pre>

المعامل Not : من المعاملات المنطقية ويستخدم للحصول على عكس الشرط (أى ما بداخل القوسين)

جدول NOT

تعكس الشرط فإذا كان التعبير الشرطي صحيح True يكون الشرط خطأ والعكس صحيح

A	Not A	السبب
True	False	انها تعكس الشرط
False	True	انها تعكس الشرط

قم بإنشاء مشروع Project يحتوي على نموذج Form

كما هو موضح و عند ادخال رقم فى مربع النص TextBox

١- عند النقر على زر "الأعداد الفردية" يتم عرض الأعداد الفردية مرتبة تصاعدياً حتى الرقم المدخل داخل صندوق القائمة ListBox

٢- أما النقر على زر "الأعداد الزوجية" يتم عرض الأعداد الزوجية مرتبة تصاعدياً حتى الرقم المدخل داخل صندوق القائمة ListBox

إرشادات

يمكن برمجة الكود Code إجراء الحدث Click الخاص بزر "أعداد فردية" بطريقتين كما يلي

الطريقة الأولى باستخدام جملة For Next	الطريقة الثانية باستخدام جملة Do While
1) Dim N , I As integer 2) N =TextBox1 . Text 3) ListBox1 . Items . Clear () 4) For I = 1 to N Step 2 5) ListBox1 . Items. Add (I) 6) Next	1) Dim N , I As integer 2) N =TextBox1 . Text 3) ListBox1 . Items . Clear () 4) I = 1 5) Do While I <= N 6) ListBox1 . Items. Add (I) 7) I = I + 2 8) Loop

نلاحظ :

- ١- تم الاعلان عن المتغيرات فى كلتا الجملتين و تم تخصيص القيم التى ستدخل داخل الخاصية Text الخاصة بالاداه TextBox1 الى المتغير N
- ٢- فى جملة For تم تحديد البداية والنهاية و مقدار الزيادة السطر (٤)
- ٣- تم اضافة العدد الى الاداه ListBox1 عن طريق الوسيلة Add الخاصة بالخاصية Items
- ٤- تم اضافة الوسيلة Clear () لمسح محتويات الاداه ListBox و جعلها فارغة من كل العناصر الموجودة بها
- ٥- فى جملة Do While تم تحديد البداية بجعل قيمة I = 1 والنهاية N و مقدار الزيادة I = I + 2 السطر (٤ ، ٧)

تدريب ١ : قم بنفسك بكتابة الكود الخاص بعرض الأعداد الزوجية.

تدريب ٢ : قم بنفسك بكتابة الكود الخاص بطباعة الأعداد تنازلياً بدلاً من تصاعدياً

الحل

- ١- عند كتابة الكود الخاص بعرض الأعداد الزوجية نقوم بكتابة نفس الكود السابق مع تغيير قيمة البداية فقط و جعلها رقم زوجي وليكن (٢)
- ٢- عند كتابة الكود الخاص بطباعة الأعداد تنازلياً بدلاً من تصاعدياً نقوم بجعل قيمة البداية اكبر من قيمة النهاية ومعدل الزيادة يجب أن يكون سالبا

قم بإنشاء مشروع يحتوي على نموذج Form كما هو موضح

١- عند ادخال رقم مثل 5 ثم انقر على زر "مجموع الأعداد الفردية" يعرض المجموع في الأداة. label3

٢- عند النقر على زر "مجموع الأعداد الزوجية" يعرض المجموع الزوجية في الأداة. label3

إرشادات

يمكن برمجة الكود Code إجراء الحدث Click الخاص بزر "مجموع الأعداد الفردية" بطريقتين كما يلي

الطريقة الأولى باستخدام جملة For Next	الطريقة الثانية باستخدام جملة Do While
1) Dim N , I , Sum As integer 2) N =TextBox1 . Text 3) For I = 1 to N Step 2 4) Sum = Sum + I 5) Next 6) Label3 . Text = Sum	1) Dim N , I , Sum As integer 2) N =TextBox1 . Text 3) I = 1 4) Do While I <= N 5) Sum = Sum + I 6) I = I + 2 7) Loop 8) Label3 . Text = Sum

نلاحظ :

١- تم الاعلان عن المتغيرات في كلتا الجملتين و تم تخصيص القيم التي ستدخل داخل الخاصية Text الخاصة بالاداه TextBox1 الى المتغير N

٢- في جملة For تم تحديد البداية والنهاية و مقدار الزيادة السطر (٣)

٣- تم تخصيص المتغير Sum كمخزن لعملية الجمع عن طريق Sum = Sum + I

٤- في جملة Do While تم تحديد البداية بجعل قيمة I = 1 والنهاية N و مقدار الزيادة I = I + 2 السطر (٣ ، ٦)

تدريب ١ : قم بنفسك بكتابة الكود الخاص بعرض مجموع الأعداد الزوجية.

الحل

١- عند كتابة الكود الخاص بعرض مجموع الأعداد الزوجية نقوم بكتابة نفس الكود السابق مع تغيير قيمة البداية فقط وجعلها رقم زوجي وليكن (٢)

أسئلة

السؤال الأول: اختر الإجابة الصحيحة لكل مما يلي:

١. الغرض من استخدام جملة Do while Loop هو:

أ. تكرار كود (Code) حتى يصبح التعبير الشرطي True

ب. تكرار كود طالما الشرط True

ج. تكرار كود طالما الشرط False

٢- في جملة (For...Next) ما وظيفة الأمر Next؟

أ. زيادة قيمة متغير الحلقة بقيمة الزيادة

ب. مقارنة قيمة الزيادة مع قيمة النهاية

ج. كل ما سبق

٣- أفضل حلقة تكرارية تستخدم عند معرفة عدد مرات التكرار مسبقاً هي:

أ. Do While

ب. For Next

ج. Select Case

٤- بعد تنفيذ الكود (Code) التالي فإن ما يتم عرضه في صندوق النص (txt_display) هو:

```
Dim y As Integer
For y = 2 To 10 Step 2
    txt_display.Text = txt_display.Text & y
Next
```

أ. ٢

ب. ٢٤٦٨١٠

ج. ٢٣٤٥٦٧٨٩١٠

٥- بعد تنفيذ الكود (Code) التالي فإن ما يتم عرضه في أداة العنوان (lbl_display) هو:

```
Dim intNum As Integer = 10
Do While intNum < 10
    intNum = intNum + 1
Loop
lbl_display.Text = intNum
```

أ. ١٠

ب. ١١

ج. ٠

٦- بعد تنفيذ الكود (Code) التالي فإن ما يتم عرضه في أداة العنوان (lbl_display) هو:

```
Dim i As Integer, Sum As Integer
For i = 1 To 5
    Sum = Sum + i
Next
lbl_display.Text = Sum
```

أ. ٥

ب. ١٠

ج. ١٥

السؤال الثاني : ضع علامة (✓) أمام العبارة الصحيحة وعلامة (X) أمام العبارة الخطأ:

()	(١) الأمر MsgBox "3" & "X" & "3" يعرض صندوق رسالة به ٩
()	(٢) قيمة (M) بعد انتهاء تنفيذ الحلقة (For m=3 to 1 Step -1) هو 0
()	(٣) في حالة عدم كتابة (Step) مع جملة (For...Next) فهذا يعني أن قيمة الزيادة صفر افتراضياً
()	(٤) في حالة تحديد القيمة 2500 للخاصة (Interval) للأداة (Timer) فهذا يعني 2.5 ثانية
()	(٥) يمكن الخروج من الحلقة التكرارية (Do...While) بناءً على تعبير شرطي.

السؤال الثالث ما هي قيمة المتغير (r) بعد تنفيذ كل كود (Code) مما يلي:

(0.5 - 1 - 1.5)	<pre>Dim r As Single = 0.5 Do While r > 1 r = r + 0.5 Loop MsgBox(r)</pre>
(Overflow - 2 - 0)	<pre>Dim r As Integer = 1 Do While r <= 1 r = r - 2 Loop MsgBox(r)</pre>
(10 - 5 - 0)	<pre>Dim r, sum As Integer For r = 5 To 10 Step -1 sum = sum + r Next MsgBox(r)</pre>

** الفصل الرابع ** الإجراءات Procedures

مقدمة

- ١- عند إضافة نافذة نموذج Form جديدة ينشأ تصنيف Class جديد باسم نافذة النموذج Form
 - ٢- يمكن الإعلان عن إجراءات الأحداث Event procedures في نطاق التصنيف Class
 - ٣- يمكن الإعلان عن المتغيرات variables والثوابت Constant سواء داخل نطاق إجراءات الأحداث Event procedures أو نطاق التصنيف Class
 - ٤- يمكن الإعلان عن ما يسمى بـ Block مثل جمل If...then و For... next وغيرها
- مثال: كود عرض الأعداد الفردية والزوجية

```

Public Class Form1
    Dim total As Integer

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        'عرض الأعداد الفردية من 1 إلى 10
        Dim i As Integer
        Label1.Text = ""
        For i = 1 To 10 Step 2
            Label1.Text = Label1.Text & " " & i
        Next
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
        'عرض الأعداد الزوجية من 1 إلى 10
        Dim i As Integer
        Label1.Text = ""
        For i = 2 To 10 Step 2
            Label1.Text = Label1.Text & " " & i
        Next
    End Sub
End Class

```

بدا الإعلان عن التصنيف

نطاق الإجراءات

بدا الإعلان عن الإجراءات

نهاية الإعلان عن الإجراءات

نطاق التصنيف

الإعلان عن Block

نهاية الإعلان عن التصنيف



واجهة المستخدم ستظهر كما هو موضح

لاحظ:

- ١- تم الإعلان عن تصنيف Class تحت اسم Form1
- ٢- تم الإعلان عن متغيرات variables تحت أسماء total, i
- ٣- إجراءات الأحداث Event procedures تحت اسم Button1_Click, Button2_Click

الإجراء Procedure

هو مجموعة من الأوامر والتعليمات Code تحت اسم ما و عند استدعاء هذا الاسم يتم تنفيذ هذه الأوامر والتعليمات.

أنواع الإجراءات في لغة Visual Basic .NET

- ١- الإجراء الفرعي Sub و هو لا يعود بقيمة
٢- الدالة Function و هي تعود بقيمة

فائدة الإجراء Procedure

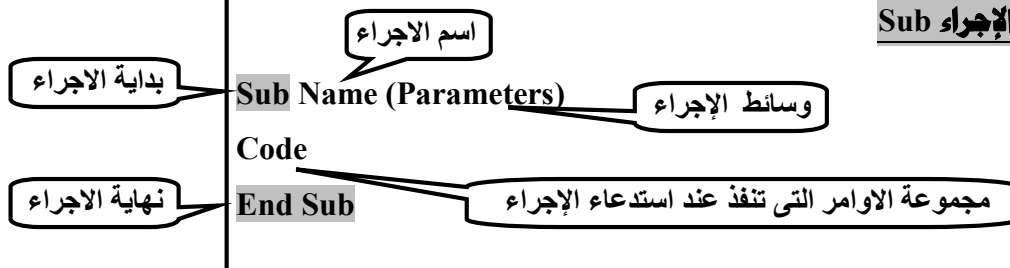
- ١- يعلن عنها مرة واحدة ويمكن استدعاؤها العديد المرات
٢- توفير تكرار كتابة Code في أماكن استدعاء هذه الإجراءات Procedures

الإجراء Sub

الهدف من الاعلان عن الإجراءات Sub

- ١- في حالة إذا ما كان لدينا كود Code سيتكرر كتابته في أكثر من موضع داخل التصنيف Class
٢- بهدف تنظيم كتابة الكود Code حتي يسهل قراءته وفهمه ومن ثم تعديله إذا لزم الأمر

الصيغة العامة للإعلان عن الإجراء Sub



شرح الصيغة العامة للإعلان عن Sub

- ١- يبدأ الإعلان عن الإجراء باستخدام الكلمة Sub
٢- كلمة Name تشير الى اسم الإجراء الذي سوف يستدعي من خلاله
٣- الوسائط Parameters هي القيم التي سوف تستقبل عند استدعاء الإجراء Procedure
٤- وسيط الإجراء Procedure إما أن يكون قيمة مجردة أو متغير Variable أو ثابت Constant أو دالة Function
٥- الكود Code هو مجموعة الأوامر والتعليمات التي ستخزن داخل الإجراء والمطلوب تنفيذها عند استدعاء الإجراء
٦- ينتهي الإجراء بكلمة End Sub

لاحظ:

أن الكود Code المكتوب داخل كل من إجراء الحدث Event procedure الخاص Button1_Click و Button2_Click في الشكل السابق رقم ١ متكرر ما عدا قيمة البداية في عملية التكرار حيث في الزوجي بدأت بالقيمة (٢) وفي الفردي بدأت بالقيمة (١)

الإعلان عن الإجراءات واستدعاه

يمكنك أن تستخدم الإجراءات Sub في عدم كتابة الكود Code

```

Public Class Form1
    Dim total As Integer

    Sub ShowOddOrEven()
        Dim i As Integer
        Label1.Text = ""
        For i = 1 To 10 Step 2
            Label1.Text = Label1.Text & " " & i
        Next
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        'عرض الأعداد الفردية من 1 إلى 10
        ShowOddOrEven()
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
        'عرض الأعداد الزوجية من 1 إلى 10
        ShowOddOrEven()
    End Sub
End Class

```

لاحظ

- ١- يبدأ الإعلان عن الإجراء باستخدام الكلمة Sub
- ٢- اسم الإجراء الذي سوف يستدعي من خلاله الإجراء ShowOddOrEven
- ٣- الكود Code هو مجموعة الأوامر داخل الإجراء والمطلوب تنفيذها عند استدعاء الإجراء
- ٤- ينتهي الإجراء بكلمة End Sub
- ٥- تم استدعاء هذا إجراءات في الأحداث Button1_Click و Button2_Click
- ٦- عند اختبار البرنامج و عند الضغط على مفتاح زوجي ومفتاح فردي نجد أن كل منهما يعطي نفس النتيجة والسبب أن قيمة بداية عملية التكرار واحدة وهي القيمة (١)



قيمة بداية عملية التكرار

```

Sub ShowOddOrEven()
    Dim i As Integer
    Label1.Text = ""
    For i = 1 To 10 Step 2
        Label1.Text = Label1.Text & " " & i
    Next
End Sub

```

الإعلان عن الوسائط Parameters واستخدامه

ولحل هذه المشكلة

- ١- لابد أن يستقبل الإجراء ShowOddOrEven قيمة إما (١) للأرقام الفردية أو (٢) للأرقام الزوجية عند استدعائه
- ٢- إضافة متغير باسم Start كوسيط Parameter و عند استدعائه تعدد قيمته ما إذا كان سيتم عرض الأعداد الفردية أم الزوجية

```
Sub ShowOddOrEven(ByVal Start As Integer)
    Dim i As Integer
    Label1.Text = ""
    For i = Start To 10 Step 2
        Label1.Text = Label1.Text & " " & i
    Next
End Sub
```

الإعلان عن Parameter

استخدام هذا Parameter

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    'عرض الأعداد الفردية من 1 إلى 10
    ShowOddOrEven(1)
End Sub
```

تحديد قيمة Argument

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
    'عرض الأعداد زوجية من 1 إلى 10
    ShowOddOrEven(2)
End Sub
```

لاحظ :

- ١- تم الإعلان إجراء Procedure تحت اسم ShowOddOrEven باستخدام الكلمة Sub والإعلان عن Parameter تحت اسم Start باستخدام الكلمة ByVal حيث تم استخدامه في الكود حتى يمكن تحديد قيمة بداية عملية التكرار وبناء عليه تعرض الأعداد الفردية أو الزوجية
- ٢- تم استدعاء الإجراء ShowOddOrEven مرتين مع إعطاء قيمة مختلفة تعدد هل يتم عرض الأعداد الفردية أم الزوجية، يطلق على هذه القيمة Argument

لاحظ

عند اختبار البرنامج نجد أن مفتاح زوجي ومفتاح فردي عند الضغط على أي منهما يعطي نتيجة مختلفة



لاحظ :

- ١- يمكنك عند الإعلان عن الإجراء Procedure استخدام أكثر من Parameter
- ٢- وظيفة أو استخدام Parameters كوسيلة استقبال قيم من خارج الإجراء غير معلومة مسبقاً وإنما تحدد عند استدعاء هذا الإجراء Procedure
- ٣- في المثال السابق تم استخدام وسيط Parameter واحد لتحديد قيمة البداية فقط وظلت قيمة النهاية في الأمرين موحدة وهي (10) فماذا لو أردنا أن تختلف قيمة النهاية أيضاً كأن نعرض الأعداد الفردية من ١ إلى ١٠ و نعرض الأعداد الزوجية من ٢ إلى ٢٠ مثلاً فما هو الحل ؟ الحل هو الإعلان عن أكثر من Parameters

الإعلان عن أكثر من Parameters

عزيزي الطالب يمكنك تطوير الإجراء ShowOddOrEven بحيث يستقبل قيمة البداية وقيمة النهاية في عملية التكرار كما هو موضح بالشكل

```
Sub ShowOddOrEven(ByVal Start As Integer, ByVal LastValue As Integer)
    Dim i As Integer
    Label1.Text = ""
    For i = Start To LastValue Step 2
        Label1.Text = Label1.Text & " " & i
    Next
End Sub
```

الإعلان عن Parameter

استخدام Parameter

ويترتب على ما سبق أن تعطى قيمتين عند استدعاء الإجراء ShowOddOrEven كما هو موضح

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    'عرض الأعداد الفردية من 1 إلى 10
    ShowOddOrEven(1, 10)
End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
    'عرض الأعداد الزوجية من 1 إلى 10
    ShowOddOrEven(2, 10)
End Sub
```

استدعاء الإجراء (ShowOddOrEven) مع إعطاء قيمتين

قيمة البداية

قيمة النهاية

لاحظ :

- ١- تم الإعلان عن وسيطين Parameter الأول باسم Start والثاني باسم LastValue
- ٢- تم استخدامهما داخل الكود لبداية ونهاية التكرار
- ٣- عند استدعاء الإجراء لابد أن تعطى قيمتين للقيمة البداية وقيمة للنهاية

لاحظ :

- ١- يمكنك استدعاء الإجراء أي عدد من المرات في برنامجك.
- ٢- يمكنك أن تتحكم في عرض الأعداد الفردية أو الزوجية
- ٣- يمكنك أن تتحكم في نطاق الأعداد

فمثلاً

عند إعطاء القيم من (35 و 5) فستعرض الأعداد الفردية من 5 إلى 35 وإذا أعطينا القيم من (45 و 8) فستعرض الأعداد الزوجية من 8 إلى 45

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
    ShowOddOrEven(8, 45)
End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
    ShowOddOrEven(5, 35)
End Sub
```

عرض الأعداد الزوجية من ٨ إلى ٤٥

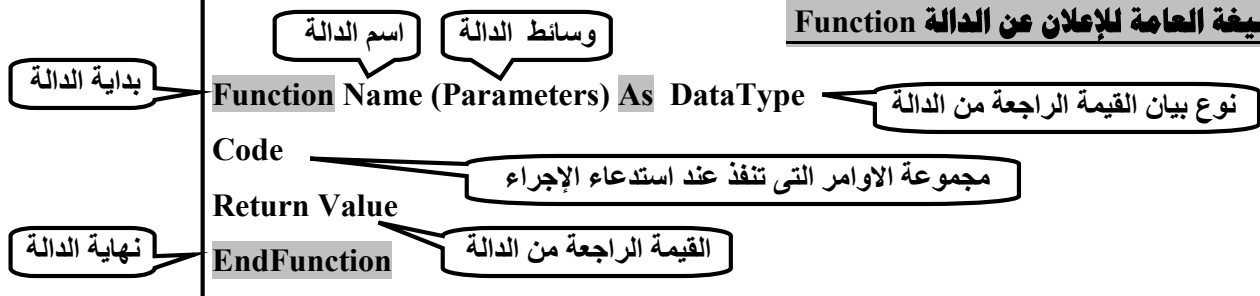
عرض الأعداد الفردية من ٥ إلى ٣٥

الإعلان عن الدالة Function

الهدف من الإعلان عن الدالة Function

- ١- في حالة إذا ما كان لدينا كود Code سينتج منه قيمة نحتاجها
مثال محيط دائرة ، مساحة مربع ، صافي أجر العامل ، الضريبة المستحقة ... الخ
- ٢- القيمة التي تنتج من الدالة يمكن استخدامها أثناء تنفيذ تعليمات البرنامج أو إخراجها للمستخدم

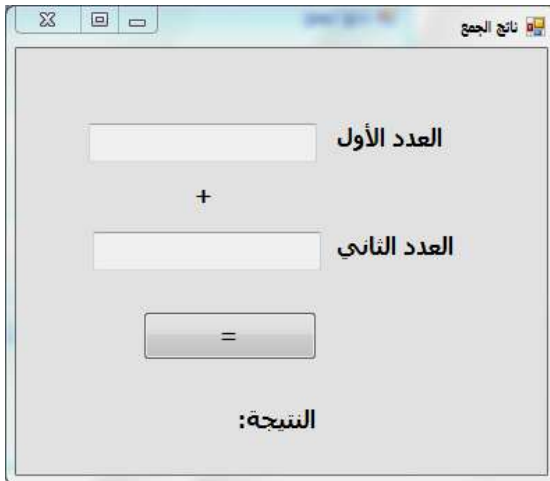
الصيغة العامة للإعلان عن الدالة Function



شرح الصيغة العامة للإعلان الدالة Function

- ١- يبدأ الإعلان عن الدالة باستخدام الكلمة Function
- ٢- كلمة Name تشير إلى اسم الدالة الذي سوف يستدعي من خلاله
- ٣- الوسائط Parameters هي القيم التي سوف تستقبل عند استدعاء الدالة Function
- ٤- DataType تحدد نوع البيان الخاصة بالقيمة الراجعة من الدالة Function
- ٥- الكود Code هو مجموعة الأوامر والتعليمات التي ستنفذ عند استدعاء الدالة Function
- ٦- Return Value هو القيمة الراجعة من الدالة
- ٧- ينتهي الإعلان عن الدالة باستخدام الكلمة EndFunction

تدريبات على الدالة Function



١- قم بإعداد نافذة النموذج Form بالشكل التالي لحساب مجموع رقمين

٢- قم بفتح نافذة الكود Code Window بالضغط على مفتاح F7

ثم اكتب الكود Code

```
Public Class Form5
    Function Sum(ByVal First As Single, ByVal Second As Single) As Single
        Dim total As Single
        total = First + Second
        Return total
    End Function
End Class
```

لاحظ:

- ١- يبدأ الإعلان عن الدالة باستخدام الكلمة Function
- ٢- كلمة Sum تشير إلى اسم الدالة الذي سوف يستدعي من خلاله
- ٣- الوسائط Parameters هما First و Second و (تم الإعلان عنهما باستخدام الكلمة ByVal وتحديد نوعهما داخل القوسين من النوع Single) وكذلك الدالة خارج القوسين من النوع Single (رقم عشري)
- ٤- تم الإعلان عن متغير باسم Total من النوع Single وتخصيصه لجمع قيمتي First و Second
- ٥- إرجاع قيمة المتغير Total باستخدام الأمر Return Total
- ٦- Return Value هو القيمة الراجعة من الدالة
- ٧- نهاية الإعلان عن الدالة باستخدام الكلمة EndFunction
- ٨- القيمة total يتم تخزينها في اسم الدالة Sum

استدعاء الدالة Sum

٣- قم بإنشاء إجراء الحدث الخاص بزر الامر Button1 ثم اكتب الكود Code

```
Private Sub Button1_Click(ByVal sender As
    Dim x As Single = TextBox1.Text
    Dim y As Single = TextBox2.Text
    Label4.Text = Sum(x, y)
End Sub
```

لاحظ :

- ١- تم الإعلان عن متغير باسم X وتخصيص مدخلات المستخدم له في الخاصية Text الخاصة بالأداة TextBox1
- ٢- تم الإعلان عن متغير باسم Y وتخصيص مدخلات المستخدم له في الخاصية Text الخاصة بالأداة TextBox2
- ٣- تم تخصيص قيمة الدالة Sum للخاصية Text الخاصة بأداة التحكم Label4 بعد استقبال القيمتين X و Y

هام :

المتغير Variables :

يمكن تخصيص قيم لها أثناء الإعلان و أثناء سير تنفيذ تعليمات البرنامج وكذلك استخدام القيم المخزنة بها.

الثوابت Constants :

يمكن تخصيص قيم لها أثناء الإعلان فقط وكذلك استخدام القيم المخزنة بها

الدوال Function :

لا يمكن تخصيص قيم لها وإنما تستدعى فتنتج قيمة تخرن بها ثم يمكنك استخدام هذه القيمة المخزنة بها.

لاحظ :

يفضل تسمية الدوال Functions تسمية متعلقة بوظيفتها

- ٤- قم بالضغط على زر F5 ثم أدخل القيم كما هو موضح بالشكل

لاحظ أنه يمكنك أن

- ١- تعلن عن دالة Function
- ٢- تعدد Parameters الخاصة بها
- ٣- تعدد نوع الدالة Function
- ٤- تكتب الكود Code في نطاق هذه الدالة Function
- ٥- ترجع قيمة باستخدام Return

حساب مضروب عدد معين

الإعلان عن الدالة Factorial لحساب مضروب عدد معين كما هو موضح

```
Function Factorial(ByVal Number As Integer) As Integer
    Dim i As Integer
    Dim res As Integer = 1
    For i = 1 To number
        res = res * i
    Next
    Return res
End Function
```


لاحظ:

- ١- يبدأ الإعلان عن الدالة باستخدام الكلمة Function
- ٢- كلمة Factorial تشير الى اسم الدالة الذى سوف يستدعي من خلاله
- ٣- الوسيط Parameter هو Number (تم تحديد نوعه داخل القوسين) وكذلك الدالة خارج القوسين من النوع Integer (رقم صحيح)
- ٤- تم الإعلان عن متغير باسم I كعداد لعملية التكرار
- ٥- تم الإعلان عن متغير باسم res من النوع Integer لتخزين ناتج حساب مضروب العدد و جعل قيمة البداية تساوى واحد
- ٦- ارجاع قيمة المتغير res باستخدام الأمر Return res
- ٧- نهاية الإعلان عن الدالة باستخدام الكلمة EndFunction

استدعاء الدالة Factorial

```
MessageBox.Show(Factorial(5))
```

لاحظ : تم استدعاء الدالة Factorial واعطاء قيمة 5 وعرض الناتج من خلال صندوق رسالة MessageBox

حساب مساحة دائرة

الإعلان عن الدالة Area لحساب مساحة الدائرة كما هو موضح

```
Function Area(ByVal Radius As Single) As Single

    Const x As Single = 22 / 7
    Dim res As Single
    res = x * Radius ^ 2
    Return res

End Function
```

لاحظ:

- ١- يبدأ الإعلان عن الدالة باستخدام الكلمة Function
- ٢- كلمة Area تشير الى اسم الدالة الذى سوف يستدعي من خلاله
- ٣- الوسيط Parameter هو radius (تم تحديد نوعه داخل القوسين) وكذلك الدالة خارج القوسين من النوع Single (رقم عشري)
- ٤- تم الإعلان عن ثابت باسم X ووضع قيمته (٢٢/٧)
- ٥- تم الإعلان عن متغير باسم res من النوع Single لتخزين ناتج حساب مساحة الدائرة
- ٦- ارجاع قيمة المتغير res باستخدام الأمر Return res
- ٧- نهاية الإعلان عن الدالة باستخدام الكلمة EndFunction


```
MessageBox.Show (Area (2) )
```

استدعاء الدالة Area

لاحظ :

تم استدعاء الدالة Area واعطاء قيمة نصف القطر ب (2) وعرض الناتج من خلال صندوق رسالة MessageBox

الدوال المعرفة Predefined Function

تعريفها :

هي دوال معرفة لدى لغة البرمجة Visual Basic.Net يتم استدعاؤها مباشرة

أمثلة على الدوال المعرفة

الدالة Now	الدالة IsNumeric	الدالة Show
------------	------------------	-------------

الدالة Show

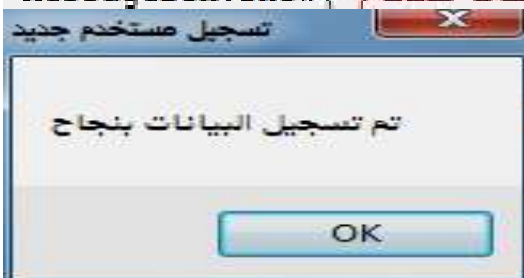
وظيفتها :

تستخدم في إظهار صندوق رسالة MessageBox يتحدد محتوياته حسب الوسائط المعطاه للدالة Show

مثال

عند استدعاؤها يظهر صندوق الرسالة كما هو موضح

```
MessageBox.Show ("تسجيل مستخدم جديد", "تم تسجيل البيانات بنجاح")
```



لاحظ :

١- الوسيط الأول للدالة Show في المثال هو نص الرسالة المطلوب من المستخدم التعامل معه (تم تسجيل البيانات بنجاح)

٢- الوسيط الثاني للدالة Show هو النص الذي يظهر في شريط عنوان مربع الرسالة (تسجيل مستخدم جديد)

٣- يفصل بين المعاملات داخل صندوق الرسالة بعلامة (,)

الدالة IsNumeric

وظيفتها :

١- تستخدم في اختبار قيمة لمعرفة هل هذه القيمة رقمية أم لا

٢- وناتجها إما True في حالة إذا كانت القيمة رقمية و ناتجها False في حالة إذا كانت القيمة غير رقمية

```
Label1.Text = IsNumeric("five")
```

مثال

عند استدعاؤها يظهر الناتج False لأن القيمة " Five " لا يمكن تحويلها لقيمة رقمية

الدالة Now

وظيفتها :

تستخدم في إظهار أو استعراض التاريخ والوقت المسجل بنظام الكمبيوتر

```
MessageBox.Show (Now ( ) )
```

مثال

عند استدعاؤها يظهر صندوق الرسالة التاريخ والوقت الحاليين الخاصين بنظام الكمبيوتر

لاحظ :

- ١- جميع الدوال Function تستخدم في الطرف الايمن من معادلة التخصيص للحصول على ناتجها
- ٢- هناك دوال لا تأخذ أي وسائط مثل الدالة Now() في حين أي دالة Function لابد وأن يكون لها ناتج
- ٣- الإجراءات Sub لا يجوز استخدامها في أي جملة تخصيص
- ٤- إجراء الحدث Event Procedure يعتبر إجراء من النوع Sub
- ٥- وسيط الإجراءات Procedure إما أن يكون قيمة مجردة أو متغير Variable أو ثابت Constant أو دالة Function

هام

قارن بين الاجراء Sub و الدالة Function

الدالة Function	الاجراء Sub
<input type="checkbox"/> تعود بقيمة <input type="checkbox"/> لها ناتج <input type="checkbox"/> تستخدم في الطرف الايمن من معادلة التخصيص للحصول على نتائجها <input type="checkbox"/> بعضها لا يأخذ وسائط مثل الدالة Now	<input type="checkbox"/> لا يعود بقيمة <input type="checkbox"/> ليس له ناتج <input type="checkbox"/> لا يستخدم في اي جملة تخصيص

الهدف من الاعلان عن الإجراءات Sub و الدالة Function

الدالة Function	الاجراء Sub
١- في حالة إذا ما كان لدينا كود Code سينتج منه قيمة نحتاجها ٢- استخدام القيمة التي ستنتج من الدالة أثناء تنفيذ تعليمات البرنامج أو اخراجها للمستخدم	١- في حالة إذا ما كان لدينا كود Code سيتكرر كتابته في أكثر من موضع داخل التصنيف Class ٢- بهدف تنظيم كتابة الكود Code حتي يسهل قراءته وفهمه ومن ثم تعديله إذا لزم الأمر

الفرق بين المتغير Variable و الثابت Constant و الدالة Function

الدالة Function	الثابت Constant	المتغير Variable
لا يمكن تخصيص قيم لها وإنما تستدعي فتنتج قيمة تترن بها	يمكن تخصيص قيم لها أثناء الإعلان فقط	يمكن تخصيص قيم لها أثناء سير تنفيذ التعليمات البرنامج

أسئلة وتدريبات

السؤال الأول:

ضع علامة (✓) أمام العبارة الصحيحة، وعلامة (✗) أمام العبارة الخطأ:

(١)	يعلن الإجراء (Procedure) مرة واحدة ويستدعي أي عدد من المرات.	(...)
(٢)	الدوال المعرفة (Predefined) يلزم الإعلان عنها أولاً.	(...)
(٣)	القيمة الراجعة للدالة (Area) هي (Single) كما هو موضح في إعلانها Function Area(ByVal radius As Single) As Integer	(...)
(٤)	وسائط (Parameters) الإجراء (Procedure) من خلالها تستقبل قيم من خارج الإجراء (Procedure).	(...)
(٥)	تستخدم الإجراءات (Sub) في طرفي جملة التخصيص أما الدوال (Functions) لا يجوز استخدامها في أي جملة تخصيص.	(...)
(٦)	الإجراء (Sub) لا يعود بأي قيمة أما الدالة (Function) تعود بقيمة.	(...)
(٧)	يشترط عند الإعلان عن الإجراءات (Procedures) أن تستخدم وسيط واحد على الأقل (Parameters).	(...)
(٨)	وسيط الإجراء (Procedure) إما أن يكون قيمة مجردة أو متغير (Variable) أو ثابت (Constant) أو دالة (Function).	(...)

السؤال الثاني:

اكتب أمام كل عبارة بالعمود (ب) الرقم المناسب من العمود (أ)

م	(أ)	م	(ب)
(١)	الإجراء (Sub)		تستخدم في الطرف الأيمن من جملة التخصيص ولا يخصص لها أي قيمة.
(٢)	الدالة (Function)		تستخدم في الطرف الأيمن من جملة التخصيص ويخصص لها قيمة عند الإعلان فقط.
(٣)	المتغيرات (Variables)		لا يستخدم في جملة التخصيص نهائياً.
(٤)	الثوابت (Constants)		تستخدم في طرفي جملة التخصيص.

السؤال الثالث:

استخرج من الكود (Code) الآتي:

```

Sub ShowOddOrEven (ByVal Start As Integer)
    Dim i As Integer
    Label1.Text = ""
    For i = Start To 10 Step 2
        Label1.Text = Label1.Text & " " & i
    Next
End Sub

```

١- اسم الإجراء :

٢- وسائط هذا الإجراء وأنواعها:

السؤال الرابع:

استخرج من الكود (Code) الآتي:

```

Function Area (ByVal Radius As Single) As Single

    Const x As Single = 22 / 7
    Dim res As Single
    res = x * Radius ^ 2
    Return res

End Function

```

١- اسم الإجراء :

٢- وسائط هذا الإجراء:

٣- القيمة المرجعة:

٣- نوع البيان الخاص ب:

أ- الدالة:

ب- وسيط الدالة:

ج- القيمة المرجعة:

السؤال الخامس:

استخرج من الكود (Code) الآتي:

```
Public Class Form5
    Function Sum(ByVal First As Single, ByVal Second As Single) As Single
        Dim total As Single
        total = First + Second
        Return total
    End Function
End Class
```

١- اسم الإجراء :

٢- وسائط هذا الإجراء:

٣- القيمة الراجعة:

٣- نوع البيان الخاص ب:

أ- الدالة:

ب- وسائط الدالة:

ج- القيمة الراجعة:

السؤال السادس:

أكتب أي من الآتي إجراء (Sub) أو دالة (Function)

أ-

```
Sub ShowOddOrEven(ByVal Start As Integer, ByVal LastValue As Integer)
    Dim i As Integer
    Label1.Text = ""
    For i = Start To LastValue Step 2
        Label1.Text = Label1.Text & " " & i
    Next
End Sub
```

.....

ب-

```
Function Area(ByVal Radius As Single) As Single

    Const x As Single = 22 / 7
    Dim res As Single
    res = x * Radius ^ 2
    Return res

End Function
```

.....

ج-

```
Label1.Text = IsNumeric("five")
```

.....

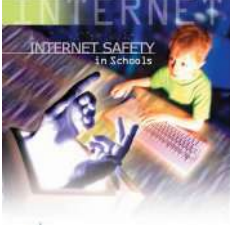
السؤال السابع:

استخرج أسماء الدوال (Functions) من الكود (Code) الآتي:

```
MessageBox.Show(Factorial(5))
```

.....

.....

**** الفصل الخامس **** **التعدي الإلكتروني Cyber bullying****مقدمة**

أخلاقيات التعامل مع الإنترنت بل مع كافة وسائل المعلومات والاتصالات أصبحت من موضوعات الساعة التي تهم الأفراد والمجتمعات والدول والعلاقات فيما بينهم، لذا نسعى في هذا الفصل إلى رفع الوعي لدى أبنائنا وبناتنا بأهمية أخلاقيات التعامل مع الإنترنت، وإكسابهم المعلومات والمهارات اللازمة لرفع درجة سلامتهم الشخصية فيما يتعلق بالتعدي عبر الإنترنت.

خلى بالك : الانترنت سلاح ذو حدين ؟ دليل على ذلك

- ١- الجانب الإيجابي يمكن من خلال الإنترنت نستطيع أن نتعلم ... نتثقف ... نتسلى ... نتواصل ... نتحاور...
- ٢- الجانب السلبي يمكن أن نحصل منها على معلومات خطأ ويمكن أن نتعرف على أشخاص سيئين.

بعض المفاهيم التي تعلمناها مسبقا و التي يجب لنا الإلمام بها

متصفح الإنترنت : هو برنامج يمكنك من خلاله استعراض المواقع و صفحات الويب المختلفة.

URL : يقصد بع عناوين مواقع الويب التي يمكنك من استعراض المواقع و صفحات الويب المختلفة

البريد الإلكتروني : هو احد خدمات الانترنت و يمكنك من إرسال واستقبال الرسائل الالكترونية و الملفات المرفقة لشخص او أكثر في نفس الوقت .

المحادثة الفورية : يمكنك من محادثة الآخرين كتابة أو صوت أو رؤية (صورة) بشكل فوري باستخدام أحد برامج المحادثة.

الفيروسات : عبارة عن برامج ضارة تقوم تصيب نظام الكمبيوتر الغرض منها (حذف - تدمير- تغيير بيانات النظام - تعطيل البرامج)

الاختراق : يعني قيام مجموعة من الأشخاص باستغلال نقاط ضعف نظام الكمبيوتر ويقوموا باختراق جهازك.

انتحال الهوية : يقصد بها أن يقوم شخص ما بانتحال شخصيتك على الإنترنت

مثال : يقوم بإجراء محادثة مع أصدقائك بدلاً منك.

تعريف التعدي الإلكتروني

عبارة عن سلوك عدواني متعمد يستخدم الوسائط الإلكترونية بغرض :

- ١- التحرش .
- ٢- المضايقة .
- ٣- إخراج
- ٤- التخويف
- ٥- التهديد

ولفهم مدى انتشار هذه الظاهرة

ففي عام ٢٠٠٥، قامت كلية إدارة العدالة الاجتماعية بجامعة "فلوريدا أتلانتيك" بالولايات المتحدة الأمريكية، بدراسة عن طرق استخدام الإنترنت من قبل ١٥٠٠ مراهق، وعند السؤال عن تجاربهم مع التعدي الإلكتروني تبين أن حوالي: ٨٠٪ من المراهقين أقرروا حدوث تعدي الكتروني عند استخدامهم الإنترنت. ٣٢٪ من الذكور و ٣٦٪ من الإناث أقرروا مواجهة نوع من أنواع التعدي الإلكتروني. ١٢٪ أقرروا مواجهة التعدي الإلكتروني بصيغة تهديد. ٥٪ أقرروا خوفهم على سلامتهم.

الوسائط الإلكترونية للتعدي

الوسائط الإلكترونية التي يستخدمها المعتدي الإلكتروني كثيرة منها ما يلي:

١- البريد الإلكتروني. E.Mail

من خلاله يمكنك إرسال واستقبال الرد وإعادة توجيه الرسائل الكترونيا مع إمكانية إرفاق ملفات بتلك الرسائل.

٢- المنتديات الإلكترونية. Forums

فيها يتم طرح موضوعات للمناقشة فيتم إرسال المشاركة وتجد الردود من الآخرين.

٣- الرسائل الفورية. Instant Message

من خلالها يحدث اتصال فوري بين طرفين أو أكثر عن طرق الكتابة أو للمحادثة الصوتية أو المرئية.

٤- التدوين الإلكتروني. Bloging

عبارة عن سجل الإلكتروني خاص يدون به صاحب المدونة ما يشاء.

٥- المواقع الاجتماعية مثل Face Book

تسمح بنشر معلومات خاصة للشخص صاحب الحساب.

٦- الهاتف المحمول.

عبارة عن هاتف محمول يمكن استخدامه بالاتصال بالآخرين مع خدمات أخرى مصاحبة.

٧- مواقع الإنترنت.

عبارة عن مجموعة من صفحات الويب التي قد تحتوي على نصوص وصور أو أصوات أو فيديو.

أشكال التعدي الإلكتروني**التخفي الإلكتروني (Anonymity):**

يقصد به : استخدام أسماء مستعارة تخفي شخصية المعتدي الإلكتروني بغرض الإفلات من العقاب
مثال :



عمرو عضو في إحدى المنتديات التي تهتم بالرياضة ويشارك بالموضوعات التي تخص فريقه الذي يشجعه بروج رياضية كلما أرسل تعليق تلقى رد من " الصقر المفترس " في صورة تعليقات غير لائقة والمشكلة انه لا يعرف من المعتدي.

**المضايقات الإلكترونية (Harassment):****يقصد به :** رسائل عدائيه موجهة ضد شخص أو أكثر**مثال :**

رامي يقوم بمحادثة زملائه مستخدماً خدمة ال chat كلما أصبح متاح. حادثة أحد المسجلين لديه بشكل مزعج ومستمر مضيقاً لوقته وهو يريد أن يتجنب تلك المضايقة..... ومنه فإن المضايقات الإلكترونية

**الملاحقة الإلكترونية (Cyber stalking):****يقصد به :** هي شكل من أشكال المضايقات الإلكترونية ولكن بشكل متكرر**مثال :**

تقوم ياسمين يومياً بفتح البريد الإلكتروني الخاص بها يومياً وتجد رسائل إعلانية لمنتجات وخدمات بكميات تستهلك منها الوقت الذي تحتاجه في قراءة الرسائل العامة..... وعليه

**السب أو القذف الإلكتروني (Flaming):****يقصد به :** عبارة عن نشر كلمات عدائيه و مبتذلة ضد شخص أو أكثر**مثال ١:**

أحد الأشخاص المشهورين يقرأ إحدى الصحف (التي تستخدم مفهوم الويب ٢ التي تتيح لقراءها أن ترسل تعليق على الموضوعات المنشورة) وجد مقال يتكلم عنه وهو يقرأ تعليقات القراء وجد إحدى التعليقات التي تسبه بشكل مبتذل.

مثال ٢:

أحد الأشخاص يتلقى رسائل نصية على هاتفه فيها شتائم.

التشهير الإلكتروني (Outing):**يقصد به :** عبارة عن نشر معلومات عن شخص محدد أو أكثر بشكل مسيء**مثال :**

أحد الطلاب قام بزيارة زميلة وجده جالس أمام جهاز الكمبيوتر الخاص به فقام بنسخ الصور العائلية الخاصة بزميلة وأرسلها لبريده دون أن يعرف زميله ثم في وقت لاحق قام هذا الولد السيئ بنشر هذه الصور على المدونة التي قام بإنشائها فسببت إحراج للولد صاحب الصور ولعائلته.

الاستثناء الإلكتروني (Exclusion):**يقصد به :** عبارة عن تجاهل شخص أو أكثر من خلال وسائط الكترونية**مثال :**

هناك مجموعة من الأولاد يستخدمون الكمبيوتر في ممارسة ألعاب الشبكة وكان عددهم ٥، وجد أنه يتم تجاهل أحدهم كأنه لا يلعب معهم مع أنه موجود على الشبكة معهم.

التهديد الإلكتروني (Cyber threats):**يقصد به :** عبارة عن إرسال رسائل الكترونية تحمل تهديد أو وعيد لشخص أو أكثر**مثال :**

مجموعة من الأولاد بينهم مشاحنات يقوم كل طرف بإرسال رسائل نصية للتهديد أو بريد إلكتروني يتوعدون فيها بعضهم البعض

التعدي من خلال النص وال دردشة (Chatting)

يقصد به: المحادثة عبر الإنترنت باستخدام نصوص وعبارات غير مرغوب منها.

نشر صور غير لائقة

يقصد به: ظهور بعض الصور غير اللائقة لبعض الأشخاص أثناء التجول عبر مواقع الإنترنت.

كيف تحمي نفسك من التعدي الإلكتروني؟

يمكنك حماية نفسك من هذا التعدي بإتباع الإرشادات الآتي:

١- لا تشارك أحد بكلمة السر

٢- تأكد من تكوين كلمة سر يصعب على غيرك تخمينها

٣- عدم نشر أى معلومات خاصة على الإنترنت.

٤- عدم حذف أو التخلص من الرسائل الإلكترونية المرسلة من المتعدي إلكترونياً

٥- عدم مقابلة أحداً قد تعرفت عليه عن طريق الإنترنت

٦- لا تتسرع بإرسال رسالة إلكترونية وانت فى حالة الغضب

٧- اطلاع ولى الامر بما يضايقك عند استخدام الإنترنت

٨- لا تقم بإنزال البرامج من الإنترنت على جهازك من غير استشارة والديك أو أحد المدرسين.

٩- احترم دائماً حق الآخرين على شبكة الإنترنت، وتعلم الخلق الحسن الذي يعتبر فن أدب الحوار على شبكة الإنترنت.

١٠- قم بالإبلاغ عن التعدي الإلكتروني للسلطات المختصة.

مواقف توضح الإرشادات السابقة

١- قال عمرو لياسمين أريد ان ارسل رسالة الى صديق وليس عندي حساب بريد الكتروني هل من الممكن ان تعطيني

اسم المستخدم وكلمة المرور الخاصين بك حتى أتمكن من ارسال هذه الرسالة

فى رأيك ماذا تفعل ؟

الاجابة : لا تشارك أحد بكلمة السر

٢- قام عمرو بإنشاء حساب بريد الكتروني خاص به فحاولت ياسمين استنتاج كلمة المرور بان كتبت اسمه ثم سنه ميلاده ففتح الحساب

فى رأيك ماذا تفعل ؟

الاجابة : تأكد من تكوين كلمة سر يصعب على غيرك تخمينها

٣- قام رامى بالاشتراك فى خدمة ال Facebook وقام بنشر معلومات خاصة بأسرته مستعرضاً صور الأسرة و انشطتهم اليومية

فقام لص بدراسة تلك المعلومات و عرف متى يكون المنزل غير متواجد به احد ثم ... ؟

فى رأيك ماذا كان يجب ان يفعل ؟

الاجابة : عدم نشر أى معلومات خاصة على الإنترنت

٤- اشتكى احد الطلاب لمعلمه ان زميلة سبه فى رسالة بريد الكتروني فقال المعلم ارنى تلك الرسالة فقال الطالب لقد حذفها فقال

المعلم لقد اضعفت الدليل

فى رأيك ماذا كان يجب ان يفعل ؟

الاجابة : عدم حذف أو التخلص من الرسائل الإلكترونية المرسلة من المتعدي إلكترونياً

٥- قال عمرو لوالده لقد تعرفت على شخص فى احدى غرف المحادثة و يود ان يقابلنى

فى رأيك ماذا كان رد والده ؟

الاجابة : عدم مقابلة أحداً قد تعرفت عليه عن طريق الإنترنت

٦- حدثت مشاحنة كبيرة بين طالبين و بعد ان انتهت كان احدهما غاضبا جدا فبعث برسالة تهديد ووعيد للآخر

فى رأيك ماذا كان يجب ان يفعل ؟

الاجابة : لا تتسرع بإرسال رسالة إلكترونية و انت فى حالة الغضب (يجب ان ينتظر حتى يهدأ)

٧- احد الطلاب يستخدم الانترنت فى عمل الأنشطة المدرسية من خلال مجموعات عمل مع زملائه كلما استخدم الانترنت تصلة

رسائل فورية Chat ميسنة له فدافعا عن نفسه قتل من استخدام الانترنت حتى يتجنب الإساءات

فى رأيك ماذا كان يجب ان يفعل ؟

الاجابة : اطلاع ولى الامر بما يضايقك عند استخدام الإنترنت (حتى يساعدك على تغطية تلك المشكلة)

٨- قامت ياسمين بانزال احدى البرامج المجانية على جهاز الكمبيوتر و نتيجة ذلك العمل ظهرت إصابات فيروسية خطيرة للبيانات

المخزنة بالجهاز تقص والدها

فى رأيك ماذا كان يجب ان يفعل ؟

الاجابة : ان تستشير والدها قبل انزال البرامج

اسئلة على التعدي الإلكتروني

السؤال الأول: ضع علامة صح (✓) امام العبارة الصحيحة وعلامة خطأ أمام العبارة الخطأ.

١- التعدي الإلكتروني هو عبارة عن سلوك عدواني متعمد، يستخدم بها المتعدّي الوسائط الإلكترونية للتحرش، والمضايقة وإحراج، وتخويف أو تهديد الآخرين.

()

٢- لا توجد صور من التعدي الإلكتروني داخل المجتمع المصري

()

٣- من أشكال التعدي الإلكتروني التخفي الإلكتروني

()

٤- المتعدّي الإلكتروني يهدف إلى إيقاع ضغوط نفسية على الآخرين باستخدام وسائل متعددة عن طريق وسائط الكترونية ()

()

٥- المضايقات الإلكترونية رسائل عدائيه موجّهة ضد شخص أو أكثر.

()

٦- الملاحقة الإلكترونية إرسال رسالة بطريق الخطأ إلى شخص ما لا تعرفه.

()

٧- التهديد الإلكتروني إرسال رسائل تهديديه المحتوى عن طريق الرسائل الإلكترونية.

()

٨- التشهير الإلكتروني عرض عام أو نشر أو مشاركة للمراسلات الشخصية أو الصور الخاصة لشخص آخر، خاصة المراسلات

التي تحتوي على معلومات شخصية حساسة

()

٩- الاستثناء الإلكتروني ضم بعض الاشخاص إلى مجموعة معينة وتجاهل ونبذ الآخرين.

()

السؤال الثاني: ضع علامة صح (✓) امام العبارة التي تساعدك لتحمي نفسك من التعدي الإلكتروني.

()

١. لا تشارك أحد بكلمة السر.

()

٢. كلمة السر يمكن ان تكون تاريخ ميلاد أو شي سهل يمكن اكتشافه.

()

٣. لا تنشر معلومات خاصة على الإنترنت.

()

٤. تقلص من الرسائل الإلكترونية المرسلة من المتعدّي إلكترونياً.

()

٥. لا ترد على المتعدّي إلكترونياً، ولا تصدق كل ما يكتبوه على الإنترنت.

()

٦. قم بالإبلاغ عن التعدي الإلكتروني للسلطات المختصة

السؤال الثالث

اشترك احد الأشخاص فى احد مواقع التواصل الاجتماعي و كلما راسل احد الاعضاء او حاول اجراء معادنة فورية معة

لاحظ عدم الرد عليه

** يعتبر ما حدث شكلاً من أشكال و يسمى

نموذج اختبار (١)

السؤال الأول : ضع علامة (✓) أمام العبارة الصحيحة وعلامة (x) أمام العبارة الخاطئة:

()	(١) عدد اختيارات التفرع الممكنة مع استخدام جملة If..Then..Else هو ٢.
()	(٢) المعامل & هو احد معاملات المقارنة المنطقية.
()	(٣) إذا كانت قيمة المتغير $X=5$ وقيمة المتغير Y تساوي 7 فإن ناتج التعبير الشرطي $X \geq 5$ and $Y \leq 7$ هو False.
()	(٤) استدعاء الاجراءات (Procedures) يكون لمرة واحدة فقط .
()	(٥) تستخدم Dim للإعلان عن المتغيرات (Variables) .

السؤال الثاني اختر الإجابة الصحيحة:

١- في حالة تحديد القيمة ٣٥٠٠ للخاصة (Interval) للأداة (Timer) فهذا يعني:

أ- ٣,٥ دقيقة ب- ٣,٥ ثانية ج- 35 ثانية

٢- أي من جمل Select التالية ناتجها True في حالة قيمة متغير الحلقة يساوي ٨:

أ- Case Is < 8 ب- Case 1 To 6, 8 ج- Case Is >= 4

٣- عند معرفة عدد مرات التكرار لكود (Code) معين مسبقاً فإن أفضل جملة تكرر هي:

أ- Do While...Loop ب- If... End If ج- For...Next

٤- ما القيمة المعروضة في أداة العنوان lblOut عند تنفيذ الكود (Code) التالي:

```
Dim i as Integer
For i = 2 To 10 Step 2
    lblOut.Text = lblOut.Text & i & ", "
Next
```

أ- ٢ ب- 2,3,4,5,6,7,8,9,10 ج- 2,4,6,8,10

٥- يتم تخصيص قيم للثوابت (Constants):

أ- عند الإعلان عن الثوابت فقط.

ب- أثناء سير تنفيذ تعليمات البرنامج فقط.

ج- عند الإعلان عن الثوابت وكذلك أثناء سير تنفيذ تعليمات البرنامج.

السؤال الثالث أجب عما يلي

١- حدد قيمة كل متغير بعد تنفيذ الكود (Code) التالي:

```
Dim N, K, C As Integer
C = 10
For N = 1 To 10 Step 5
    K = K + N
Next
If N >= C Then
    C = C + 1
End If
```

المتغير	القيمة
C	
K	
N	

٢- رتب السطور الآتية وفقا لأولويات تنفيذ العمليات الحسابية

العملية	الترتبة
الضرب والقسمة من اليسار إلى اليمين.	()
الأقواس من الداخل إلى الخارج.	()
الجمع والطرح من اليسار إلى اليمين.	()
الأسس.	()

٣- في الكود (Code) الآتي:

```
Dim Total As Decimal
Dim FirstNumber As Decimal
Dim SecondNumber As Decimal
```



١

```
FirstNumber = 10
SecondNumber = 20
Total = FirstNumber + SecondNumber
```



٢

فسر الكود (Code) المشار إليه بالأرقام على الشكل

-١

-٢

نموذج اختبار (٢)

السؤال الأول : ضع علامة (✓) أمام العبارة الصحيحة وعلامة (x) أمام العبارة الخطأ:

()	(١) تستخدم جملة If..Then..ElseIf في حالة اختبار أكثر من تعبير شرطي.
()	(٢) المتغيرات من أنواع (Integer & Long & Double) تستخدم لتخزين الأعداد الصحيحة فقط.
()	(٣) باستخدام معامل المقارنة Or يجب أن يكون كافة نواتج الشروط المستخدمة False لكي يكون النتائج النهائي False.
()	(٤) تستخدم جملة Select Case في حالة اختبار أكثر من تعبير شرطي بجملة شرطية واحدة.
()	(٥) المتغير من النوع Double يأخذ القيمة True أو False.

السؤال الثاني اختر الإجابة الصحيحة:

١- بفرض وجود دالة باسم AddSum() أي من التعبيرات الآتية غير صحيح عند استدعائها:

أ- $X=Y+AddSum-$ ب- $AddSum()=X+Y-$ ج- $X=AddSum()-$ ٢- لإنهاء حلقة تكرارية عند تحقق الشرط $A=2$ تكتب بدايتها بالشكل التالي:أ- $Do While (a = 2)-$ ب- $Do While (a < 2) Or (a >= 2)-$ ج- $Do While (a < > 2)-$

٣- حدد الخطأ في الحلقة التكرارية التالية:

Dim J As Byte

For J=1 to 500 Step 3

ListBox1.Items.Add(J)

Loop

أ- يجب استخدام جملة Do...While بدلا من For...Next

ب- الحلقة غير منتهية

ج- خطأ في نوع متغير الحلقة

٤- يتوقف التكرار في جملة For m=10 to 1 Step -2 عندما تصبح قيمة المتغير M:

أ- أكبر من قيمة النهاية

ب- أصغر من قيمة النهاية

ج- أصغر من أو تساوى قيمة النهاية

٥- حدد ناتج تنفيذ العملية الحسابية $2 - (7+2) * 2$

أ- ١٦ ب- ١٤ ج- ٥

السؤال الثالث أجب عما يلي

١- أعد كتابة الكود التالي باستخدام جملة Select...Case

Select Case a	<pre> Dim x As Integer x = Me.textbox1.text If x = 0 Then label1.text = "القيمة تساوى صفر" End If If x > 0 Then Label1.Text = "القيمة أكبر من الصفر" End If If x < 0 Then Label1.Text = "القيمة أصغر من الصفر" End If </pre>
End Select	

٢- اكتب أمام كل عبارة بالعمود (ب) الرقم المناسب من العمود (أ)

م	(أ)	م	(ب)
١-	الإجراء (Sub)		تستخدم في الطرف الأيمن من جملة التخصيص ولا يخصص لها أي قيمة.
٢-	الدالة (Function)		تستخدم في الطرف الأيمن من جملة التخصيص ويخصص لها قيمة عند الإعلان فقط.
٣-	المتغيرات (Variables)		لا تستخدم في جملة التخصيص نهائياً.
٤-	الثوابت (Constants)		تستخدم في طرفي جملة التخصيص.

٣- حدد نوع الخطأ في الكود (Code) الآتي ثم صوبه

الكود (Code)	نوع الخطأ	التصويب
Dim x As Sangle		
Const x As Integer X=10		

نموذج اختبار (٣)

السؤال الأول : ضع علامة (✓) أمام العبارة الصحيحة وعلامة (x) أمام العبارة الخاطئة:

()	(١) الخاصية SelectedIndex تشير الى العنصر المحدد بالأداة ListBox.
()	(٢) تستخدم Select..Case في حالة وجود أكثر من احتمالين للتفرع.
()	(٣) الجملة If A > B And B > C Then True ناتجها True في حالة A=3, B=5, C=2.
()	(٤) ناتج الدالة IsNumeric دائما يكون True أو False.
()	(٥) المتغيرات من أنواع (Integer & Long) تستخدم لتخزين الأعداد الصحيحة فقط.

السؤال الثاني اختر الإجابة الصحيحة:

أولا- اختر الإجابة الصحيحة:

١- الهدف من الكود (Code) التالي هو:

```
Dim N As Integer
Dim S As Integer
For N = 1 to 100
    S=S +N
Next
Label1.Text = S
```

أ- عرض الأعداد من ١ الى ١٠٠

ب- عرض مجموع الأعداد من ١ الى ١٠٠

ج- عرض حاصل ضرب الأعداد من ١ الى ١٠٠

٢- ما القيمة المعروضة في اداة العنوان (Label1) عند تنفيذ الكود التالي:

```
Dim intNum as Integer = 10
intNum= intNum+٢
intNum = intNum Mod 3
Label1.Text = intNum
```

أ-3

ب-0

ج-4

٣- لإنهاء حلقة تكرارية عند تحقق الشرط $A < > 2$ تكتب بدايتها بالشكل التالي:

أ- Do While (a = 2)

ب- Do While (a < 2) Or (a >= 2)

ج- Do While (a < > 2)

ثانياً - أي من الآتي إجراء (Sub) أو دالة (Function):

أ-

```
Function Concatenating(ByVal fn As String, ByVal ln As String) As String
    Dim FullName As String
    FullName = fn + " " + ln
    Return FullName
End Function
```

ب-

```
Sub ShowResult()
    MessageBox.Show(2 * (7 + 2) - 2)
End Sub
```

السؤال الثالث أجب عما يلي:

١ - حدد قيمة كل متغير بعد تنفيذ الكود (Code) التالي:

```
IF x < y Then
    z = K + J
    x = x + 1
    y = y - 1
Else
    z = K - y
    x = x + 2
    y = y - 2
End IF
```

المتغير	القيمة
x	
y	
z	

٢ - استخرج من الكود (Code) الآتي:

```
Function calc(ByVal fn As Single, ByVal sn As Single) As Single
    Dim res As Single
    res = fn + sn
    Return res
End Function
```

١ - اسم الإجراء :

٢ - وسائط هذا الإجراء:

٣ - القيمة الراجعة:

٣ - نوع البيان الخاص ب:

أ - الدالة:

ب - وسيط الدالة:

ج - القيمة الراجعة: