

## ملخص عام لمنهج الكمبيوتر

### Problem Solving (حل المشكلات) (الفصل الأول)

**المشكلة Problem** : موقف يتطلب إيجاد حل له ، أى هدف مطلوب الوصول إليه من خلال إتباع عدة خطوات بترتيب محدد .

**حل المشكلة Problem Solving** : عبارة عن الخطوات والأنشطة والعمليات التى ينبغى القيام بها للوصول إلى الهدف أو الناتج .

### مراحل حل المشكلة Problem Solving Stages

#### (١) تحديد المشكلة ( Problem Definition ) :

أى تحديد المخرجات المطلوبة والمدخلات المتوفرة وعمليات المعالجة الحسابية أو المنطقية .

#### (٢) إعداد خطوات الحل الخوارزمية ( Algorithm ) :

هى أحد الأساليب المستخدمة فى حل المشكلة من خلال مجموعة من الإجراءات المرتبة منطقياً .

**الخوارزمية** : هى مجموعة من الإجراءات ( الخطوات ) المرتبة ترتيباً منطقياً للوصول إلى الهدف أو الناتج المطلوب ويتم تمثيلها فى شكل خريطة التدفق .

#### (٣) تصميم البرنامج على الكمبيوتر ( Program Design ) :

يتم ترجمة خريطة التدفق إلى برنامج باستخدام إحدى لغات البرمجة .

#### (٤) اختبار صحة البرنامج وتصحيح الأخطاء ( Program Testing ) :

ويتم ذلك عن طريق إدخال بيانات معروف نتائجها مسبقاً حتى نتأكد من مقارنة النتائج التى نحصل عليها بالنتائج الفعلية وبذلك يمكن أن نكتشف الأخطاء ونقوم بتصحيحها .

#### (٥) توثيق البرنامج ( Program Documentation ) :

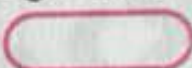
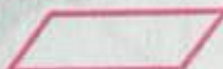
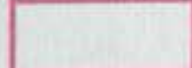


هى مرحلة يتم فيها كتابة كل الخطوات التى اتخذت لحل المشكلة من مدخلات ومخرجات وخطة حل وخريطة التدفق المستخدمة واللغة التى كتب بها البرنامج وأوامر البرنامج وتاريخ آخر تعديل فى البرنامج ومن شارك فى عمل البرنامج للاحتفاظ به موثقاً ( مكتوب ) للرجوع إليه مرة أخرى .



## خرائط التدفق Flowcharts

هى تمثيل تخطيطى يعتمد على الرسم بأشكال قياسيه لتوضيح ترتيب العمليات اللازمة لحل مسألة أو مشكلة محددة .

**بعض الرموز القياسية الشائعة التى تستخدم فى رسم خرائط التدفق :**

الرمز	الوظيفة
شكل بيضاوى 	الرمز الطرفى ( Terminal ) يعبر عن البداية والنهاية .
متوازى أضلاع 	يعبر عن الإدخال والإخراج ( Input / Output ) .
مستطيل 	يعبر عن معالجة أو عملية ( Process ) .
المعين 	يعبر عن اتخاذ القرار ( Decision ) .
خطوط الاتجاه 	تعبر عن اتجاه خط سير ( تدفق ) العمليات .

- لا يمكن استخدام أى شكل هندسى عند رسم خرائط التدفق .
- مدلول كل رمز من رموز خرائط التدفق ثابت لا يمكن أن يتغير .

## مميزات خرائط التدفق

- ١- تيسر قراءة وفهم المشكلة وتوضح للمبرمج ما يجب عمله .
- ٢- مفيدة فى شرح البرنامج للآخرين .
- ٣- تساعد خريطة التدفق فى توثيق أفضل للبرنامج وخصوصا إذا كان البرنامج

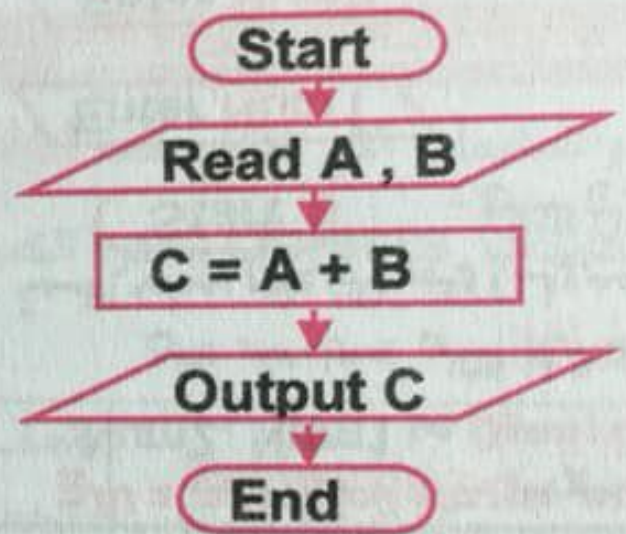
معقدًا .



- ١- خرائط التدفق البسيطة ( Simple Flowcharts )
- ٢- خرائط تدفق تعتمد على التفرع ( اتخاذ قرار Decision )
- ٣- خرائط تدفق تعتمد على الحلقات التكرارية ( Loops )

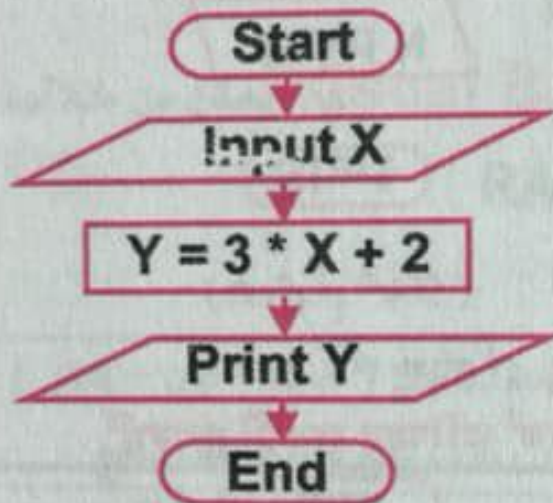
## خرائط التدفق البسيطة Simple Flowcharts

١- ارسم خريطة تدفق لإيجاد مجموع عددين يتم إدخالهما وإظهار الناتج :

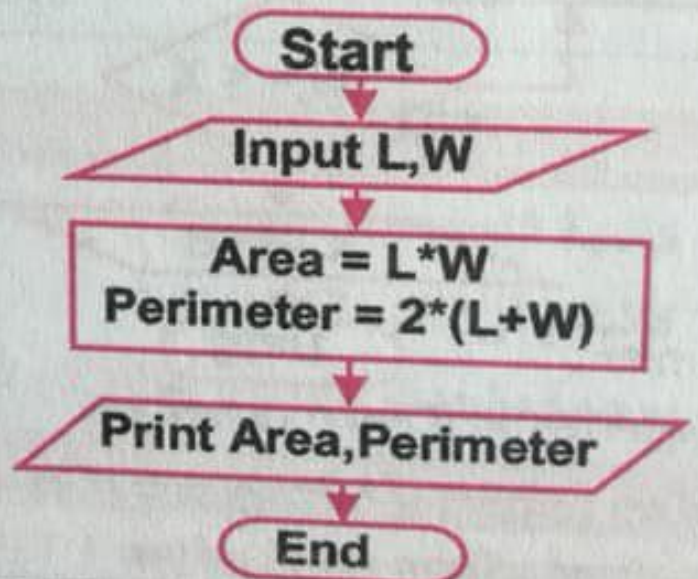


٢- ارسم خريطة تدفق لإيجاد ناتج معادلة من الدرجة الأولى :

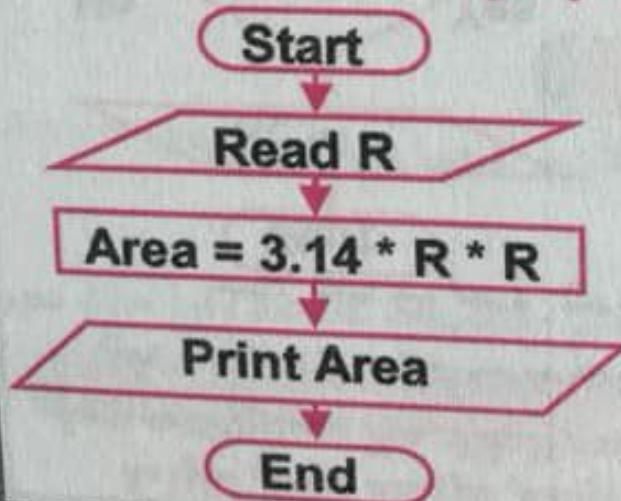
باستخدام المعادلة  $Y = 3 * X + 2$



٣- ارسم خريطة تدفق لحساب مساحة ومحيط مستطيل بمعلومية الطول L والعرض W



٤- ارسم خريطة تدفق لحساب مساحة دائرة بمعلومية نصف القطر ( R ) مع العلم أن المعادلة الخاصة بمساحة الدائرة هو  $Area = 3.14 * R * R$

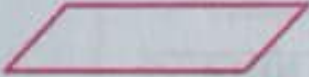





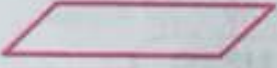
- يجب أن تبدأ خريطة التدفق برمز البداية **Start** وتنتهي برمز النهاية **End**

- المعادلة  $C = A + B$  تعنى جمع قيمة المتغير **A** وقيمة المتغير **B** ووضع الناتج فى المتغير **C** .

- يطلق على كل من **A** و **B** و **C** اسم متغير **Variable** ويعنى مخزن بالذاكرة يحتوى على قيمة .

- يتم التعبير عن إدخال قيم **A** و **B** باستخدام لفظ " Enter " داخل شكل متوازي الأضلاع  ويمكن استخدام أي لفظ آخر يؤدي نفس المعنى مثل **Read** أو **Get** أو **Input** أو أدخل .

- يتم وضع أى معادلة داخل مستطيل  حيث إنها تمثل عملية حسابية .

- يتم التعبير عن المخرج وهو قيمة المتغير **C** داخل الشكل  بكلمة **Output** ويمكن استخدام أي لفظ آخر يؤدي نفس المعنى مثل **Print** أو اطبع أو اخرج .

- لاحظ خط الاتجاه ↓ يوضح ترتيب تدفق خطوات الحل .

- رمز اتخاذ القرار فى خرائط التدفق يدخل به خط اتجاه واحد ويخرج منه خطين اتجاه على الأقل.

- يفضل أن يدل اسم المتغير على محتواه.

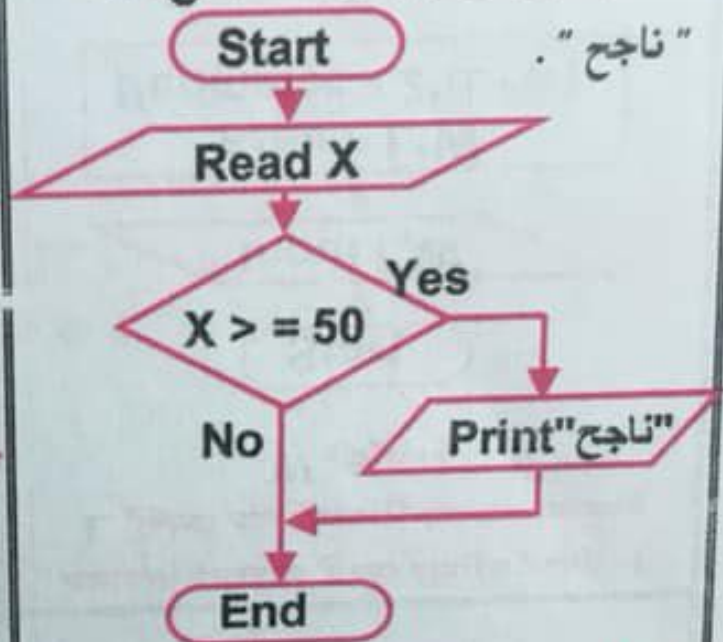
- يحتوى الطرف الأيسر لأى معادلة على متغير واحد وهو ناتج ( مخرج ) المعادلة ، ويحتوى الطرف الأيمن على قيم ( مجردة أو تعبير حسابى )

و قد يحتوى على متغير أو أكثر ( مدخلات ) .

## خرائط تدفق تعتمد على التفرع ( اتخاذ القرار Decision )

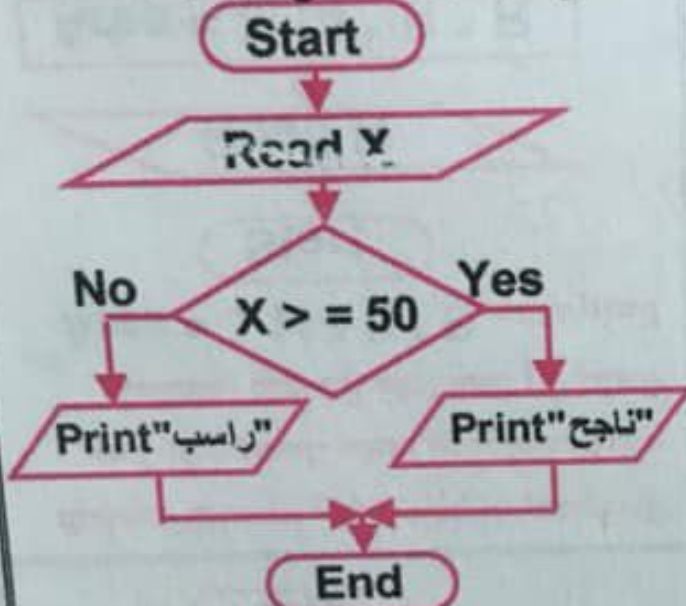
### ١- ارسم خريطة تدفق لقراءة

درجة الطالب  $X$  بحيث إذا كانت درجة الطالب أكبر من أو يساوي ٥٠ يطبع كلمة "ناجح".



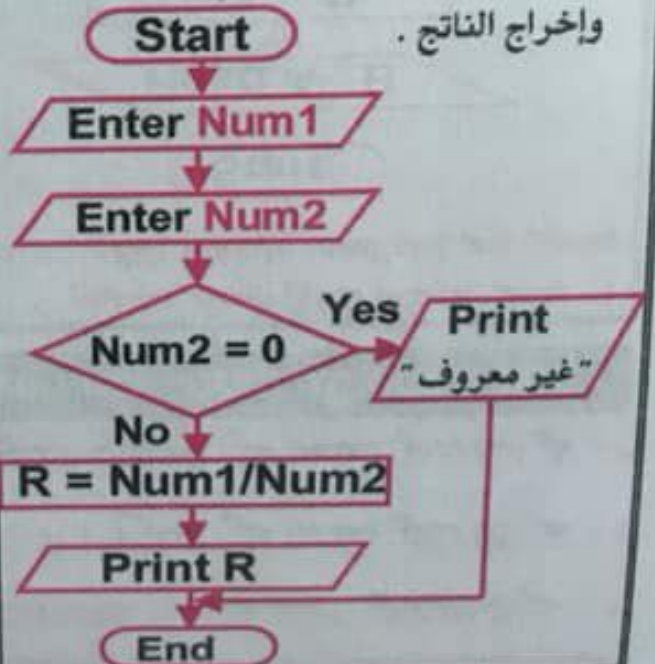
### ٢- ارسم خريطة تدفق لقراءة

درجة الطالب  $X$  بحيث إذا كانت درجة الطالب المدخلة أكبر من أو يساوي ٥٠ يطبع "ناجح" و غير ذلك يطبع "راسب".



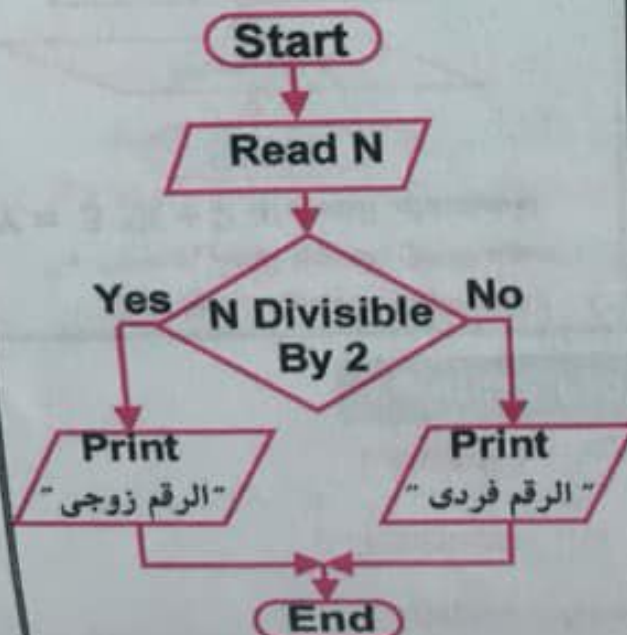
### ٣- ارسم خريطة تدفق لإيجاد ناتج

قسمة رقمين  $Num1$ ،  $Num2$  وإذا كان الرقم الثاني يساوي صفر يطبع (غير معروف) وغير ذلك يقوم بعملية القسمة وإخراج الناتج.



### ٤- ارسم خريطة تدفق لإخال

عدد  $N$  ثم طباعة نوع العدد (زوجي أم فردي).

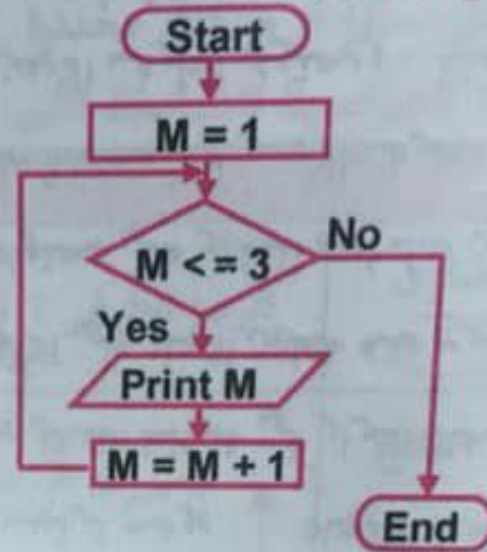


الامتحان التعليمي  
[www.exam-eg.com](http://www.exam-eg.com)



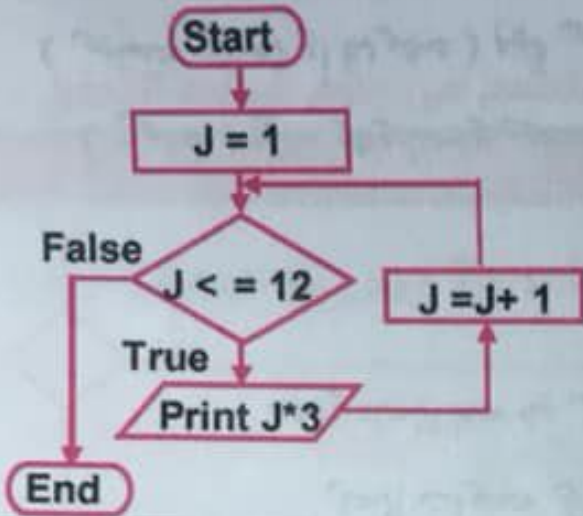
# خرائط تدفق تعتمد على الحلقات التكرارية Loops

١- ارسم خريطة تدفق لعرض الأعداد من 1 : 3 .



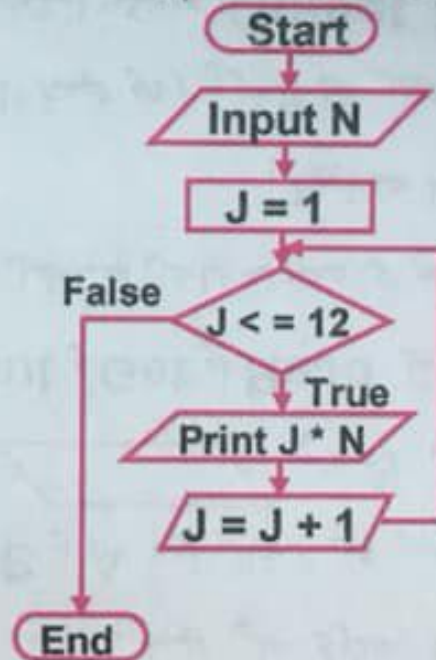
٢- ارسم خريطة تدفق

لطباعة جدول ضرب العدد 3 .



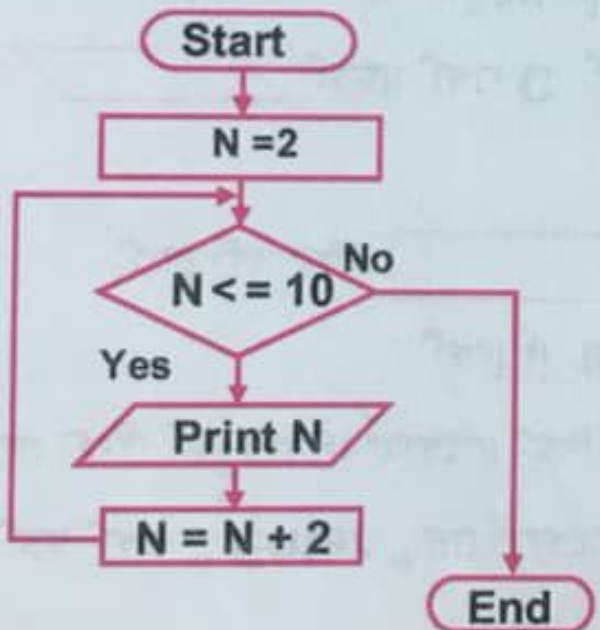
٣- ارسم خريطة تدفق

لطباعة جدول الضرب لعدد يتم إدخاله N



٤- ارسم خريطة تدفق لطباعة

الأعداد الزوجية من ( 1 : 10 ) .



- العبرة ليست بشكل خريطة التدفق ولكن بترتيب خطوات الحل .

- يطلق على المتغير في ( التكرار ) ب Counter أى عداد حيث يتسبب في تكرار

خطوات معينة لعدد محدد من المرات .

- فى النوع الثالث من خرائط التدفق ( التكرار ) يمكن حساب عدد مرات التكرار

وأخر قيمة للعداد .

## لغة فيجوال بيزيك دوت نت Visual Basic.Net :

هي إحدى لغات البرمجة عالية المستوى High Level Language ومصممة لتكون سهلة التعلم وأوامرها وتعليماتها مستمدة من مفردات اللغة الإنجليزية .  
 تُستخدم لغة **VB.Net** لإنتاج تطبيقات ذات الواجهة الرسومية ( GUI ) منها :

١- تطبيقات مكتبية ( Windows Applications ) .

٢- تطبيقات ويب ( Web Applications ) .

## البرمجة وذاكرة الكمبيوتر :

الأوامر والتعليمات التي تكتب بلغة **Visual Basic.Net** يمكن من خلالها إنشاء الكائنات **Objects** بذاكرة الكمبيوتر **RAM** .  
وكل كائن له :

١- خصائص <b>Properties</b>	تصف الكائن وتحدده مثل ( حجم - لون - شكل ) الخط للنص الذي يكتب على واجهة البرنامج .
٢- أحداث <b>Events</b>	تقع على الكائن مثل حدث النقر <b>Click</b> على زر الأمر .
٣- إجراءات <b>Procedures</b>	يحتوي كل كائن على أوامر وتعليمات ، تنفذ تلك الأوامر والتعليمات عندما يستدعى هذا الإجراء .

## تعتبر لغة Visual Basic.Net :

١- كائنية التوجه **Object Oriented** : لأن برامجها تعمل من خلال كائنات في ذاكرة الكمبيوتر .

٢- موجهة بالحدث **Event Driven** : لأن الأوامر والتعليمات تنفذ عند وقوع حدث معين .

## لغة Visual Basic.Net وإطار العمل .NET Framework :

### يوفر إطار العمل .NET Framework الآتي :

١- المكتبات : التي يتم منها إنشاء الكائنات .



٢- بيئة تشغيل : تسمى Runtime بذاكرة الكمبيوتر ، وتعمل فيها التطبيقات المنتجة بلغة **Visual Basic.Net**

٣- المترجمات **Compilers** : التي تترجم الأوامر والتعليمات المكتوبة بلغة البرمجة للغة الآلة التي يتعامل معها الكمبيوتر .

## لغة Visual Basic.Net وشاشة IDE :

يحتاج المبرمج إلى بيئة تطوير متكاملة يطلق عليها ( **IDE** ) وهي اختصار لـ **Integrated Development Environment** - وهي توفر للمبرمج كل ما يحتاجه من أدوات وميزات تمكنه من إنشاء تطبيقات ( ويندوز - موبايل - ويب ... إلخ ) والذي يمثلها ما يطلق عليه ( **Visual Studio** ) .

## مكونات شاشة IDE :

(١) النموذج **Form** : هو النافذة التي يصمم عليها واجهة البرنامج التي يتعامل معها المستخدم من خلال وضع أدوات التحكم ( **Controls** ) المختلفة عليها .

- عند إنشاء مشروع جديد يتم تلقائياً إنشاء نموذج جديد باسم **Form1** .

## (٢) صندوق الأدوات **Toolbox** :

- يحتوي على أدوات تحكم ويتم تصنيفها وعرضها في صورة فئات .

- ومن هذه الفئات ( **Common Controls** ) : أدوات التحكم الشائعة .

، ( **Menus & Toolbars** ) : القوائم وأشرطة الأدوات .

- ويمكن عرض كافة أدوات التحكم باختيار الفئة ( **All Windows Forms** )

## (٣) نافذة الخصائص **Properties Window** :

- تُستخدم لتحديد وضبط قيم خصائص أداة التحكم النشطة بنافذة النموذج .

- الخصائص توجد في العمود الأيسر وقيم الخصائص توجد في العمود الأيمن .

- تختلف الخصائص المعروضة في نافذة الخصائص حسب الجزء النشط في شاشة

الـ ( **IDE** ) .

## (٤) مستعرض الحل **Solution Explorer** :

نافذة يعرض بها قائمة بمجلدات وملفات المشروع أو المشروعات الموجودة ضمن الحل .



١ الخصائص المميزة للنموذج Form :

الخاصية	الوظيفة	أثر ضبط الخاصية يظهر في نمط التصميم
Name	تحدد اسم النموذج الذي سيستخدم في نافذة الكود .	التصميم
Text	تحدد النص الظاهر على شريط عنوان نافذة النموذج .	التصميم والتشغيل
BackColor	تحدد لون خلفية النموذج .	التصميم والتشغيل
FormBorderStyle	تحدد نمط حدود النموذج .	التصميم والتشغيل
 ControlBox	تتحكم في إظهار أو إخفاء صندوق تحكم نافذة النموذج .	التصميم والتشغيل
 MinimizeBox	تتحكم في إظهار أو إخفاء صندوق التصغير .	التصميم والتشغيل
 MaximizeBox	تتحكم في إظهار أو إخفاء صندوق التكبير .	التصميم والتشغيل
RightToLeft	تحدد اتجاه أدوات التحكم على نافذة النموذج من اليمين إلى اليسار - وقيمتها ( No / Yes ) .	التصميم والتشغيل
RightToLeftLayout	تحدد اتجاه تخطيط أدوات التحكم على نافذة النموذج من اليمين إلى اليسار - وقيمتها ( True / False ) .	التصميم والتشغيل
WindowState	تحدد حالة نافذة النموذج على الشاشة ( عادي / تكبير / تصغير ) .	التشغيل



- هناك خصائص مشتركة بين أدوات التحكم المختلفة مثل :

( Text – Name – ForeColor – BackColor – RightToLeft )

- هناك خصائص لا يظهر أثرها على أدوات التحكم إلا بعد ضبط خصائص أخرى

مثل **RightToLeftLayout** لدى النموذج **Form** لا تعمل إلا إذا

كانت قيمة الخاصية **RightToLeft** تساوى **Yes**.

- هناك خصائص إذا تم ضبطها لنافذة النموذج تطبق على أدوات التحكم التي يتم

وضعها على النموذج مثل خاصية **Font** و **ForeColor**.

- هناك خصائص عند ضبطها يظهر أثر ضبطها على النموذج فوراً في وضع تصميم البرنامج.

- هناك خصائص لا يظهر أثر ضبطها على النموذج أو أدوات التحكم إلا في وضع

تشغيل البرنامج مثل خاصية **WindowState**.

- في نافذة الخصائص العمود الأيسر يوجد به الخصائص والعمود الأيمن يوجد به

قيم الخصائص.

## ٢ زر الأمر **Button** :

- هو أحد أدوات التحكم التي يمكن إدراجها على نافذة النموذج والذي من خلاله

يستطيع مستخدم البرنامج النقر عليه لتنفيذ مهمة معينة.

- يستخدم مع أداة الزر **Button** ثمان مربعات تحجيم لتغيير أبعاد الأداة.

- يمكن تغيير موضع زر الأمر **Button** باستخدام عملية السحب والإفلات عند

الوقوف على زر الأمر بمؤشر الفأرة وتغيير شكل المؤشر لشكل رباعي الرؤوس.



### من أشهر خصائص زر الأمر Button :

الخاصية	الوظيفة	أثر ضبط الخاصية يظهر في نمط
Name	تحدد اسم أداة زر الأمر الذي سيستخدم في الكود .	التصميم
Text	تحدد النص الظاهر على أداة زر الأمر .	التصميم والتشغيل
BackColor	تحدد لون خلفية أداة زر الأمر .	التصميم والتشغيل
ForeColor	تحدد اللون الأمامي للنص الظاهر على أداة زر الأمر .	التصميم والتشغيل
Font	تحدد ( شكل وحجم ونمط ) خط النص الظاهر على زر الأمر .	التصميم والتشغيل
Size	تحدد ارتفاع وعرض أداة زر الأمر .	التصميم والتشغيل
Location	تحدد موقع إدراج زر الأمر على نافذة النموذج .	التصميم والتشغيل

### ٣ أداة العنوان Label :

– أداة تُستخدم في عرض نص على نافذة النموذج لا يمكن تغييره أثناء تشغيل البرنامج.

### ومن أشهر خصائصها :

الخاصية	الوظيفة	أثر ضبط الخاصية يظهر في نمط
AutoSize	تحدد إذا كان حجم أداة العنوان Label يتحدد تلقائياً حسب النص المكتوب عليه أم لا وقيمتها ( True أو False ) .	التصميم والتشغيل
BorderStyle	اختيار نمط إطار ( حدود ) أداة العنوان .	التصميم والتشغيل

– يمكن تغيير حجم أداة العنوان يدوياً عن طريق السحب والإفلات عندما تكون القيمة

False للخاصية AutoSize من خلال المقابض الثمانية في وضع التصميم فقط .



#### ٤ أداة صندوق الكتابة TextBox :

- أداة لإدخال بيانات نصية من مُستخدم البرنامج أثناء تشغيل البرنامج .
- يمكن تعديل محتواها أثناء التشغيل .

◀ من أشهر خصائصه :

الخاصية	الوظيفة	أثر ضبط الخاصية يظهر في نمط
MaxLength	تحدد الحد الأقصى لعدد الحروف التي يتم كتابتها داخل الصندوق .	التشفيل
PasswordChar	تحديد رمز يستخدم في الظهور بدلاً من النص المكتوب في حالة إذا ما أردنا عمل كلمة مرور مثلاً .	التصميم والتشفيل
Multiline	تتيح إمكانية تعدد الأسطر داخل الأداة وتأخذ هذه الخاصية القيمة ( True أو False ) .	التصميم والتشفيل

#### ٥ صندوق القائمة ListBox :

تستخدم في عرض قائمة من العناصر .

◀ من أشهر خصائصه :

الخاصية	الوظيفة	أثر ضبط الخاصية يظهر في نمط
Items	عبارة عن مجموعة العناصر التي تعرض على صندوق القائمة .	التصميم والتشفيل
Sorted	تحدد إذا ما كانت العناصر داخل القائمة مرتبة أبجدياً أم لا .	التصميم والتشفيل
SelectionMode	تحدد إذا ما كان من الممكن اختيار عنصر واحد أو أكثر من العناصر المعروضة على صندوق القائمة .	التشفيل

#### ٦ صندوق التحرير والسرد ComboBox :

عبارة عن صندوق به قائمة عناصر تنسدل لاختيار إحداها .



« ومن أشهر خصائصها :

الخاصية	الوظيفة	أثر ضبط الخاصية يظهر في نمط
Items	عبارة عن مجموعة العناصر التي تعرض على صندوق التحرير والسرد .	التشغيل
AutoCompleteSource	عبارة عن مصدر العناصر المقترحة لعملية الإكمال .	التشغيل
AutoCompleteMode	يحدد الطريقة التي سوف يتم بها عملية إكمال القائمة .	التشغيل

## ٧ أداة صندوق المجموعة GroupBox :

- تستخدم في احتواء أدوات التحكم ذات الوظيفة الواحدة على نافذة النموذج .
- يستخدم في تنظيم وضع الأدوات على نافذة النموذج .

## ٨ زر اختيار بديل واحد RadioButton :

- تتيح للمستخدم اختيار بديل واحد فقط من بين عدة بدائل . ومن أشهر خصائصها :

الخاصية	الوظيفة
Checked	توضح إذا ما كان زر اختيار بديل واحد تم اختياره أم لا .
Text	النص الظاهر على أداة زر اختيار بديل واحد .

## ٩ أداة صندوق الاختيار CheckBox :

- يستخدم في إدراج عدة بدائل بحيث يمكن للمستخدم اختيار بديل أو أكثر من البدائل المعروضة ومن أشهر خصائصها :

الخاصية	الوظيفة
Checked	توضح إذا ما كان زر الاختيار تم اختياره أم لا
Text	النص الظاهر على الأداة .



## الفصل الرابع ( نافذة الكود Code Window )

**نافذة الكود:** هي النافذة التي يمكن من خلالها كتابة الأوامر والتعليمات الخاصة بلغة **Visual Basic.Net**

### فتح نافذة الكود بأربع طرق :

- (١) اضغط على مفتاح **F7** من لوحة المفاتيح .
- (٢) انقر على الأداة مرتين متتاليتين
- (٣) اختر أمر **Code** من قائمة **View** .
- (٤) من نافذة الحل اضغط بالزر الأيمن على **Form1.VB** واختار **View Code**

### تحتوى نافذة الكود على :

١- **قائمة Class Name** تعرض أسماء أدوات التحكم المدرجة على النموذج .

٢- **قائمة Method Name**

تعرض الأحداث الخاصة بالأداة المختارة من قائمة **Class Name** .

### معالج الحدث Event Handler :

عبارة عن إجراء يحتوى على كود يتم تنفيذه عندما يقع الحدث المرتبط به .

◀ يتكون اسم معالج الحدث من ( اسم أداة التحكم واسم الحدث ) .

- **End Class** تعبر عن نهاية التصنيف و **End Sub** تعبر عن نهاية الإجراء .

- تشغيل البرنامج بالضغط على **F5** من لوحة المفاتيح .

### ◀ صيغة ضبط الخصائص Properties برمجيا يكون كالتالى :

**ControlName.Property=Value**

القيمة = الخاصية . اسم الكائن أو ( أداة التحكم )

### أكواد هامة :

١- كود ضبط الخاصية **Text** لأداة العنوان **Label1** إلى "جمهورية مصر العربية" .

**Label1.Text = "جمهورية مصر العربية"**

٢- كود يجعل أداة العنوان **Label1** يتحدد حجمها طبقا للنص المكتوب بها تلقائيا .

**Label1.AutoSize = True**

مدرسة اون لاين