



الوحدة السادسة

البرمجة بلغة (فيجول بيسك ستوديو)

في هذه الوحدة سنتعلم :

- مراحل كتابة البرنامج بلغة (فيجول بيسك ستوديو).
- طريقة تعامل البرنامج مع البيانات.
- العمليات الحسابية والمنطقية.
- أدوات البرمجة بلغة (فيجول بيسك ستوديو).
- بعض الأوامر الأساسية للغة (فيجول بيسك ستوديو).



هناك عدة لغات برمجية لإنشاء برامج خاصة بالحاسب كما تعلمت سابقاً، وسوف ندرس في هذا الوحدة إحدى اللغات العالية (High-Level Language)، وبالتحديد إحدى لغات البرمجة بالعناصر أو البرمجة المرئية وهي لغة (فيجول بيسك ستوديو) (Visual Basic Studio).

وتعد البرمجة باستخدام (فيجول بيسك ستوديو) شيقة وممتعة، وذلك لما تمتاز به من تحكم المبرمج في البرامج التي يقوم بتصميمها من ناحية : واجهات الإدخال للمستخدم، والعمليات الإجرائية للبرنامج، وأخيراً المخرجات التي يحصل عليها المستخدم لهذا البرنامج.

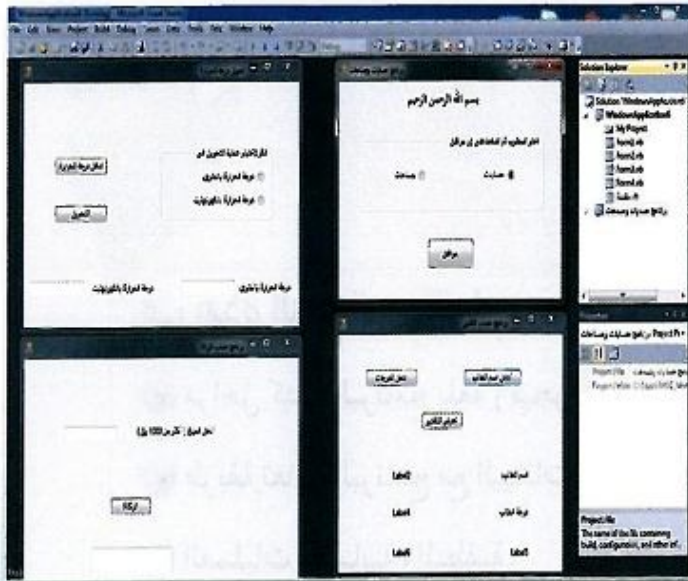
مراحل كتابة البرنامج بلغة (فيجول بيسك ستوديو):

تعلمنا في الوحدة السابقة خطوات المرحلة الأولى من حل المسألة وهي : فهم المسألة وتحديد عناصرها، وكتابة الخوارزم والخطوات المنطقية للحل، والتمثيل البياني للخوارزم عن طريق مخططات الانسياب. وفي هذه الوحدة سنتعلم المرحلة الثانية وهي مرحلة كتابة البرنامج باستخدام لغة (فيجول بيسك ستوديو)، والتي تتكون من ثلاث خطوات:

١- تصميم الواجهات.

٢- ضبط خصائص الأدوات.

٣- كتابة أوامر البرمجة.



أولاً / تصميم الواجهات :

وهنا نبدأ تصميم الواجهات التي سوف تظهر للمستخدم، من: تحديد عدد النوافذ التي يحتاجها البرنامج، والأدوات التي نحتاجها على كل نافذة، كالأزرار ومربعات النصوص والقوائم، وغيرها كما في الشكل (١-٦).

شكل (١-٦) : نماذج لواجهات برامج



الوحدة السادسة: البرمجة بلغة (فيجول بيسك ستوديو)

ثانياً / ضبط خصائص الأدوات :

بعد أن نضع الأدوات على النافذة، تأتي مرحلة تحديد خصائص هذه الأدوات، حيث توجد لكل أداة من الأدوات عدة خصائص (Properties) كشكلها، ولونها، والخط المستخدم فيها، وعنوانها، وغير ذلك. وهذه الخصائص افتراضية، لذا نقوم بتغيير خصائص هذه الأدوات لتناسب البرنامج، كما في الشكل (٢-٦).



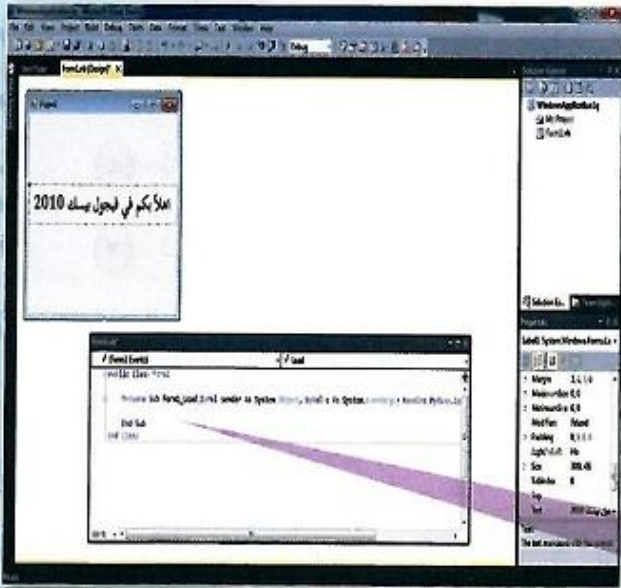
نافذة خصائص الأدوات وفيها نحدد خصائص كل أداة .

مثلاً لتغيير خاصية عنوان أداة تجد في نافذة الخصائص خاصية (Text) وهكذا لبقية الخصائص.

شكل (٢-٦) : نافذة البرنامج وخصائص الأدوات

ثالثاً / كتابة أوامر البرمجة :

بعد أن تنتهي من المرحلتين السابقتين، تأتي مرحلة كتابة الأوامر التي نريد من (فيجول بيسك ستوديو) أن ينفذها عند وقوع حدث معين، فمثلاً عندما يضغط المستخدم على زر الأوامر ففي هذه الحالة يكون الحدث، وهنا نكتب الأوامر التي نريد من البرنامج أن ينفذها، كما في الشكل (٣-٦).



شاشة كتابة الأوامر حيث يكون الاعلان عن المتغيرات التي تحتاجها وغيرها من الأوامر.

شكل (٣-٦) : شاشة كتابة أوامر البرمجة

الهدف الرئيس من أي برنامج هو معالجة البيانات. وتختلف أنواع البيانات، فقد تكون حروفاً أو أرقاماً أو تواريخ أو غيرها. وتأتي البيانات غالباً من مستخدم البرنامج؛ حيث يدخلها عن طريق أجهزة الإدخال المتصلة بجهاز الحاسب الآلي كلوحة المفاتيح أو الفأرة مثلاً.

يستقبل البرنامج البيانات ويخزنها في الذاكرة الرئيسة حتى يستطيع استخدامها والتعامل معها. لذلك لا بد من إعطاء هذه البيانات أسماء معينة ليتمكن من الرجوع إليها، وتصنف هذه البيانات إلى نوعين: ثوابت ومتغيرات.

أولاً / الثوابت وأنواعها:

نحتاج في بعض البرامج إلى التعامل مع بعض الثوابت باستمرار، وبدلاً من كتابة قيمة هذا الثابت في كل مرة نستخدمه يمكن إعطاء هذا الثابت اسماً معيناً يستخدم بدلاً منه.

الثابت هو إعطاء اسم لقيمة معينة ويستخدم داخل البرنامج، ولا يمكن تغيير هذه القيمة عند تنفيذ البرنامج.

فائدة

عندما نحتاج إلى تغيير قيمة الثابت، فإنه يمكن تغييره في المكان الذي تم تعريفه فيه بدلاً من تغيير هذه القيمة في كل مرة استخدمنا فيها الثابت في البرنامج.

أنواع الثوابت:

١ ثابت عددي.

٢ ثابت حرفي.

طريقة تعريف الثوابت:

تُعرف الثوابت باستخدام الأمر (Const)

Const Const1 = Value



الوحدة السادسة: البرمجة بلغة (فيجول بيسك ستوديو)

حيث إن :

- ١ **Const**: الأمر الذي نستخدمه لتعريف الثابت.
- ٢ **Const I**: اسم الثابت، ويتم اختياره من قبل المبرمج.
- ٣ **Value**: القيمة التي سوف تخزن في هذا الثابت.

مثال

إذا أردنا تعريف ثابت الدائرة (النسبة بين محيط الدائرة وقطرها) يكون كالتالي :

$Const Pi = 3.14$

ثانياً / المتغيرات وأنواعها :

المتغير هو مكان في الذاكرة الرئيسية تخزن فيه بيانات وتعطى اسماً معيناً حتى يتم استرجاعها والتعامل معها داخل البرنامج، ويمكن تغيير ما يتم تخزينه، لذلك سميت بالمتغيرات. وتختلف المتغيرات باختلاف البيانات التي تخزن بها. كل متغير له اسم ونوع وقيمة.

أنواع المتغيرات :

تتعدد المتغيرات بحسب نوع البيانات التي تخزن بها، فمثلاً قد تكون رقماً صحيحاً أو رقماً عشرياً أو حرفاً أو مجموعة حروف. ويمكن تقسيم الأنواع إلى:

- ١ متغير عددي.
- ٢ متغير حرفي.
- ٣ متغير منطقي.

شروط تسمية المتغير :

ذكرنا أن المتغير يعطى اسماً من قبل المبرمج، ولكن لهذا الاسم شروطاً معينة وهي:

- ١ أن يتكون من حروف انجليزية (A..Z)، وأرقام، والرمز (_)، ولا يحتوي على فراغ أو أي رمز آخر.
- ٢ أن لا يبدأ برقم.
- ٣ أن لا يتجاوز (٢٥٥) حرف.
- ٤ أن لا يكون محجوزاً للغة البرمجة.

أمثلة على أسماء صحيحة للمتغيرات:

X Y A23 C_d

أمثلة على أسماء غير صحيحة للمتغيرات:

DIM 2DF IF@

فائدة

الأسماء المحجوزة (reserved words) :
هي الأسماء التي تكون إما أسماء لأنواع البيانات الموجودة في لغة البرمجة، أو أسماء لأوامر في اللغة لا يسمح باستخدامها. مثلاً في لغة (فيجول بيسك ستوديو) من الأمثلة على الأسماء المحجوزة الكلمات التالية: DIM, IF, FOR

طريقة تعريف المتغيرات :

تُعرَّف المتغيرات في لغة (فيجول بيسك ستوديو) باستخدام الأمر (Dim) وصيغته كالتالي:

Dim Var1 As Type

حيث إن :

- ١ **Var1**: اسم المتغير.
- ٢ **As**: رابط بين اسم المتغير ونوعه (من الأسماء المحجوزة للغة فيجول بيسك ستوديو).
- ٣ **Type**: نوع المتغير.

ويمكن تعريف أكثر من متغير في الأمر نفسه : *Dim Var1 As Type, Var2 As Type, ...*



الوحدة السادسة: البرمجة بلغة (فيجول بيسك ستوديو)

مثال

إذا أردنا تعريف متغير لتخزين اسم الطالب وليكن (name)، وهو من نوع متغير حرفي،

Dim name As String

يكون كالتالي:

Dim name As String, age As Integer

ولتعريف أكثر من متغير:

أنواع البيانات:

تتعامل لغة (فيجول بيسك ستوديو) مع أنواع مختلفة من البيانات، ولكل من هذه الأنواع اسم معين وسعة تخزينية معينة، نلخصها في الجدول التالي:

نوع البيانات	الاسم	الحجم	طريقة التعريف	مثال
عدد صحيح	Integer	٢ بايت	Dim X As Integer	X=25
عدد صحيح طويل	Long	٤ بايت	Dim Y AS Long	Y=12500000
عدد عشري	Single	٤ بايت	Dim X2 As Single	X2=10.5
عدد عشري مضاعف	Double	٨ بايت	Dim Y2 As Double	Y2=10.55555678
العملة	Currency	٨ بايت	Dim SR as Currency	SR=100.00
سلسلة نصية	String	بايت لكل حرف	Dim UserName as String	UserName="Admin"
منطقي	Boolean	٢ بايت	Dim B As Boolean	B=true
تاريخ	Date	٤ بايت	Dim D As Date	D=#04-10-99#
متنوع (أي يمكن تخزين أي من الأنواع السابقة)	Variant	١٦ بايت	Dim Var as Variant	Var=55.12 Var="Hello"

العمليات الحسابية والمنطقية:

٤-٦

أولاً / العمليات الحسابية في البرمجة:

تحتوي جميع لغات البرمجة على عمليات الحساب الأساسية: الجمع والطرح والضرب والقسمة والأس. وتختلف طريقة كتابة المعادلات الحسابية عن الطريقة الجبرية كما هو موضح في الجدول التالي:

العملية	الرمز	الصيغة الجبرية	الصيغة البرمجية
الجمع	+	$x + y$	$x + y$
الطرح	-	$x - y$	$x - y$
الضرب	*	xy	$x * y$
القسمة	/	$x \div y$ أو $\frac{x}{y}$	x / y
الأس	^	x^y	x^y

لاحظ الاختلاف في طريقة كتابة عمليتي الضرب والقسمة والأس.

قد تحتوي المعادلة الحسابية على أكثر من عملية مثال: $X + Y / Z$

عند محاولة حل هذه المعادلة يبرز لدينا سؤال مهم: هل ننفذ عملية الجمع أولاً أو عملية القسمة؟

بافتراض أن: $x=2, y=4, z=2$

جرب تنفيذ عملية الجمع أولاً ثم عملية القسمة ثم اعكس الترتيب.

هل الناتج نفسه؟

نخلص من هذا إلى أن الترتيب في تنفيذ العمليات يؤثر على الناتج، لذلك يجب أن تكون لدينا قوانين نتبعها لنعرف أي العمليات ننفذ أولاً.

ترتيب العمليات الحسابية :

١ العمليات التي في داخل الأقواس.

٢ عمليات الأس.

٣ عمليات الضرب والقسمة، وإذا تعددت نبدأ التنفيذ من اليسار إلى اليمين.

٤ عمليات الجمع والطرح، وإذا تعددت نبدأ التنفيذ من اليسار إلى اليمين.





مثال ١



ما نتيجة تنفيذ العملية التالية على جهاز الحاسب: $M = 2 * 6 / 3$ ؟

الحل:

حيث إن العمليات هنا هي الضرب والقسمة ولها الأولوية نفسها فسوف نبدأ التنفيذ من اليسار لليمين:

$$M = 12 / 3$$

تنفذ عملية الضرب أولاً:

$$M = 4$$

ثم عملية القسمة ثانياً:

مثال ٢



ما نتيجة تنفيذ العملية التالية على جهاز الحاسب: $M = 2 * 6 + 3^2$ ؟

الحل:

حيث إن العمليات هنا هي الضرب والجمع والأس ولها أولويات مختلفة فسوف نبدأ التنفيذ بالترتيب:

$$M = 2 * 6 + 9$$

تنفذ عملية الأس أولاً:

$$M = 12 + 9$$

ثم عملية الضرب ثانياً:

$$M = 21$$

وأخيراً عملية الجمع:

مثال ٣



ما نتيجة تنفيذ العملية التالية على جهاز الحاسب: $M=2*(6+3)^2$ ؟

الحل:

حيث أن العمليات هنا هي الضرب والجمع والأس ولها أولويات مختلفة، لكن لا يوجة أقواس حول عملية الجمع لذلك نبدأ بها:

$$M=2*9^2$$

نتفذ عملية الجمع أولاً:

$$M=2*81$$

ثم عملية الأس ثانياً

$$M=162$$

وأخيراً عملية الضرب

ثانياً / العمليات المنطقية في البرمجة:

ويقصد بها العمليات التي تتم فيها المقارنة بين قيمتين، سواء أكانتا عدديتين أو حرفيتين، متساويتين أو غير متساويتين، أو إحداهما أكبر أو أصغر من الأخرى. ويوضح الجدول التالي عمليات المقارنة المستخدمة في (فيجول بيسك ستوديو).

العامل	معناه
=	يساوي
<>	لا يساوي
>	أكبر من
<	أصغر من
>=	أكبر من أو يساوي
<=	أصغر من أو يساوي

يكون الناتج في عمليات المقارنة إما القيمة (True) أي: صحيح أو (False) أي: خطأ.

لو كان لدينا عمليات حسابية ومعها عملية مقارنة فإن أولوية التنفيذ تكون للعمليات الحسابية.



الوحدة السادسة: البرمجة بلغة (فيجول بيسك ستوديو)

مثال ١



ما نتيجة تنفيذ العملية التالية على جهاز الحاسب: $10 > 4$ ؟

الحل: النتيجة: (True) أي : صحيحة؛ لأن 10 فعلاً أكبر من 4

مثال ٢



ما نتيجة تنفيذ العملية التالية على جهاز الحاسب: $12 > 20$ ؟

الحل: النتيجة: (False) أي : خطأ؛ لأن 12 ليست أكبر من 20

مثال ٣



ما نتيجة تنفيذ العملية التالية على جهاز الحاسب: $4 + 3 * 5 < 4 * 6$ ؟

الحل: نتفذ العمليات الحسابية أولاً:

$$4 + 15 < 24$$

$$19 < 24$$

النتيجة: (True) أي : صحيحة؛ لأن 19 فعلاً أصغر من 24

ثالثاً / تحويل المعادلات الجبرية إلى الصيغة المستخدمة في البرمجة :

لاحظنا عند دراسة العمليات الحسابية أن طريقة كتابتها بالصيغة الجبرية تختلف عن طريقة كتابتها بالصيغة البرمجية. وعند قيامك بخطوة صياغة حل المسألة فغالباً ما تكون العمليات الحسابية مكتوبة بالصيغة الجبرية؛ لذلك يجب عليك عند كتابة البرنامج تحويل العمليات الحسابية من الصيغة الجبرية إلى الصيغة البرمجية.

حوّل المعادلة الجبرية الآتية إلى معادلة بصيغة برمجية.

$$Num = \frac{X^2}{A + B}$$

$$Num = X^2 / (A + B) \quad \text{الحل:}$$

مثال ١



حوّل المعادلة الجبرية الآتية إلى معادلة بصيغة برمجية.

$$X = 5Y - 4 \div 1$$

$$X = 5 * Y - 4 / 1$$

الحل:

مثال ٢



أدوات البرمجة بلغة (فيجول بيسك ستوديو):

٥-٦

أولاً / أدوات البرمجة :

الأدوات (Tools) تعرف بأنها أجزاء برامج جاهزة للاستخدام، أي أنها أعدت مسبقاً لتوفّر على المبرمج الوقت والجهد. وتستخدم هذه الأدوات لإجراء عمليات الإدخال والإخراج، ويتم ربطها بأوامر البرمجة التي تعالج البيانات المدخلة لتخرج لنا المعلومات المطلوبة.





الوحدة السادسة: البرمجة بلغة (فيجول بيسك ستوديو)

فكل ما يجب عليك عمله لاستخدام هذه الأدوات هو:

● وضعها على النموذج في المكان المناسب.

● ضبط الخصائص الخاصة بالأداة.

● كتابة أوامر البرمجة التي تتعامل مع هذه الأداة.

ثانيًا / خصائص الأدوات :

تختلف خصائص الأدوات باختلاف الأدوات، ولكن هناك خصائص مشتركة تشترك فيها كل الأدوات وهي التي سنتعرف عليها هنا. أما الخصائص الخاصة بكل أداة فسوف نتعرف عليها عند شرح كل أداة.

١ الخصائص المشتركة بين الأدوات :

الأداة	الخاصية
Name	تحديد اسم الأداة.
Textalign	تحديد محاذاة النص المكتوب (يمين- يسار-وسط).
Text	إظهار عنوان للنموذج أو نص داخل الأداة على الواجهة.
Font	تغيير نوع الخط وحجمه ونمطه.
ForeColor	تغيير اللون المكتوب به النص.
BackColor	تغيير لون الخلفية للأداة أو النموذج.
Location	تحديد موقع الأداة داخل النموذج.
Size	تغيير حجم النموذج أو الأداة.
Visible	إظهار أو إخفاء الأداة.

٢ ضبط خصائص الأدوات :

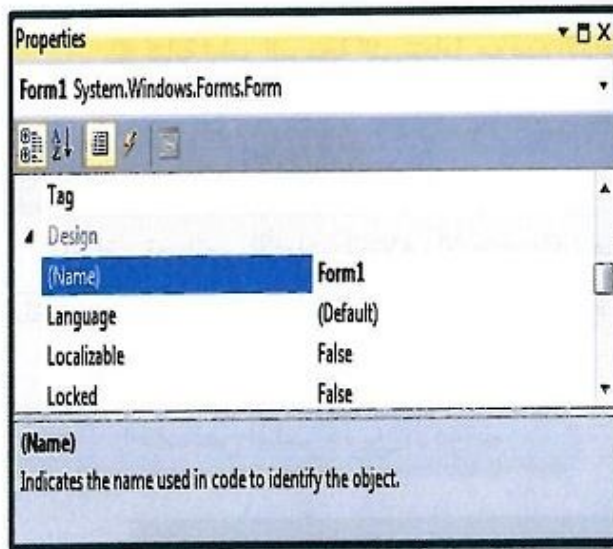
هناك طريقتان للتحكم بخصائص الأدوات، هما:

أ ضبط خصائص الأدوات أثناء تصميم البرنامج :

نستخدم إحدى الطرق التالية:

١- كتابة القيمة :

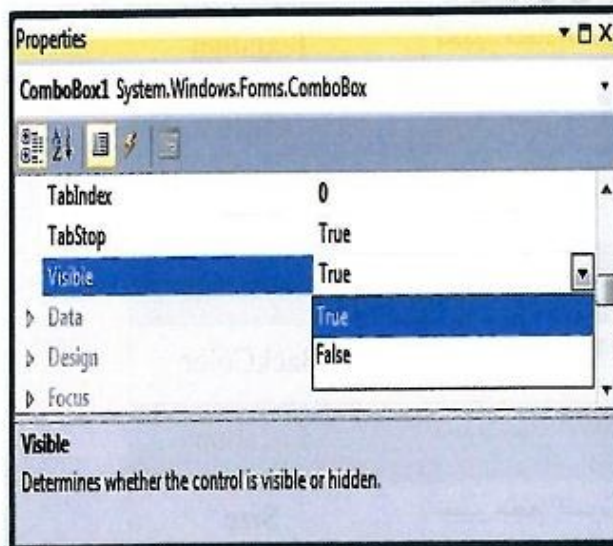
نكتب القيمة بواسطة لوحة المفاتيح في الخانة المخصصة داخل إطار الخصائص. كما في خاصية الاسم (Name) في الشكل (٤-٦)، حيث كتبنا القيمة (Form1) داخل خانة الاسم).



شكل (٤-٦) : خاصية الاسم للنموذج

٢ - اختيار القيمة :

نجد في خانة إدخال قيمة الخاصية مجموعة من الاختيارات نقوم باختيار إحداها في ضبط خاصية المشاهدة كما في الشكل (٥-٦) (visible)، أي هل النموذج أو الأداة تظهر في الواجهة أم لا؟ حيث يعرض خياران إما (False) أو (True).

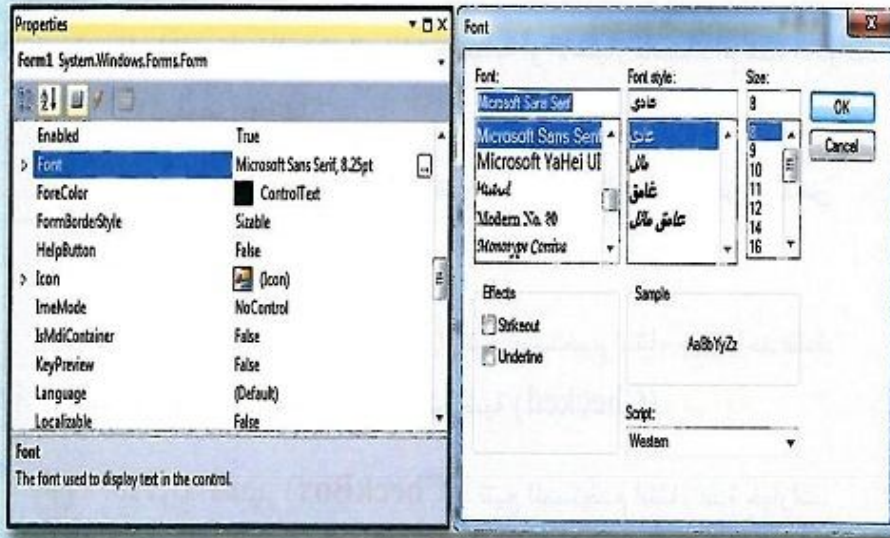


شكل (٥-٦) : خاصية المشاهدة



الوحدة السادسة: البرمجة بلغة (فيجول بيسك ستوديو)

٣ - ظهور نافذة خيارات:



عند اختيار نوع الخط بالنقر على خاصية الخط (Font) كما في الشكل (٦-٦). تظهر نافذة خصائص نوع الخط.

شكل (٦-٦) : خاصية الخط و نافذة الخط



فائدة

يقوم برنامج (فيجول بيسك ستوديو) بإعطاء قيم افتراضية للخواص، وذلك تسهيلاً للمبرمج. فمثلاً يعطي أسماء تلقائية لكل أداة تقوم برسمها، فعندما ترسم أداة تسمية لأول مرة فإن (فيجول بيسك ستوديو) يعطيها اسم (label1)، وعندما ترسم أداة التسمية مرة أخرى في النموذج نفسه فإن (فيجول بيسك ستوديو) يعطيها اسم (label2).. وهكذا لبقية الأدوات. وبإمكانك تغيير هذه الأسماء كما تريد.

ب ضبط خصائص الأدوات أثناء تشغيل البرنامج:

لتغيير خاصية الأدوات أثناء تنفيذ البرنامج فإننا نستخدم الصيغة التالية للوصول إلى تلك الخاصية:

القيمة الجديدة = الخاصية. اسم الأداة

مثلاً لتغيير خاصية (النص) في أداة مربع النص (Textbox1) نكتب مايلي:

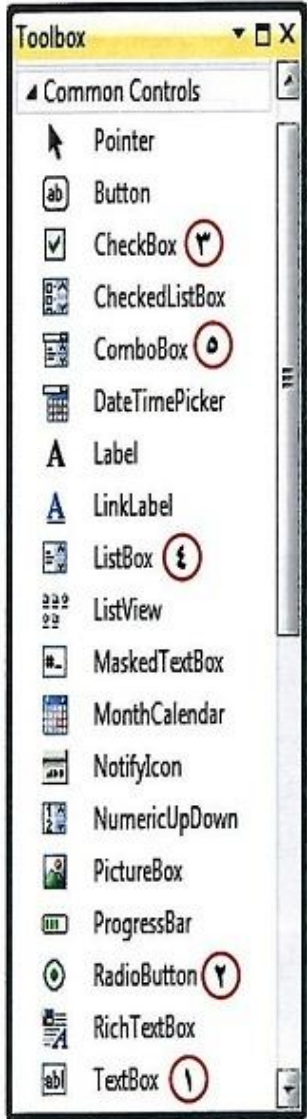
Textbox1.Text = "مدرسة الرياض"

تذكر

يجب وضع نقطة بين اسم الأداة والخاصية التي نريد الوصول إليها.

ثالثاً / أدوات إدخال البيانات :

نستخدم أدوات إدخال البيانات للحصول على البيانات التي يجب على البرنامج معالجتها. يقوم المستخدم بإدخال هذه البيانات بالكتابة أو الاختيار باستخدام هذه الأدوات، والشكل (٦-٧) يعرض بعضها.



١ أداة مربع النص (**TextBox**): تتيح للمستخدم كتابة نص وتخزين النص في الخاصية (**Text**).

٢ أداة زر الخيار (**RadioButton**): تتيح للمستخدم انتقاء خيار واحد فقط من عدة خيارات، وتخزن قيمها في الخاصية (**Checked**).

٣ أداة مربع الاختيار (**CheckBox**): تتيح للمستخدم انتقاء عدة خيارات، وتخزن قيمها في الخاصية (**Checkstate**) والخاصية (**Checked**).

٤ أداة مربع القائمة (**ListBox**): تعرض قائمة مكونة من عناصر يختار المستخدم أحدها وتخزن خيار المستخدم في الخاصية (**Text**) أو (**SelectedIndex**).

٥ أداة الخانة المركبة (**ComboBox**): تعطي المستخدم حرية الاختيار من قائمة أو إدخال اختياره كتابة وتخزنها في الخاصية (**Text**).

وتختلف طريقة الحصول على البيانات من هذه الأدوات؛ لذلك سوف نتعرف على طريقة كل أداة على حدة:

شكل (٦-٧): مربع الأدوات

١ مربع النص (**Text Box**)

إن البيانات التي نحصل عليها من مربع النص تختلف بحسب ما يدخله المستخدم، فقد تكون أرقاماً أو حروفاً. للحصول على البيانات من مربع النص نستخدم الصيغة التالية:

Var1=TextBox.Text



الوحدة السادسة: البرمجة بلغة (فيجول بيسك ستوديو)

حيث إن :

Var1 : متغير لتخزين البيانات فيه أياً كان نوعها عددية أم حرفية.

TextBox : اسم أداة مربع النص على النموذج.

Text : خاصية النص في أداة مربع النص التي تستقبل البيانات من المستخدم.

مثال



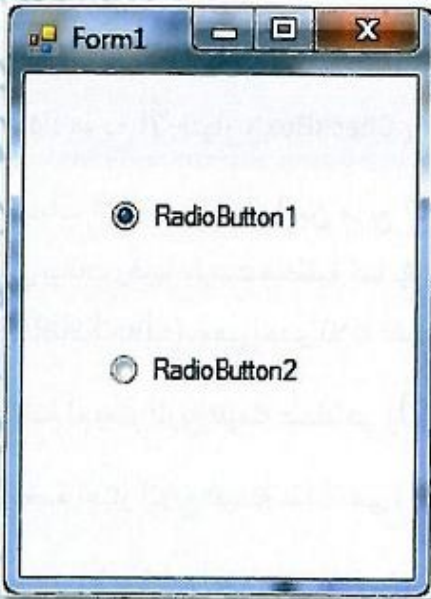
للحصول على القيمة المدخلة لاسم المستخدم في مربع النص والمسمى (Textbox1) وتخزينها في المتغير (Username) نكتب الأمر التالي:

`Username = Textbox1.Text`

في هذا المثال كما في الشكل (٨-٦) سوف يخزن في المتغير القيمة التالية:

`Username="Administrator"`

شكل (٨-٦) : أداة مربع النص



٢ أداة زر الخيار (RadioButton)

إن البيانات التي نحصل عليها من زر الخيار كما في الشكل (٩-٦) هي بيانات منطقية تخزن في الخاصية (checked) وهي إحدى قيمتين، إما:

➡ أن الزر قد تم اختياره فقيمته عندئذ هي (True).

➡ أن الزر لم يتم اختياره فقيمته عندئذ هي (False).

شكل (٩-٦) : أداة زر الخيار

والصيغة العامة للحصول على البيانات هي:

`Var1=RaidoButton.Checked`

حيث إن :

`Var1`: متغير لتخزين البيانات من نوع منطقي.

`RaidoButton`: اسم أداة زر الخيار على النافذة.

`Checked`: خاصية أداة زر الخيار التي تستقبل البيانات من المستخدم، إما (`True`) أو (`False`).

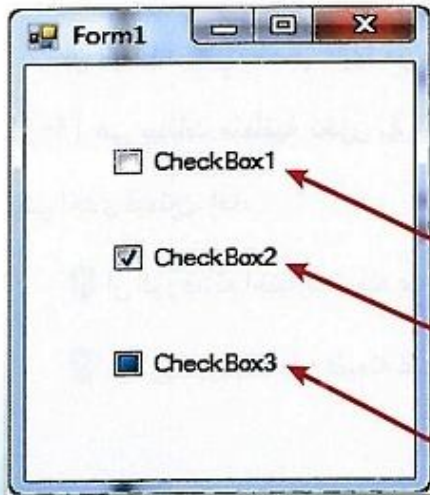
مثال



لمعرفة أن الزر المسمى (`OB1`) قد تم اختياره أم لا نكتب السطر التالي:

`H = OB1.Checked`

حيث إن (`H`) يمثل متغيراً منطقياً يُخزن إحدى القيمتين، إما صح (`True`) أو خطأ (`False`) لمعرفة إن كان الزر قد تم اختياره أم لم يتم.



شكل (١٠-٦) : أداة مربع الاختيار

٣ أداة مربع الاختيار (`CheckBox`)

إن البيانات التي نحصل عليها من مربع الاختيار كما في الشكل (١٠-٦) هي بيانات رقمية وليست منطقية كما في زر الخيار، وتخزن في الخاصية (`checkstate`)، وهي إحدى ثلاثة أشياء، إما:

● أن المستخدم لم يختار المربع فقيمته عندئذ هي (0).

● أن المستخدم اختار المربع فقيمته عندئذ هي (1).

● أن المستخدم لا يستطيع اختيار المربع فقيمته عندئذ هي (2).



الوحدة السادسة: البرمجة بلغة (فيجول بيسك ستوديو)

والصيغة العامة للحصول على البيانات هي :

Var1 = CheckBox.Checkstate

حيث إن :

Var1: متغير لتخزين البيانات فيه من نوع عددي.

CheckBox: اسم أداة مربع الاختيار على النافذة.

Checkstate: خاصية أداة مربع الاختيار التي تستقبل البيانات من المستخدم.

مثال



لمعرفة أن مربع الاختيار المسمى (CheckBox2) قد تم اختياره أم لا نكتب السطر التالي:

A = CheckBox2.Checkstate

حيث إن (A) يمثل متغيراً عددياً يخزن أحد الأعداد التالية (2, 1, 0) لمعرفة إن كان مربع الاختيار قد تم اختياره، أم لم يتم، أم لا يمكن اختياره.

٤ أداة مربع القائمة (ListBox)

البيانات الموجودة في أداة مربع القائمة مكونة من عدة عناصر، وللحصول على البيانات من القائمة عندما يختار

المستخدم أحد العناصر يعني أحد شيئين:

● رقم العنصر في القائمة بواسطة الخاصية (SelectedIndex).

● قيمته بواسطة الخاصية (Text).



الصفة العامة للحصول على رقم العنصر (SelectedIndex) هي:

`Var1 = ListBox.SelectedIndex`

حيث إن :

Var1 : متغير لتخزين البيانات فيه من نوع رقمي.

ListBox : اسم أداة مربع القائمة على النافذة.

SelectedIndex : خاصية أداة مربع القائمة التي تحدد رقم العنصر الذي اختاره المستخدم.

الصفة العامة للحصول على قيمة العنصر (Text) هي:

`Var1 = ListBox.Text`

حيث إن :

Var1 : متغير لتخزين البيانات فيه أياً كان نوعها عددية أم حرفية.

ListBox : اسم أداة مربع القائمة على النافذة.

Text : خاصية أداة مربع القائمة التي تحدد قيمة العنصر الذي اختاره المستخدم.

مثال



لمعرفة رقم العنصر الذي اختاره المستخدم من القائمة (ListBox1) التي تمثل هنا أسماء الدول:

`C = ListBox1.SelectedIndex`

حيث إن (C) يمثل متغيراً عددياً يخزن رقم العنصر الذي اختاره المستخدم.

شكل (٦-١١) : أداة مربع القائمة

وفي هذا المثال سوف يكون مخزن في المتغير (C) الرقم (3)، كما في الشكل (٦-١١).



مثال



شكل (٦-١٢) : أداة مربع القائمة

لمعرفة قيمة العنصر الذي اختاره المستخدم في القائمة (ListBox1):

$D = \text{ListBox1.Text}$

حيث إن (D) يمثل متغيراً يخزن قيمة العنصر في القائمة.

وفي هذا المثال تكون القيمة: "الإمارات العربية المتحدة" $D =$

كما في الشكل (٦-١٢).

٥ أداة الخانة المركبة (ComboBox)

أداة الخانة المركبة تجمع بين ميزات أداة النص وأداة مربع القائمة، حيث يستطيع المستخدم أن يختار من القائمة أو يكتب قيمة جديدة.

للحصول على البيانات من أداة الخانة المركبة نستخدم الصيغة التالية:

$\text{Var1} = \text{ComboBox.Text}$

حيث إن :

Var1: متغير لتخزين البيانات فيه أياً كان نوعها عددية أم حرفية.

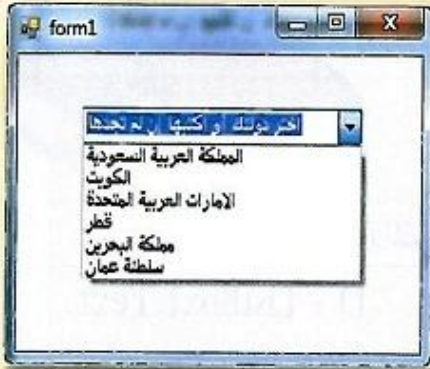
ComboBox: اسم أداة الخانة المركبة على النافذة.

Text: خاصية أداة الخانة المركبة التي تستقبل البيانات من المستخدم سواء باختياره من

القائمة أو بإدخاله للبيانات بالكتابة مباشرة.



مثال



شكل (٦-١٣) : أداة الخانة المركبة

للحصول على القيمة التي أدخلها المستخدم أو اختارها من القائمة في أداة الخانة المركبة (ComboBox1) وتخزينها في المتغير (C) نكتب الأمر التالي:

$C = \text{ComboBox1.Text}$

في هذا المثال سوف يخزن في المتغير (C) الدولة التي يختارها المستخدم، كما في الشكل (٦-١٣).

رابعاً / أدوات إخراج المعلومات :

نستخدم أدوات إخراج المعلومات لإظهار المعلومات للمستخدم على الواجهة بعد أن عالج البرنامج البيانات التي أدخلها المستخدم، ومن هذه الأدوات :

- ١ أداة مربع النص (TextBox) : وتخرج المعلومات بواسطة الخاصية (Text).
- ٢ أداة التسمية (Label) : وتخرج المعلومات بواسطة الخاصية (Text).

١ طريقة إخراج المعلومات إلى مربع النص (TextBox)

لإخراج المعلومات إلى مربع النص نستخدم الصيغة التالية :

$\text{TextBox.Text} = \text{Var1}$

حيث إن :

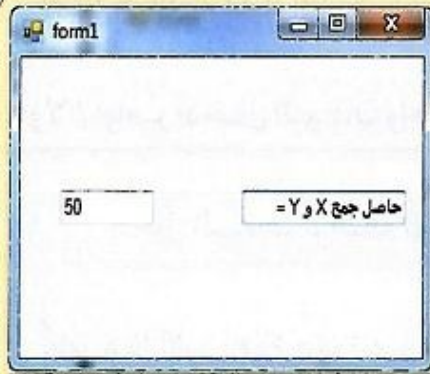
TextBox : اسم أداة مربع النص على النافذة.

Text : خاصية النص في أداة مربع النص التي سوف نخزن فيها قيمة المتغير (Var1).

Var1 : اسم المتغير.



مثال



شكل (١٤-٦) : استخدام أداة (Text)

لإظهار حاصل جمع عددين $(X+Y)$ في الأداة المسماة (Text4) نكتب السطر التالي :

`Text4.Text = X+Y`

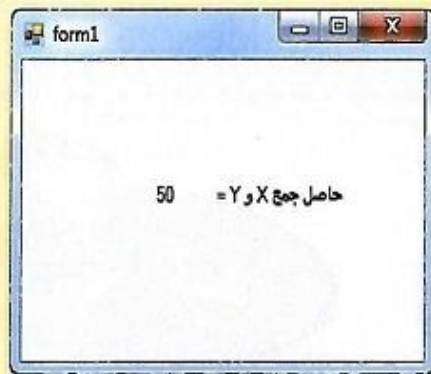
في هذا المثال كما في الشكل (١٤-٦) يظهر الناتج.

٢ طريقة إخراج المعلومات إلى أداة التسمية (Label)

لإخراج المعلومات إلى أداة التسمية نستخدم الصيغة التالية :

`Label.Text = Var1`

مثال



شكل (١٥-٦) : استخدام أداة (Label)

لإظهار حاصل جمع عددين $(X+Y)$ في الأداة المسماة (Label1) نكتب السطر التالي :

`Label1.Text = X+Y`

في هذا المثال كما في الشكل (١٥-٦) يظهر الناتج.

بعض الأوامر الأساسية للغة (فيجول بيسك ستوديو):

٦-٦

هناك أوامر داخلية في (فيجول بيسك ستوديو) تظهر للمستخدم نافذة مصممة سابقاً من قبل الشركة المنتجة للغة البرمجة، سواء لإدخال البيانات أو إخراج المعلومات، ومن هذه الأوامر:

أولاً / أوامر إدخال البيانات وإخراج المعلومات:

١ إدخال البيانات بواسطة الأمر (InputBox)



شكل (٦-١٦) : نافذة أمر (InputBox)

يُظهر هذا الأمر نافذة صغيرة غير النافذة الرئيسة في البرنامج تحتوي على مربع نص وزر أمر كما في الشكل (٦-١٦) ليدخل المستخدم البيانات التي يريدها في مربع النص ثم يضغط على زر الأمر. لإنشاء هذه النافذة يجب أن نكتب الأمر الخاص بها، وصيغته كالتالي:

Var1 = InputBox (message)

حيث إن : Var1 : اسم المتغير الذي سوف تخزن فيه البيانات التي أدخلها المستخدم وقد تكون حرفية أو رقمية.
InputBox : أمر إنشاء هذه النافذة.
Message : النص الثابت الذي يظهر في النافذة ويوضع بين أقواس اقتباس هكذا " " .

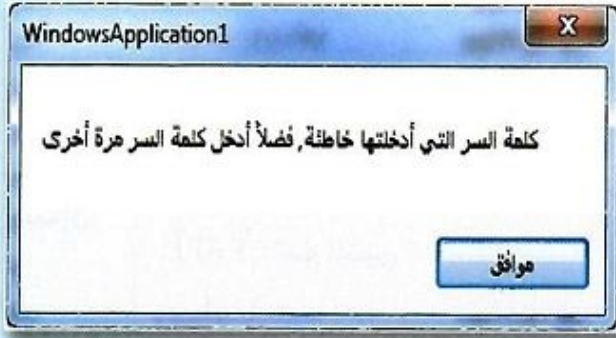
مثال



لو أردنا أن نطلب من المستخدم إدخال اسم المستخدم فإننا نكتب:
Username=InputBox("فضلاً أدخل اسم المستخدم")



٢ إخراج المعلومات بواسطة الأمر (MsgBox)



شكل (١٧-٦) : نافذة أمر (MsgBox)

يظهر هذا الأمر نافذة صغيرة غير النافذة الرئيسة في البرنامج تحتوي على المعلومات التي نريد للمستخدم قراءتها كما في الشكل (١٧-٦)، ولإنشاء هذه النافذة يجب أن نكتب الأمر الخاص بها: وصيفته كالتالي:

MsgBox (message)

حيث إن :

MsgBox: أمر إنشاء هذه النافذة.

message: اسم المتغير أو نص ثابت يوضع بين أقواس اقتباس هكذا " " ليظهر في النافذة.

مثال



لو أردنا أن نخبر المستخدم أن كلمة السر التي أدخلها خاطئة فإننا نكتب:

("كلمة السر التي أدخلتها خاطئة، فضلاً أدخل كلمة السر مرة أخرى") MsgBox

وإذا أردنا عرض قيمة المتغير A فنكتب :

MsgBox (A)

ثانياً / أمراً إسناد :

تذكر

عند إسناد سلسلة نصية إلى متغير
حرفي نضعها داخل علامتي اقتباس .

يقصد به تخزين قيمة معينة داخل متغير، وقد تكون هذه القيمة
عدد أو عملية حسابية أو سلسلة حرفية.
وصيغته كالتالي:

$Var1 = Value$

حيث إن :

Var1 : اسم المتغير.

Value : القيمة التي نريد تخزينها في المتغير.

مثال

لو أردنا تخزين اسم في متغير حرفي تم تعريفه مسبقاً : $Name = "Nor"$

لو أردنا تخزين رقم في متغير عددي تم تعريفه مسبقاً : $Num = 10$

لو أردنا تخزين ناتج عملية حسابية في متغير عددي تم تعريفه مسبقاً : $X = V + 20 \times I$

ثالثاً / الجمل الشرطية :

يوجد في لغة (فيجول بيسك ستوديو) عدة جمل للتحقق من الشرط وهي :

١ الجمل الشرطية (IF) :

وتعد من أهم الأوامر في البرمجة، وتوجد في أغلب لغات البرمجة. ونستخدمها في البرنامج عند حاجتنا إلى اتخاذ
قرارات مختلفة على حسب شرط معين، مثلاً إن كانت درجة الطالب أكبر من أو تساوي (60) فهو ناجح، وإن كانت أقل
من (60) فهو غير مجتاز.



الوحدة السادسة: البرمجة بلغة (فيجول بيسك ستوديو)

وتحتوي لغة (فيجول بيسك ستوديو) على عدة صيغ، منها:

أ صيغة (IF-THEN) :

IF condition THEN statement

فإذا تحقق الشرط (condition)، أي كان صحيحاً (True) فسيتم تنفيذ الأمر (statement)، وإذا لم يكن الشرط صحيحاً فلن يتم تنفيذ أي أمر.

مثال



IF grade >= 60 THEN Result = "ناجح"

ب صيغة (IF-THEN-END IF) :

```
IF condition THEN
...
statements
...
END IF
```

هذه الصيغة هي الصيغة الأولى نفسها، إلا أنه سيتم تنفيذ أكثر من أمر إذا كان الشرط صحيحاً؛ لذلك احتجنا إلى وضع كلمة (END IF) لتوضيح نهاية مجموعة الأوامر التي سيتم تنفيذها.

مثال



```
IF grade >= 60 THEN  
Result = "ناجح"  
Text1 = "مبروك"  
END IF
```

جـ صيغة (IF-THEN-ELSE) :

```
IF condition THEN  
...  
statements1  
...  
ELSE  
...  
statements2  
END IF
```

تختلف هذه الصيغة عن الصيغ السابقة، حيث يتم تنفيذ أوامر (statements1) في حالة كون الشرط صحيحاً (True)، وفي حالة كون الشرط غير صحيح (False) يتم تنفيذ مجموعة الأوامر (statements2).





مثال



```
IF grade >= 60 THEN
Result = "ناجح"
ELSE
Result = "غير مجتاز"
END IF
```

د صيغة (IF-THEN-ELSEIF) :

```
IF condition1 THEN
...
statements1
...
ELSEIF condition2 THEN
...
statements2
...
ELSE
...
Statements3
...
END IF
```

تختلف هذه الصيغة عن الصيغة السابقة في وجود أكثر من شرط يتم التحقق منه.
إذا كان الشرط الأول (condition1) صحيحاً فإنه ينفذ الأوامر (statements1) فقط.
أما إذا كان الشرط الأول غير صحيح فإنه يختبر شرطاً جديداً وهو (condition2) وإذا كان صحيحاً فإنه ينفذ مجموعة الأوامر (statements2) فقط.
أما إذا كان الشرط الثاني غير صحيح فإنه ينفذ مجموعة الأوامر (statements3). وقد يكون لدينا شرط ثالث ورابع وهكذا.

مثال



```
IF grade >= 90 THEN
Result="ممتاز"
ELSEIF grade >= 80 THEN
Result="جيد جدا"
ELSEIF grade >= 70 THEN
Result="جيد"
ELSEIF grade >= 60 THEN
Result="مقبول"
ELSE
Result="غير مجتاز"
END IF
```

SELECT CASE expression
CASE prob1

...
statements1

...
CASE prob2

...
statements2

...
[CASE ELSE

...
statements3

...]
END SELECT

٢ الجملة الشرطية (Select Case) :

تستخدم هذه الجملة إذا كان هناك عدة احتمالات للشرط. فبدلاً من استخدام جملة (IF) طويلة ومعقدة تقوم هذه الجملة بالعمل نفسه ولكن بطريقة أسهل. حيث تختبر هذه الجملة تعبيراً أو شرطاً معيناً قد يكون لقيمته أكثر من احتمال.

وصيغتها :





SELECT CASE : بداية الجملة.

expression: الشرط أو التعبير الذي نريد اختبار قيمته، وقد يكون

متغيراً أو عملية حسابية أو عملية منطقية.

CASE: توضع قبل كل احتمال.

prob1.prob2....: القيم المحتملة للتعبير.

statements1: الأوامر التي تنفذ في حالة تحقق القيمة.

CASE ELSE: إذا لم يتحقق أي احتمال من الاحتمالات السابقة

فسوف تنفذ الأوامر التي بعد هذه العبارة، وهي اختيارية، أي إذا لم تكن بحاجة لها فلا يجب استخدامها.

END SELECT : نهاية الجملة.

تذكر

وجود الجزء (CASE ELSE) داخل الأقواس [] وذلك لأنه جزء اختياري من الصيغة إذا كنا بحاجة إليه نضعه. وعدم وجوده لا يؤثر على صحة الجملة.

طريقة عمل هذه الجملة كالتالي:

يقوم البرنامج بتقييم التعبير (expression)، ثم يقارنه مع الاحتمالات الواردة عند كل كلمة (CASE prob1.prob2,...)، فإذا وافق قيمة التعبير أحد هذه الاحتمالات فسوف ينفذ الأوامر التي جاءت بعد الاحتمال الصحيح وحتى جملة (CASE) التالية. فلو كان التعبير يوافق الاحتمال الأول (prob1) فإن البرنامج سوف ينفذ مجموعة الأوامر (statements1) فقط، ويذهب إلى نهاية الجملة.

أما إذا لم يوافق التعبير أيّاً من الاحتمالات الموجودة، فإذا كان لدينا (CASE ELSE) فإن البرنامج سوف ينفذ مجموعة الأوامر التي تأتي بعده، وإذا لم يكن لدينا (CASE ELSE) (لأنه اختياري لا يلزم وجوده دائماً) فإن الجملة تنتهي دون تنفيذ أي أوامر.

تذكر

يجب التأكد من أن نوع بيانات (expression) هو نفسه نوع البيانات الموجودة في الاحتمالات.

مثال



لأوردنا تطبيق المثال السابق نفسه في جملة (IF-THEN-ELSEIF) ولكن باستخدام جملة

:SELECT CASE

SELECT CASE grade

CASE 90 to 100

Result="ممتاز"

CASE 80 to 89

Result="جيد جدا"

CASE 70 to 79

Result="جيد"

CASE 60 to 69

Result="مقبول"

CASE ELSE

Result="غير مجتاز"

END SELECT



رابعاً / حلقات التكرار:

من أهم أوامر البرمجة التي تساعدنا على تكرار مجموعة من الأوامر الأخرى عدة مرات. ويوجد في لغة (فيجول بيسك ستوديو) عدة أوامر للتكرار ومن أهمها:

١ الأمر (For .. Next) :

يكرر هذا الأمر مجموعة من الأوامر بعدد من المرات محدد ومعروف مسبقاً.

صيغته:

```
FOR counter=start TO end [STEP step]
```

```
.....
```

```
statements
```

```
.....
```

```
NEXT
```

حيث إن :

Counter : هو متغير يخزن فيه عدد مرات التكرار يبدأ من قيمة أولية ويتغير إلى أن يصل إلى القيمة النهائية المحددة له، ويسمى هذا المتغير بالعداد.

Start : القيمة الأولية التي يبدأ بها العداد.

end : القيمة النهائية التي يجب أن يتوقف عندها العداد.

Step : القيمة التي يتم بها زيادة العداد في كل دورة تكرار. وهي اختيارية، فإذا لم نذكرها فإن الزيادة سوف تكون (1).

Statements : مجموعة الأوامر أو قد يكون أمراً واحداً تُنفذ بعدد مرات التكرار.

NEXT : نهاية جملة التكرار، أي أن الأوامر التي تأتي بعده لا تدخل في التكرار.

مثال



لو أردنا جمع الأعداد من (1) إلى (10) وتخزينها داخل المتغير (sum):
يمكن أن نكتب أوامر بهذه الطريقة:
يجب أن نضع قيمة ابتدائية في المتغير قبل أن نجمع عليه

Sum=0

ثم نبدأ بجمع الأعداد واحداً تلو الآخر، وهذا يتطلب منا أن نكتب (10) أوامر كالتالي:

sum=sum+1

sum=sum+2

.....

sum=sum+10

الأفضل من هذه الطريقة أن نستخدم جملة تكرار كالتالي:

For count=1 to 10

sum=sum+count

Next

حيث إننا لم نحدد قيمة (Step) هنا فإن الزيادة سوف تكون (1) في كل مرة. أي يبدأ العداد من القيمة (1) ويزداد إلى أن يصل إلى القيمة (10). وفي كل مرة يجمع هذه القيم على المتغير (sum). وبعد تنفيذ التكرار سوف يكون لدينا في المتغير (sum) مجموع الأعداد من (1) إلى (10).



مثال



لو أردنا جمع الأعداد الفردية من (1) إلى (11) فإننا سوف نستخدم الطريقة السابقة نفسها، وسنقوم بتحديد مقدار الزيادة على أن يكون (2) كالتالي:

sum=0

For count=1 to 11 STEP 2

sum=sum+count

Next

فكر



ما الذي سيجعل الشرط غير صحيح؟
لابد أننا سوف نقوم بعمليات داخل التكرار تؤثر على الشرط.

٢ الأمر (DO WHILE) :

نستخدم هذا الأمر إذا كان عدد مرات التكرار غير محدد، ولكن لدينا شرطاً هو الذي يحدد متى ينتهي التكرار، أي أنه متى ما كان الشرط صحيحاً نفذنا الأوامر واستمر التكرار، ومتى ما صار الشرط غير صحيح توقف التكرار.

صيغته:

Do While condition

.....
statements

.....
Loop

حيث إن :

condition : الشرط الذي يتم التحقق منه، ثم تنفيذ التكرار إذا كان صحيحاً والتوقف إذا كان خاطئاً.
statements : مجموعة الأوامر التي تنفذ داخل التكرار.

مثال



```
A=1
sum=0
Do While A <= 10
sum=sum+A
A=A+1
Loop
```

في هذا المثال تُجمع الأرقام من (1) إلى (10) كما في المثال السابق. ونلاحظ هنا أن شرط التوقف هو وصول قيمة المتغير (A) إلى (10).

مثال



لو أردنا جمع الأعداد الزوجية من (0) إلى (10).

```
A=0
sum=0
Do While A <=10
sum=sum+A
A=A+2
Loop
```




خامساً / المصفوفات :

لو كان لديك درجات (100) طالب تريد عمل بعض الإحصاءات عليها كمعرفة المتوسط وأعلى درجة وأقل درجة. فأين سوف تخزن هذه الدرجات؟ هل سوف تعرّف (100) متغير لتخزينها؟ يبدو هذا غير منطقي، أليس كذلك؟ يوجد في لغة (فيجول بيسك ستوديو) وفي أغلب لغات البرمجة ما يسهل علينا عملية تعريف عدد كبير من المتغيرات تشترك في كونها تمثل نوع البيانات نفسه وهي المصفوفات.

المصفوفة (Array) هي مجموعة من المتغيرات لها الاسم نفسه ونوع البيانات نفسه ويتم تعريفها في جملة واحدة.

صيغتها :

`Dim var1(n) As Type`

حيث إن :

`var1` : اسم المصفوفة.

`n` : (عدد عناصر المصفوفة - 1).

`Type` : نوع البيانات المخزنة في العناصر.

مثال



`Dim Grades(99) As Integer`

هنا عرّفنا مصفوفة لتخزين درجات (100) طالب.
أليس هذا أفضل من تعريف (100) متغير؟

١ فوائد المصفوفات :

كما لاحظت في المثال السابق، فإن استخدام المصفوفة قد وفر علينا كثيراً من الوقت والجهد الذي كنا سنبدله في تعريف (100) متغير ومعالجة كل متغير على حدة. فالمصفوفات سهلت لنا هذه المهمة، ونستطيع باستخدام أوامر التكرار أن نتعامل مع المصفوفات بسهولة. كما يؤدي استخدام المصفوفات إلى صغر حجم البرنامج.

٢ التعامل مع المصفوفات :

للوصول إلى عنصر من عناصر المصفوفة نكتب اسم المصفوفة وبين قوسين رقم العنصر، ولكن يجب التنبيه إلى أن ترقيم العناصر في المصفوفة يبدأ من الصفر، أي أن أول عنصر في المصفوفة رقمه (0) ثم العنصر الثاني (1) وهكذا إلى آخر عنصر في المصفوفة الذي يكون رقمه (عدد عناصر المصفوفة - 1).

مثال: لو عرفنا مصفوفة فيها (10) أعداد كالتالي: Dim A(9) AS Integer

وخرزنا فيها مجموعة من الأرقام، سيكون شكل المصفوفة كالتالي:

المصفوفة A									
رقم العنصر	0	1	2	3	4	5	6	7	8
القيمة	4	3	5	6	2	15	7	9	12

لو أردنا تغيير قيمة العنصر الخامس لكتبنا:

$$A(4)=10$$

غالباً ما تُعالج جميع عناصر المصفوفة بالتسلسل، أي واحداً تلو الآخر، وما يسهل علينا هذه المعالجة هو استخدام حلقات التكرار، حيث نجعل العداد يمثل رقم العنصر كما في المثال التالي:

تذكر

أن تبدأ العداد من الصفر وتنتهي بعدد العناصر - 1 عند استخدامك للمصفوفات.



مثال

لقراءة درجات (100) طالب نقوم بالتالي:

```
Dim Grades(99) As Integer
FOR count=0 To 99
Grades(count)=InputBox ("أدخل الدرجة")
NEXT
```

لو أردنا أن نجد متوسط درجات الطلاب من المثال السابق، فيجب علينا
أولاً أن نجمع جميع الدرجات ثم نقسم على عدد الطلاب.
نعرف أولاً متغيراً لحساب المجموع وآخر لحساب المعدل:

```
Dim sum As Integer, average As Single
sum=0
For count=0 To 99
sum=sum+Grades (count )
Next
average=sum/100
```

لو أردنا أن نجد أعلى درجة من درجات الطلاب .
نعرف أولاً متغيراً لتخزين أعلى درجة:

```
Dim max As Integer
max=0
For count=0 To 99
IF Grades(count)>max THEN max=Grades(count)
Next
```


تمريبات



س ١: ما مراحل كتابة البرنامج بلغة (فيجول بيسك ستوديو)؟

تصميم الواجهات.

ضبط خصائص الواجهات. كتابة أوامر البرمجة.

س ٢: ماذا نسمي أماكن تخزين البيانات في الذاكرة الرئيسية؟

هياكل البيانات.

س ٣: ماذا يعني الأمر التالي: Dim Number As Integer؟

هذا من أوامر برنامج بلغة فيجوال بيسك أي تعريف للمصفوفات والثوابت الصحيحة أي تستخدم الأمر Dim في طريقة تعريف الثوابت والذي تستخدمه أيضاً لتعريف المصفوفة.

س ٤: ما الفرق بين الثوابت والمتغيرات؟

الثابت: هو إعطاء اسم لقيمة معينة ويستخدم داخل البرنامج، ولا يمكن تغيير هذه القيمة عند تنفيذ البرنامج.

المتغير: هو مكان في الذاكرة الرئيسية تخزن فيه بيانات وتعطى اسماً معيناً.



تمريبات



س٥: هل الأسماء التالية يمكن استخدامها لتسمية المتغيرات :

2ABC, 123, AB2, AB_2, Num one, While, aBxY, Case

نعم يمكن ذلك.

س٦: بافتراض المتغيرات والقيم التالية : $X=20, Y=33, Z=9, A=2$

ما نواتج العمليات الحسابية التالية :

$$X+Z*A^2$$

عملية الأس أولاً: $20 + 9 * 4$

عملية الجمع ثانياً: $20 + 36$

أخيراً عملية الضرب: 56

تمريبات



$$(Y+X/A+1) / (Z+A)$$

العمليات لتي داخل الأقواس أولاً:

$$(33+20 / 2+1) / (9+2)$$

$$(53/11) / 11$$

$$17,7 / 11$$

ثانياً عملية القسمة: ١,٦

$$X*5^A$$

عملية الأس أولاً: ٢٥ * ٢٠

ثم عملية الضرب: ٥٠٠



تمارينات



س ٧: حوّل العمليات الجبرية التالية إلى صيغة برمجية:

$$\frac{x+y}{9*3} + M^x$$

$$3*9/(X+Y)M^X$$

$$zx+4+y$$

$$X^2+Y+4$$

$$3y^{x+6}$$

$$X^y(x+6)$$





س ٨: ماذا تسمى أجزاء البرامج الجاهزة التي توفرها لغة (فيجول بيسك ستوديو) لتوفر على المبرمج الجهد والوقت؟

الأدوات: Tools

س ٩: كيف نغير النص المكتوب على زر أمر اسمه (Button)؟

سيظهر لنا يمين الشاشة **Caption** أمامها **Command 1** نحذفها ونكتب النص الذي نريده.

س ١٠: ماذا نسمي الأدوات التي تستقبل البيانات من المستخدم؟ اذكر ثلاثاً منها، واذكر متى تستخدم.

أداة مربع النص: تتيح للمستخدم كتابة نص واستخدام وتخزين النص في الخاصية **Text**
أداة زر الخيار: تتيح للمستخدم انتخاب خيار واحد فقط من عدة خيارات، وتخزن قيمتها في الخاصية **Checked**
أداة الخانة المركبة: تعطي المستخدم حرية الاختيار منقائمة أو إدخال اختياره كتابة وتخزينها في الخاصية **Text**



س ١١: ضع علامة (✓) أمام العبارة الصحيحة ، وعلامة (X) أمام العبارة الخاطئة ، مع تصحيح الخطأ :

أ) يجب علينا عند البدء في عمل برنامج بلغة (فيجول بيسك ستوديو) كتابة أوامر البرمجة أولاً. (✓)

ب) يمكن للبرنامج أن يغير قيمة الثابت عند تنفيذ عملية حسابية. (X)

خطأ - لا يمكن تغيير قيمة الثابت في البرنامج.

ج) نتائج العمليات المنطقية هي دائماً أرقام. (✓)

د) ننفذ عمليات الضرب والقسمة قبل عمليات الجمع والطرح. (✓)

هـ) إذا أردنا المستخدم أن يدخل رقم هاتفه نستخدم أداة مربع الاختيار. (✓)

س ١٢: اذكر ثلاث طرق لإخراج معلومات للمستخدم.

طريقة إخراج المعلومات إلى مربع النص Text Box

طريقة إخراج المعلومات إلى أداة التسمية Label

إخراج المعلومات بواسطة الأمر Msg Box



الوحدة السادسة: البرمجة بلغة (فيجول بيسك ستوديو)

س ١٣ : هل يمكن أن تستخدم الأداة نفسها للإدخال والإخراج؟ وضح إجابتك.

نعم يمكن ذلك – هناك أوامر داخلية في (فيجول بيسك ستوديو) تظهر للمستخدم نافذة مصممة سابقاً من قبل الشركة المنتجة للغة البرمجة، سواء لإدخال البيانات أو لإخراج المعلومات.

س ١٤ : ما الذي يحدث بعد تنفيذ الإجراء التالي :

```
Dim Num As Integer, Name As String
```

```
Num=0
```

```
If Num<1 Then Name=InputBox("أهلاً بك الرجاء إدخال اسمك")
```

```
MsgBox("أهلاً بك يا " + Name)
```

```
ENDIF
```

ستظهر نافذة صغيرة تحتوي على الرسالة التالية "أهلاً بك الرجاء إدخال اسمك" وتحتوي على مربع إدخال تقوم بكتابة الاسم فيه.

بعد إدخال الاسم وليكن مثلاً "سيف" والضغط على مفتاح الإدخال يظهر مربع الرسالة ويحتوي على العبارة "أهلاً بك يا سيف".

س ١٥ : اكتب الأمر التالي، ولكن باستخدام جملة (Select) :

```
IF price >= 1000 Then
MsgBox("السعر غالي جدا")
ElseIF price >= 500 Then
MsgBox("السعر غالي")
ElseIF price >= 200 Then
MsgBox("السعر معقول")
Else MsgBox("السعر رخيص")
ENDIF
```

```
SELECT CASE price
CASE price >= 1000
RESULT = MsgBox("السعر غالي جدا")
CASE 500 TO 599
RESULT = MsgBox("السعر غالي")
CASE 200 TO 499
RESULT = MsgBox("السعر معقول")
CASE ELSE
Result = MsgBox("السعر رخيص")
End SELECT
```


س١٦: لو كان لديك مصفوفة اسمها (Grades) ومخزن فيها درجات (100) من الطلاب، فما أقل درجة؟

بعد تعريف المصفوفة نقوم بكتابة الكود التالي لحساب أقل درجة:

```
Dim min
min = 0
for count = 0 to 99
if Grades(count) < Grades(count+1)
then min = Grades(count)
next
```





تدريبات الوحدة السادسة

البرمجة بلغة (فيجول بيسك ستوديو)

..... التدريب الأول : التعرف على بيئة العمل

في هذا التدريب ستتعلم :

- تشغيل برنامج (فيجول بيسك ٢٠١٠) والبدء في مشروع جديد.
- مكونات برنامج (فيجول بيسك ٢٠١٠).
- مربع الأدوات.



متطلبات التدريب

- برنامج (فيجول بيسك ٢٠١٠).

مقدمة التدريب

تعد البرمجة بلغة (فيجول بيسك) شيقة وممتعة؛ وذلك لما تمتاز به من تحكم المبرمج في البرامج التي يقوم بتصميمها من ناحية واجهات الإدخال للمستخدم، والعمليات الإجرائية للبرنامج، وأخيراً المخرجات التي يحصل عليها المستخدم، وستتعرف في التدريبات القادمة على البرمجة باستخدام (فيجول بيسك)، لتكتمل لديك المهارات اللازمة في إنشاء برامج متكاملة.



خطوات التدريب

أولاً/ تشغيل برنامج (فيجول بيسك) والبدء بمشروع جديد:

لتشغيل برنامج (فيجول بيسك):

١- انقر على زر (ابدأ).

٢- افتح قائمة (كافة البرامج).

٣- اختر المجلد (Microsoft Visual Studio 2010)

حتى تتسدل القائمة الموجودة بداخله.

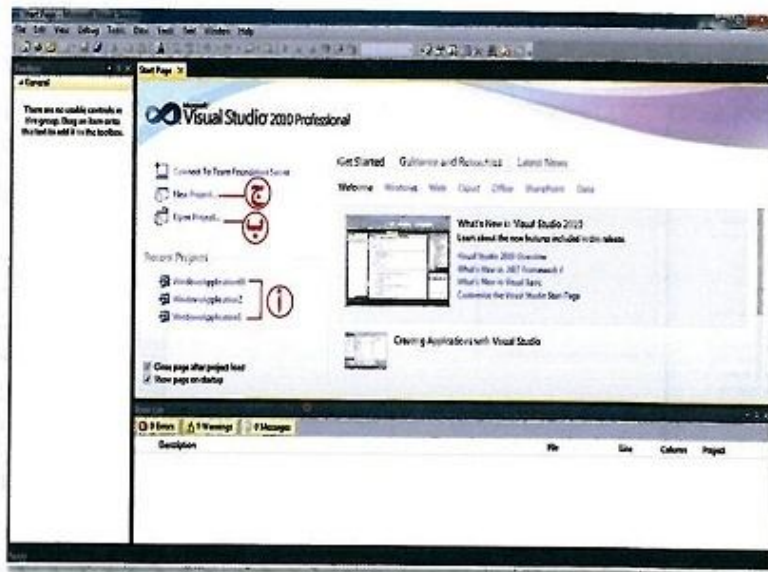
٤- انقر على برنامج (Microsoft Visual Studio 2010)

كما في الشكل (١-١-٦).

٥- سيشتغل برنامج (فيجول بيسك ٢٠١٠)، وتظهر نافذة (StartPage)

كما في الشكل (٢-١-٦)، التي من خلالها نستطيع عمل التالي:

شكل ١-١-٦: تشغيل برنامج (فيجول بيسك ٢٠١٠)



شكل ٢-١-٦: النافذة الافتتاحية لبرنامج (فيجول بيسك ٢٠١٠)



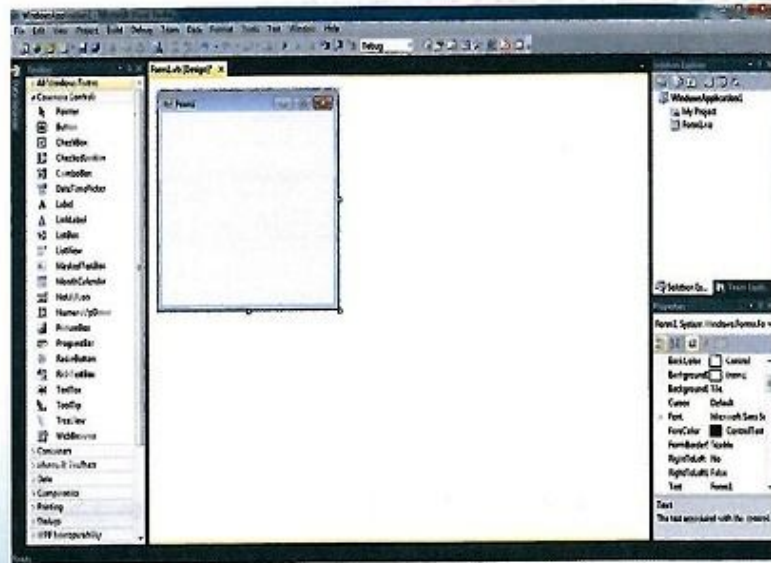
- أ - عرض المشاريع الأخيرة (Recent Project ...):
تُعرض في هذا الخيار المشاريع التي تم العمل عليها مؤخراً.
- ب - فتح مشروع موجود سابقاً (Open Project...):
تُفتح في هذا الخيار مشاريع موجودة مسبقاً على القرص.
- ج - إنشاء مشروع جديد (New Project...):

يقوم (فيجول بيسك ٢٠١٠) بإطلاق اسم مشروع على مجموعة الملفات التي تمثل البرنامج. ولإنشاء مشروع جديد اتبع الخطوات التالية:



شكل ٦-١-٣: نافذة أنواع المشاريع

- ١ اختر (New Project...)
فتظهر النافذة كما في الشكل (٦-١-٣) وفيها أنواع متعددة من المشاريع.
- ٢ اختر (Visual Basic).
- ٣ اختر (Windows Forms Application).
- ٤ انقر (Ok).
- ٥ تظهر نافذة مشروع جديد كما في الشكل (٦-١-٤)، تمثل واجهة المستخدم في برنامج (فيجول بيسك).



شكل ٦-١-٤: مشروع جديد في (فيجول بيسك ٢٠١٠)



ثانياً/ مكونات برنامج (فيجول بيسك):

لنتمكن من العمل على برنامج (فيجول بيسك ستوديو) لا بد من التعرف على مكوناته وأجزائه وطريقة العمل فيه، وتتكون واجهة البرنامج من المكونات التالية:

١ شريط القوائم:

يحتوي هذا الشريط على القوائم الرئيسة في (فيجول بيسك)، وتحتوي كل قائمة على مجموعة من الأوامر المتعلقة بموضوع معين.

File Edit View Project Build Debug Team Data Format Tools Test Window Help

٢ شريط الأدوات القياسي:

يحتوي هذا الشريط على مجموعة الأوامر الموجودة في شريط القوائم، ولكنها وضعت هنا لتسهيل الوصول إليها اختصاراً للوقت.



٣ مربع الأدوات (Toolbox):

هي الأدوات التي نستخدمها في بناء واجهة البرامج كما في الشكل (٥-١-٦)، ولإظهارها اختر الأمر (Toolbox) من قائمة (View).

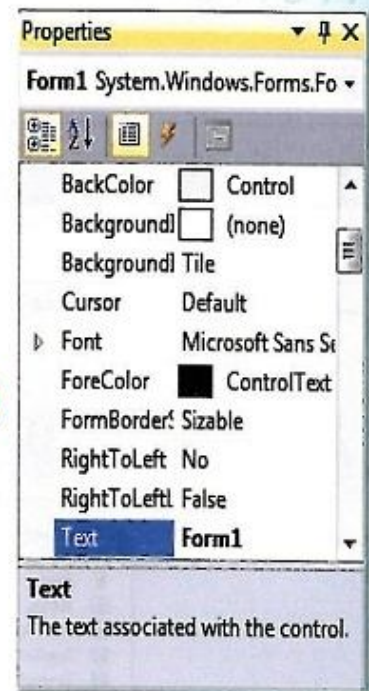


شكل ٥-١-٦: مربع الأدوات



شكل ٦-١-٦: محتويات المشروع

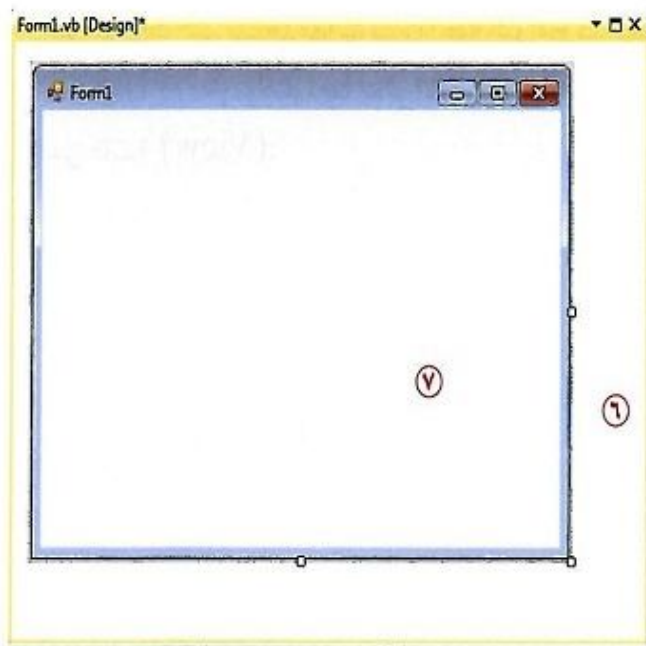
٤ محتويات المشروع:
نافذة تحتوي على جميع الملفات والنماذج التي يتكون منها المشروع، كما في الشكل (٦-١-٦).



شكل ٧-١-٦: إطار الخصائص

٥ إطار الخصائص (Properties):

لكل نموذج وأداة في البرنامج مجموعة من الخصائص، وتُحدّد قيم هذه الخصائص في هذا الإطار كما في الشكل (٧-١-٦).



شكل ٨-١-٦: إطار المشروع ونافذة النموذج

٦ إطار المشروع:

يتيح التعامل مع نوافذ النماذج والأدوات والملفات الموجودة في البرنامج الحالي، كما في الشكل (٨-١-٦).





نافذة النموذج:


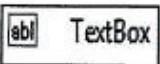
وتُبنى الواجهة بداخلها، كما في الشكل (٦-١-٨)، وذلك بإضافة الأدوات عليها.


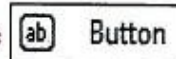
وقد يكون لدينا أكثر من نموذج في البرنامج الواحد، وهو الذي يظهر أمام المستخدم عند تنفيذ البرنامج.


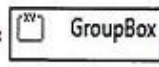
ثالثاً/ مربع الأدوات:



هناك كثير من الأدوات التي يمكن استخدامها لبناء واجهة البرنامج، وسوف نتعرف على أهمها:


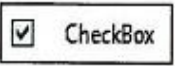
 أداة التسمية (Label)  : تستخدم لعرض نص ثابت على النموذج.



 أداة مربع النص (TextBox)  : تستخدم لإدخال البيانات حيث يقوم المستخدم بالكتابة بداخلها.

 أداة زر الأمر (Button)  : تستخدم لتنفيذ الأوامر بعد أن ينقر المستخدم عليها.

 أداة إطار التجميع (GroupBox)  : تستخدم لتجميع عدة أدوات بعضها مع بعض، وذلك برسم الإطار، ثم وضع الأدوات المراد تجميعها بداخلها.

 أداة زر الخيار (RadioButton)  : تستخدم لعرض عدة خيارات يختار المستخدم أحدها، ويفضل تجميع هذه الخيارات داخل إطار التجميع.

 أداة مربع الاختيار (CheckBox)  : تستخدم لعرض خيارات للمستخدم ليضع علامة (✓) إذا أراد اختيار أي منها أو جميعها.

 أداة مربع القائمة (ListBox)  : تستخدم لعرض قائمة من عدة خيارات يختار المستخدم أحدها.

 أداة الخانة المركبة (ComboBox)  : تستخدم لعرض قائمة مثل مربع القائمة، مع إمكانية كتابة قيمة جديدة غير الموجودة في القائمة الأصلية.

رابعاً / وضع الأدوات على النموذج:

تُوضع الأدوات على النموذج كما في الشكل (٩-١-٦)، باتباع الخطوات التالية:

١- انتقل إلى مربع الأدوات، واختر أداة التسمية بالنقر عليها بواسطة زر الفأرة الأيسر مرة واحدة.



٢- انتقل بالفأرة إلى إطار النموذج، وارسم أداة التسمية بالنقر على زر الفأرة الأيسر مع التحريك.

٣- لتعديل موقع أداة التسمية على النموذج انقر زر الفأرة الأيسر على الأداة المرسومة مع السحب إلى الموقع المرغوب.

٤- لتغيير حجم الأداة المرسومة انقر على الأداة المرسومة بواسطة زر الفأرة الأيسر مرة واحدة، ويظهر على جوانبها ووسطها مربعات تحكم في الحجم، وبالنقر على زر الفأرة الأيسر باستمرار مع سحب الفأرة تستطيع تغيير الحجم.

٥- ضع أداة التسمية في أعلى النموذج كما هو موضح في الشكل السابق (٩-١-٦).

٦- لإنشاء بقية الأدوات استخدم الخطوات من ١-٥ مع تغيير الأداة في الخطوة رقم ١.

شكل ٩-١-٦: نافذة النموذج

خامساً / إغلاق برنامج (فيجول بيسك):

يُغلق برنامج (فيجول بيسك) بأحدى الطريقتين التاليتين:

١- اختيار الأمر (Exit) من قائمة (File).

٢- النقر على الرمز (X) في يمين أعلى واجهة البرنامج.

فائدة

هناك طريقة مختصرة لرسم مربع الأداة على النموذج، وذلك بالنقر المزدوج على زر الفأرة الأيسر، فيتم رسم الأداة في النموذج.



تمريبات

س١ : ما الفرق بين (Recent Project...) و (Open Project...) ؟



Right to Left : اتجاه النص اليميني.

Name : تحديد اسم الأداة.

Text : إظهار العنوان داخل الأداة على الشاشة.

Font : تغيير نوع الخط ونمطه وحجمه.

س٢ : ما الفرق بين (ListBox) و (ComboBox) ؟

Check Box : تستخدم لعرض خيارات للمستخدم يضع علامة صح إذا

اختيار أى منها أو جميعها.

Radio Button : تستخدم لعرض عدة خيارات يختار المستخدم أحدها،

ويفضل تجميع هذه الخيارات داخل إطار التجميع.

تمريبات

س ٣: كيف يتم إظهار مربع الأدوات (ToolBox) ؟



- العنوان الأول الذي يحمل اسم Label ١ ويحتوي على كلمة تحويل درجات الحرارة.
- العنوان الثاني الذي يحمل اسم Label ٢ ويحتوي على النص الدرجة المئوية.
- العنوان الثالث الذي يحمل اسم Label ٣ ويحتوي على النص الدرجة بالفهرنهايت.
- مربع النص (Text Box) الأول الذي يحمل اسم celsius ويكون فارغا في البداية.
- مربع النص الثاني الذي يحمل اسم fah ويكون فارغا في البداية.
- زر الأمر (Command Button) الذي على اليسار ويحمل اسم Exit.
- زر الأمر الذي على اليمين ويحمل اسم Convert.



تمريبات



- قم بتغيير خصائص الاسم (Name) والعنوان (Caption) والمحتوى (Text) لمختلف العناصر لتتوافق مع المعلومات المذكورة.
- قم بتنشيط العنوان الأول Label1 وغير الخصائص Alignment و Font و Fore color ليصبح موسطاً باللون الأحمر عريض وبحجم ١٦ نقطة.
- انقر نقراً مزدوجاً على الزر Convert ثم أكتب الشفرة المناسبة لحساب درجة الحرارة بالفهرنهايت باستخدام المعادلة التالية: $F = \frac{9}{5} * C + 32$
- انقر نقراً مزدوجاً على زر الخروج Exit ثم أكمل الشفرة.
- انقر على الزر F5 لتشغيل البرنامج وتجربته عدة مرات.

The screenshot shows a Visual Studio 2010 project named 'Project1 - Form1 (Code)'. The form, titled 'Form1', has a title bar with standard Windows controls. The form's content is titled 'تحويل درجات الحرارة' (Temperature Conversion). It contains two text boxes: the first is labeled 'الدرجة المئوية' (Celsius) and contains the value '10'; the second is labeled 'الدرجة بالفهرنهايت' (Fahrenheit) and contains the value '50'. Below the text boxes are two buttons: 'Exit' and 'Convert'. The code window on the right shows the following code:

```

Private Sub Convert_Click()
    Dim C As Single
    Dim F As Single
    C = celsius.Text
    F = (9 / 5) * C + 32
    fah.Text = F
End Sub

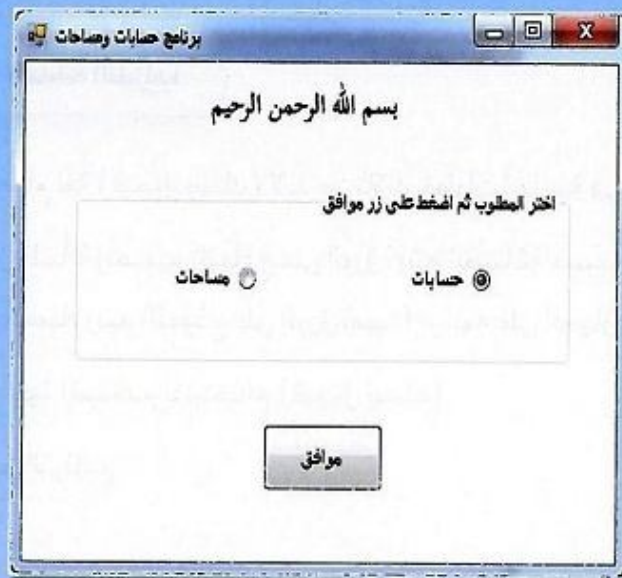
Private Sub exit_Click()
    End
End Sub

```




تمريبات

س ٤: ما الأدوات التي نحتاجها لتصميم نافذة النموذج التالي؟



رسم أداة التجميع أولاً قبل مربعات الاختيار.

رسم مربع الاختيار داخل أداة الإطار.

التدريب الثاني : مشروع برنامج (حسابات ومساحات) : تصميم نموذج المقدمة والنموذج الثاني

في هذا التدريب ستتعلم :

- إنشاء مشروع باستخدام (فيجول بيسك ٢٠١٠) .
- تصميم النماذج .
- حفظ النماذج .



متطلبات التدريب

- برنامج (فيجول بيسك ٢٠١٠) .

مقدمة التدريب

لإنشاء المشروع باستخدام لغة (فيجول بيسك) لابد من ثلاث خطوات أساسية هي:

- ١) صياغة حل المسألة وتصميم النماذج على الورق، وتتم الصياغة حسبما تعلمته في الجزء النظري، أما التصميم على الورق فمعناه رسم النموذج على الورق تمهيداً لرسمه على الجهاز باستخدام (فيجول بيسك) .
- ٢) تصميم واجهة المستخدم باستخدام (فيجول بيسك) .
- ٣) كتابة أوامر البرنامج .

وفي هذا التدريب والتدريب القادم سيتم - بإذن الله - تصميم مشروع لبرنامج (حسابات ومساحات) يحتوي على ثلاثة نماذج كالتالي:



- ١ النموذج الأول (مقدمة): يعرض هذا النموذج لوحة ترحيب مع اختيار، ويختار المستخدم أحد البرنامجين (حسابات) أو (مساحات).
- ٢ النموذج الثاني (حسابات): يدخل المستخدم عددين، ثم يحسب البرنامج العمليات الحسابية الأربع لهذين العددين (الجمع - الطرح - الضرب - القسمة) حسب اختيار المستخدم.
- ٣ النموذج الثالث (مساحات): تُحسب مساحة المستطيل وذلك بمعلومية الطول والعرض.

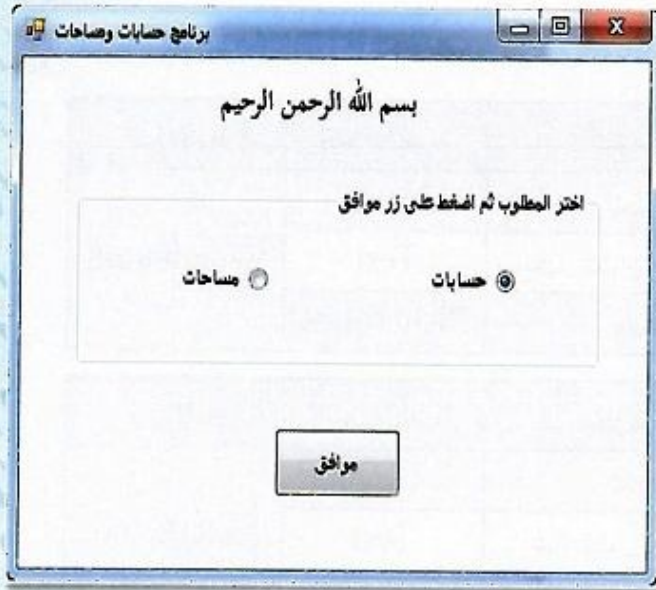
خطوات التدريب

أولاً/ التخطيط للنموذج الأول (مقدمة) :

هذا النموذج لا يحتاج إلى عملية صياغة؛ لأنه بمثابة ربط للبرنامجين (حسابات ومساحات). حيث يتمكن المستخدم من اختيار أحد البرنامجين مع عرض شاشة ترحيبية للمستخدم، والتصميم يكون كما في الشكل (١-٢-٦).

ثانياً/ أدوات وخصائص النموذج الأول:

- ١ عدّل خصائص نافذة النموذج الأول لتصبح كالتالي:



شكل ١-٢-٦: الواجهة الرئيسة للبرنامج

القيمة	الغرض من الخاصية	اسم الخاصية
Form1	تحديد اسم النموذج	Name
برنامج حسابات ومساحات	إظهار عنوان للنموذج	Text
Yes	اتجاه النص لليمين	RightToLeft

٢) أنشئ أداة التسمية (Label A) وغيّر خصائصها كالتالي:

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
Label	Name	تحديد اسم الأداة	Label1
	Text	إظهار العنوان داخل الأداة على الشاشة	بسم الله الرحمن الرحيم
	Font	تغيير نوع الخط وحجمه ونمطه	يغير الحجم إلى (١٨)

٣) أنشئ أداة إطار التجميع (☐ GroupBox) وغيّر خصائصها كالتالي:

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
GroupBox	Name	تحديد اسم الأداة	GroupBox1
	Text	إظهار العنوان داخل الأداة على الشاشة	اختر المطلوب ثم اضغط على زر (موافق)
	RightToLeft	اتجاه النص لليمين	Yes

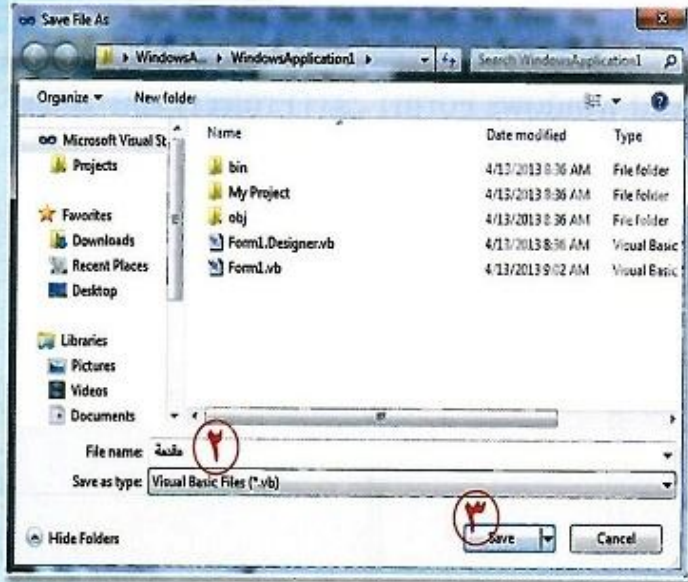
٤) أنشئ أداتين لزر الخيار (☐ RadioButton) وغيّر خصائصها كالتالي:

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
RadioButton	Name	تحديد اسم الأداة	RadioButton1
	Text	إظهار العنوان داخل الأداة على الشاشة	حسابات
	RightToLeft	اتجاه الزر لليمين	Yes

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
RadioButton	Name	تحديد اسم الأداة	RadioButton2
	Text	إظهار العنوان داخل الأداة على الشاشة	مساحات
	RightToLeft	اتجاه الزر لليمين	Yes

٥) أنشئ أداة زر الأمر (Button) وغيّر خصائصها كالتالي:

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
Button	Name	تحديد اسم الأداة	Button1
	Text	إظهار العنوان داخل الأداة على الشاشة	موافق



شكل ٢-٢-٦: حفظ النموذج

ثالثاً / حفظ النموذج الأول:

- ١ من قائمة (File) اختر الأمر (Save Form1 As...).
- ٢ تسمية النموذج (مقدمة) كما في الشكل (٢-٢-٦).
- ٣ انقر على زر (Save).

رابعاً / التخطيط للنموذج الثاني (حسابات):



شكل ٣-٢-٦: تصميم واجهة (حسابات)

١ تحليل عناصر المسألة:

- مخرجات البرنامج: ناتج العملية الحسابية.
 - مدخلات البرنامج: عددين مدخلان ولنرمز لهما (أ ، ب).
 - عمليات المعالجة: عمليات الجمع والطرح والضرب والقسمة.
 - ٢ التصميم على ورق خارجي:
- بعد معرفة مدخلات ومخرجات البرنامج يمكن لنا أن نصمم الآن واجهة البرنامج المفترضة على ورق خارجي تمهيداً لتصميمها على برنامج (فيجول بيسك) في التدريب التالي، وهذا التصميم يكون كما في الشكل (٣-٢-٦).

خامساً / البدء في تصميم النموذج الثاني (برنامج حسابات):

لإنشاء شاشة (حسابات) نحتاج إلى إنشاء نموذج جديد، وخطوات الإنشاء كالتالي:

١ من قائمة (Project) اختر (Add windows Form).

٢ تظهر نافذة (Add New Item) اختر منها (windows Form) كما في الشكل (٤-٢-٦).

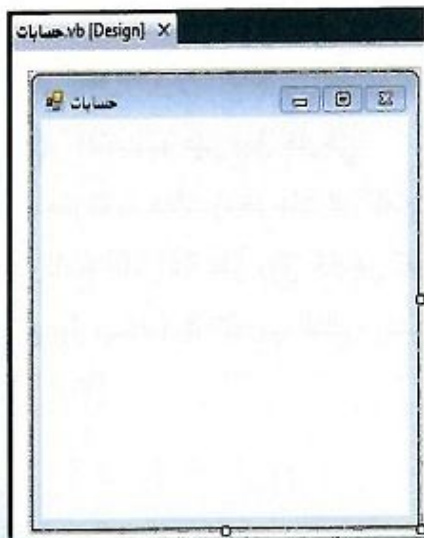
٣ اكتب اسم النموذج وليكن (حسابات).

٤ اضغط على زر (Add).



شكل ٤-٢-٦ : إنشاء نموذج جديد

٥ يتكون لديك نموذج جديد اسمه (حسابات) كما في الشكل (٥-٢-٦).



شكل ٥-٢-٦ : نموذج حسابات





سادساً / أدوات وخصائص النموذج الثاني (حسابات):

١ عدّل خواص إطار النموذج الثاني لتصبح كالتالي:

القيمة	الغرض من الخاصية	اسم الخاصية
Form2	تحديد اسم النموذج	Name
حسابات	إظهار عنوان النموذج	Text
Yes	اتجاه النص لليمين	RightToLeft

٢ أنشئ أداة التسمية (Label A) وغير خصائصها كالتالي:

القيمة	الغرض من الخاصية	اسم الخاصية	اسم الأداة
Label1	تحديد اسم الأداة	Name	Label
أدخل عددين	إظهار العنوان داخل الأداة على الشاشة	Text	

٣ أنشئ أداتي مربع النص (TextBox) وغير خصائصها كالتالي:

القيمة	الغرض من الخاصية	اسم الخاصية	اسم الأداة
TextBox1	تحديد اسم الأداة	Name	TextBox
(فارغ)	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	

القيمة	الغرض من الخاصية	اسم الخاصية	اسم الأداة
TextBox2	تحديد اسم الأداة	Name	TextBox
(فارغ)	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	

٤ أنشئ أداة إطار التجميع (☐ GroupBox) وغيّر خصائصها كالتالي:

القيمة	الغرض من الخاصية	اسم الخاصية	اسم الأداة
GroupBox1	تحديد اسم الأداة	Name	GroupBox
حدد العملية	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	



عند تجميع مربعات الاختيار داخل أداة إطار التجميع ينبغي عمل الآتي:

١ - رسم أداة إطار التجميع أولاً قبل مربعات الاختيار.

٢ - جعل مربع الاختيار داخل أداة الإطار.

٥ أنشئ أربع أدوات مربع الاختيار (☒ CheckBox) وغيّر خصائصها كالتالي:

القيمة	الغرض من الخاصية	اسم الخاصية	اسم الأداة
CheckBox1	تحديد اسم الأداة	Name	CheckBox
جمع	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه الأداة لليمين	RightToLeft	

القيمة	الغرض من الخاصية	اسم الخاصية	اسم الأداة
CheckBox2	تحديد اسم الأداة	Name	CheckBox
طرح	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه الأداة لليمين	RightToLeft	

القيمة	الغرض من الخاصية	اسم الخاصية	اسم الأداة
CheckBox3	تحديد اسم الأداة	Name	CheckBox
ضرب	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه الأداة لليمين	RightToLeft	





اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
CheckBox	Name	تحديد اسم الأداة	CheckBox4
	Text	إظهار العنوان داخل الأداة على الشاشة	قسيمة
	RightToLeft	اتجاه الأداة لليمين	Yes

٦ أنشئ أداة زر الأمر (Button) وغير خصائصها كالتالي:

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
Button	Name	تحديد اسم الأداة	Button1
	Text	إظهار العنوان داخل الأداة على الشاشة	حساب

٧ أنشئ أربع أدوات مربع النص (TextBox) وغير خصائصها كالتالي:

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
TextBox	Name	تحديد اسم الأداة	TextBox3
	Text	إظهار العنوان داخل الأداة على الشاشة	(فارغ)
	RightToLeft	اتجاه النص لليمين	Yes

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
TextBox	Name	تحديد اسم الأداة	TextBox4
	Text	إظهار العنوان داخل الأداة على الشاشة	(فارغ)
	RightToLeft	اتجاه النص لليمين	Yes

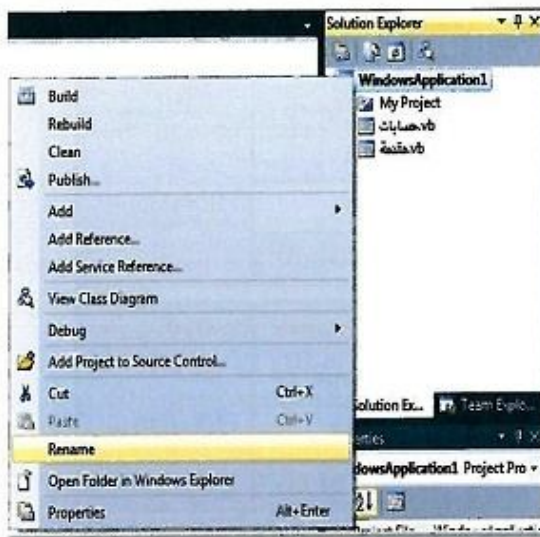
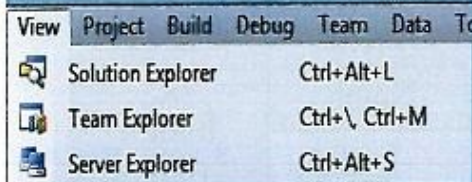
اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
TextBox	Name	تحديد اسم الأداة	TextBox5
	Text	إظهار العنوان داخل الأداة على الشاشة	(فارغ)
	RightToLeft	اتجاه النص لليمين	Yes

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
TextBox	Name	تحديد اسم الأداة	TextBox6
	Text	إظهار العنوان داخل الأداة على الشاشة	(فارغ)
	RightToLeft	اتجاه النص لليمين	Yes

فائدة

في حال عدم ظهور نافذة (Solution Explorer)،
يمكن إظهارها عن طريق:

- ١ - النقر بالزر الأيسر على قائمة (View).
- ٢ - اختيار الأمر (Solution Explorer).



شكل ٦-٢-٦: إعادة تسمية المشروع

فائدة

سيُحفظ المشروع باسم
(WindowsApplication1)

سابقاً / حفظ النموذج الثاني (حسابات):

اختر الأمر (Save) من قائمة (File).

ثامناً / إعادة تسمية المشروع:

١ انقر بالزر الأيمن على اسم المشروع في مربع
(Solution Explorer) كما في الشكل (٦-٢-٦).

٢ اختر الأمر (Rename).

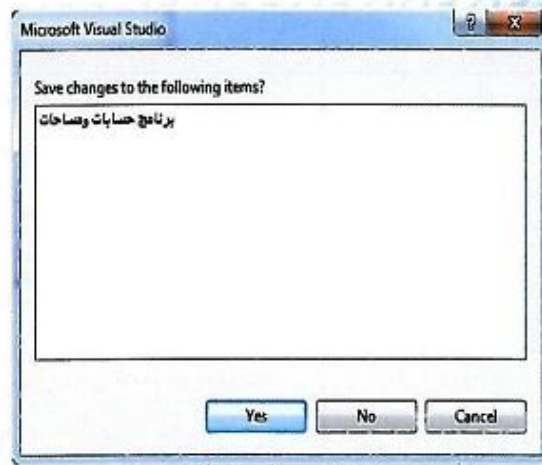
٣ اكتب الاسم الجديد للمشروع وليكن (برنامج
حسابات ومساحات).

تاسعاً / حفظ المشروع وإغلاقه:

١ من قائمة (File) اختر الأمر (Close Project).

٢ ستظهر نافذة حفظ المشروع كما في الشكل (٦-٢-٧).

٣ انقر على زر (Yes) لحفظ المشروع.



شكل ٧-٢-٦: حفظ المشروع



تمريبات

س ١: ما الغرض من الخواص التالية:

- Font
- Text
- Name
- RightToLeft



Right Left : to اتجاه النص اليمين.

Name : تحديد اسم الأداة.

Text : إظهار العنوان داخل الأداة على الشاشة.

Font : تغيير نوع الخط ونمطه وحجمه.

س ٢: ما الفرق بين الأداة (CheckBox) والأداة (RadioButton) ؟

Check Box : تستخدم لعرض خيارات للمستخدم يضع علامة

صح إذا اختار أي منها أو جميعها.

Radio Button : تستخدم لعرض عدة خيارات يختار المستخدم أحدها، ويفضل تجميع هذه الخيارات داخل إطار التجميع.

تمريبات

- س٣: صمم واجهة مستخدم رئيسة لبرنامج يقوم بالتحويل من الدرجة المئوية إلى (فهرنهايت) وكذلك العكس، وذلك عن طريق:
- تحديد الأدوات التي رسمتها على النموذج.
 - تحديد الخواص لكل أداة رسمتها على النموذج.



- العنوان الثاني الذي يحمل اسم Label٢ ويحتوي على النص الدرجة المئوية.
- العنوان الثالث الذي يحمل اسم Label٣ ويحتوي على النص الدرجة بالفهرنهايت.
- مربع النص (Text Box) الأول الذي يحمل اسم celsius ويكون فارغا في البداية.
- مربع النص الثاني الذي يحمل اسم fah ويكون فارغا في البداية.
- زر الأمر (Command Button) الذي على اليسار ويحمل اسم Exit
- زر الأمر الذي على اليمين ويحمل اسم Convert



تمرينات



Form1

تحويل درجات الحرارة

10

لدرجة المئوية

50

لدرجة بالفهرنهايت

Exit

Convert

Project1 - Form1 (Code)

exit

Click

```

Private Sub Convert_Click()
    Dim C As Single
    Dim F As Single
    C = celsius.Text
    F = (9 / 5) * C + 32
    fah.Text = F
End Sub

Private Sub exit_Click()
    End
End Sub

```




تمرينات



- قم بتغيير خصائص الاسم (Name) والعنوان (Caption) والمحتوى (Text) لمختلف العناصر لتتوافق مع المعلومات المذكورة.
- قم بتنشيط العنوان الأول Label1 وغير الخصائص Alignment و Fore color و Font ليصبح متوسطاً بالون الأحمر عريض وبحجم ١٦ نقطة.
- أنقر نقراً مزدوجاً على الزر Convert ثم أكتب الشفرة المناسبة لحساب درجة الحرارة بالفهرنهايت باستخدام المعادلة التالية: $F = (9/5) * C + 32$
- أنقر نقراً مزدوجاً على زر الخروج Exit ثم أكمل الشفرة
- لتشغيل البرنامج وتجربته عدد المرات F5 أنقر على الزر



تمريبات

س ٤: ما الخطوات اللازمة لتجميع مربعات الاختيار أو أزرار الخيار داخل إطار التجميع؟



رسم أداة التجميع أولاً قبل مربعات الاختيار.

رسم مربع الاختيار داخل أداة الإطار.

.....التدريب الثالث : مشروع برنامج (حسابات ومساحات) : تصميم النموذج الثالث

في هذا التدريب ستتعلم :

🟡 تصميم نموذج مساحات لمشروع برنامج (حسابات ومساحات).



متطلبات التدريب

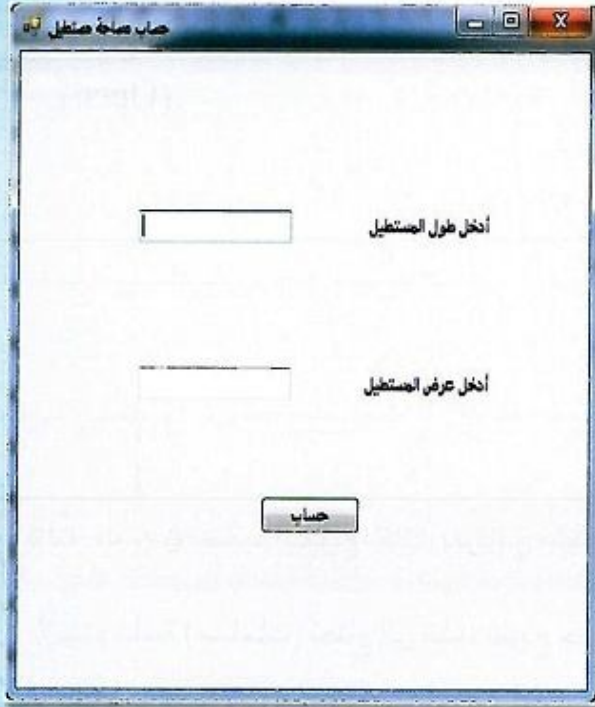
🟡 برنامج (فيجول بيسك ٢٠١٠).

🟡 مشروع (windows application) الذي تم إنشاؤه في التدريب السابق.

مقدمة التدريب

في التدريب السابق تمكنت من إنشاء الواجهة الرئيسة ونموذج حسابات لمشروع برنامج (حسابات ومساحات) وتم حفظ المشروع وإغلاقه، وفي هذا التدريب سستكمل المشروع بتصميم النموذج الثالث لحساب مساحة المستطيل.





شكل ١-٣-٦: تصميم واجهة (مساحات)

خطوات التدريب

أولاً/ التخطيط للنموذج الثالث (مساحات):

تحليل عناصر المسألة:

- ١ مخرجات البرنامج: مساحة المستطيل:
- مدخلات البرنامج: عدنان مدخلان ولنرمز لهما ب: أ (الطول)، ب (العرض).
- عمليات المعالجة: قانون إيجاد مساحة المستطيل = الطول × العرض.

٢ التصميم على ورق خارجي:

يُصمم هذا النموذج كما في الشكل (١-٣-٦).

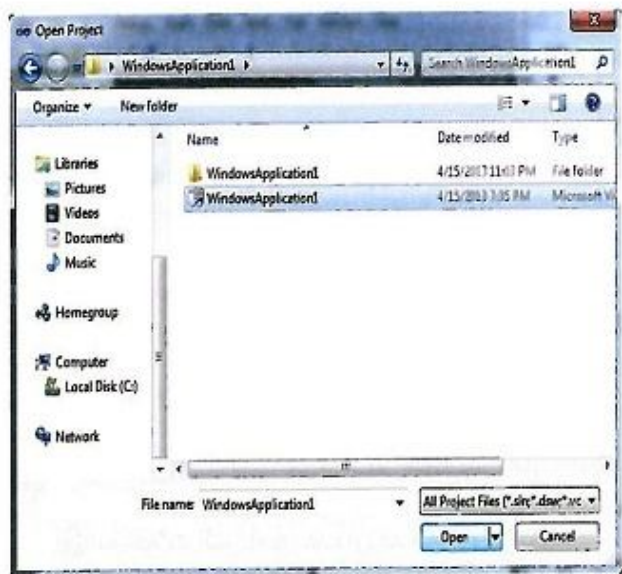
ثانياً/ فتح ملف مشروع (حسابات ومساحات):

يُفتح المشروع باستخدام إحدى الطرق التالية:

- ١ اختيار اسم المشروع (WindowsApplication1) من قائمة المشاريع الأخيرة (Recent Projects) كما في شكل (٢-٣-٦).



شكل ٢-٣-٦: الشاشة الرئيسية



٢ النقر على الأمر (Open Project) من الشاشة الرئيسية كما في شكل (٢-٣-٦)، ثم اختيار اسم المشروع، كما في الشكل (٣-٣-٦)، ثم النقر على زر (Open).

شكل ٣-٣-٦: نافذة اختيار ملف مشروع

ثالثاً/ البدء في تصميم النموذج الثالث (برنامج مساحات):

لإنشاء شاشة (مساحات) نحتاج إلى إنشاء نموذج جديد، وخطوات الإنشاء كالتالي:

- ١ اختر الأمر (Add windows Form) من قائمة (Project).
- ٢ تظهر نافذة (Add New Item) اختر منها (windows Form).
- ٣ اكتب اسم النموذج وليكن (مساحات).
- ٤ انقر على زر (Add).
- ٥ يتكون لديك نموذج جديد اسمه (مساحات).

رابعاً/ أدوات وخصائص النموذج الثالث (برنامج مساحات):

١ عدّل خصائص نافذة النموذج الثالث لتصبح كالتالي:

القيمة	الفرض من الخاصية	اسم الخاصية
Form3	تحديد اسم النموذج	Name
مساحات	إظهار عنوان النموذج	Text
Yes	اتجاه النص لليمين	RightToLeft



٢ أنشئ أداتي التسمية (Label A) وغير خصائصها كالتالي:

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
Label	Name	تحديد اسم الأداة	Label1
	Text	إظهار العنوان داخل الأداة على الشاشة	أدخل طول المستطيل
	RightToLeft	اتجاه النص لليمين	Yes

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
Label	Name	تحديد اسم الأداة	Label2
	Text	إظهار العنوان داخل الأداة على الشاشة	أدخل عرض المستطيل
	RightToLeft	اتجاه النص لليمين	Yes

٣ أنشئ أداتي مربع النص (TextBox ab) وغير خصائصها كالتالي:

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
TextBox	Name	تحديد اسم الأداة	TextBox1
	Text	إظهار العنوان داخل الأداة على الشاشة	(فارغ)
	RightToLeft	اتجاه النص لليمين	Yes

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
TextBox	Name	تحديد اسم الأداة	TextBox2
	Text	إظهار العنوان داخل الأداة على الشاشة	(فارغ)
	RightToLeft	اتجاه النص لليمين	Yes

٤ أنشئ أداة زر الأمر (Button ab) وغير خصائصها كالتالي:

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
Button	Name	تحديد اسم الأداة	Button1
	Text	إظهار العنوان داخل الأداة على الشاشة	حساب

٥ أنشئ أداة مربع النص (TextBox) وغير خصائصها كالتالي:

اسم الأداة	اسم الخاصية	الفرص من الخاصية	القيمة
TextBox	Name	تحديد اسم الأداة	TextBox3
	Text	إظهار العنوان داخل الأداة على الشاشة	(فارغ)
	RightToLeft	اتجاه النص لليمين	Yes


خامساً/ حفظ النموذج الثالث (مساحات):

اختر الأمر (Save) من قائمة (File).

سادساً/ حفظ المشروع وإغلاقه:

احفظ المشروع وأغلقه.

فائدة

يمكن أيضاً إغلاق المشروع من خلال إغلاق برنامج (فيجول بيسك) بالضغط على زر  بعد ذلك حفظ المشروع إن لزم الأمر

تمارين

س ١: ما الفرق بين المشروع والنموذج والخاصية؟

المشروع: يتيح التعامل مع نوافذ النماذج والأدوات في الملفات الموجودة في البرنامج الحالي.

النموذج: تبنى الواجهة بداخلها وذلك بإضافة الواجهات عليها وقد يكون لدينا أكثر من نموذج في البرنامج الواحد، وهو الذي يظهر أمام المستخدم عند تنفيذ البرنامج.

الخاصية: لكل نموذج وأداة في البرنامج مجموعة من الخصائص، وتحدد قيم هذه الخصائص في هذا الإطار.



تمريبات

س ٢: صمم واجهة مستخدم لبرنامج يختار المستخدم اسم دولة من قائمة منسدلة

بعد ذلك يعرض عاصمة تلك الدولة، وذلك عن طريق:

أ - رسم نموذج للواجهة.

ب - تحديد الأدوات التي رسمتها على النموذج.

ج - تحديد الخواص لكل أداة رسمتها على النموذج.



Form1

اختر الدولة

السعودية
مصر
السودان
البحرين
الإمارات

العاصمة

تمرينات



القيمة	اسم الخاصية	اسم الأداة
اختر الدولة	Text	label
العاصمة	Text	label
"فارغ"	Text	label
"أسماء الدول"	Items	ListBox

تمرينات

س ٣: ما خطوات إنشاء نموذج جديد في (فيجول بيسك) ؟

من قائمة **Project** نختار **Add Windows Form**
تظهر لنا نافذة **Add New Items** اختر منها **Windows Form**
اكتب اسم النموذج
اضغط على زر **Add** يتكون لديك نموذج جديد.

س ٤: ما الفرق بين أداة (Label) وأداة (Textbox) ؟

Label : تستخدم لعرض نص ثابت على النموذج.
Textbox : يستخدم لإدخال البيانات حيث يقوم المستخدم بكتابة بداخلها.





..... التدريب الرابع : مشروع برنامج (حسابات ومساحات) : كتابة الأوامر البرمجية لنموذج (حسابات)

في هذا التدريب ستتعلم :

● الدخول لشاشة البرمجة .

● كتابة أوامر البرمجة .

● تنفيذ البرنامج .



متطلبات التدريب

● برنامج (فيجول بيسك ٢٠١٠) .

● مشروع (windows application) الذي تم إنشاؤه في التدريب السابق.

مقدمة التدريب

في هذا التدريب ستقوم بكتابة أوامر نموذج (حسابات) لمشروع (حسابات ومساحات) الذي سبق تصميم واجهة المستخدم له، وعند كتابة هذه الأوامر لابد أن تتعرف على شاشة البرمجة، التي من خلالها تستطيع كتابة أوامر هذا البرنامج.

خطوات التدريب

أولاً/ فتح النموذج الثاني (حسابات) :



شكل ٦-٤-١ : نافذة محتوى المشروع

١ افتح مشروع (حسابات ومساحات) الذي أنشئ في التدريبات السابقة وحفظ باسم (مشروع).

٢ اختر النموذج الثاني (حسابات)، بالنقر المزدوج عليه من محتوى المشروع كما في الشكل (٦-٤-١).

٣ تظهر شاشة نموذج (حسابات)، التي تتكون من أدوات أنشئت في التدريبات السابقة.

٤ اكتب الأوامر المتعلقة بهذا النموذج، التي تقوم بإدخال عددين في أداتي مربع النص، وباختيار العملية الحسابية المطلوبة من أداة مربع الاختيار تظهر النتيجة في أدوات التسمية الأخرى.

٥ قبل الانتقال لكتابة الأوامر لابد من معرفة المكان الذي سوف يحتوي هذه الأوامر وهي شاشة البرمجة، فكيف يتم الدخول إلى هذه الشاشة والتعامل معها؟

ثانياً/ الدخول إلى شاشة البرمجة :

فائدة

يمكن أيضاً الدخول إلى شاشة البرمجة عن طريق تحديد الأداة ثم الضغط على مفتاح (F7).

١ النقر المزدوج على الأداة التي تنفذ الأمر مثلاً زر (حساب).

٢ تظهر شاشة البرمجة كما في الشكل (٦-٤-٢)، وتتكون هذه الشاشة من ثلاثة أجزاء:

أ. مربع سرد الأدوات المرتبطة بالنموذج: حيث يمكن تحديد الأداة المراد كتابة الأوامر فيها.





ب - مربع سرد الإجراءات الموجودة في (فيجول بيسك): حيث يمكن تحديد الإجراء المناسب للأداة التي تم اختيارها في مربع سرد الأدوات.

ت - منطقة البرمجة: ويكتب فيها أوامر (فيجول بيسك)، وتحتوي على:

Public Class): التعريف العام لطبقة النموذج.

Private Sub): الإجراء الفرعي داخل البرنامج، وهو ثابت لجميع الإجراءات.

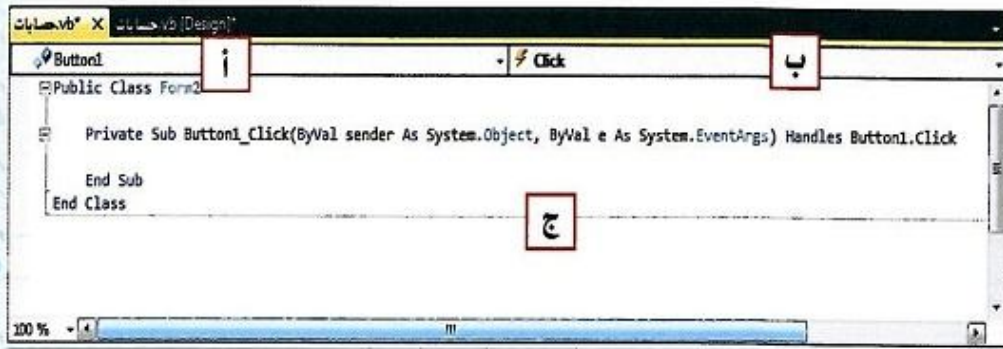
(الإجراء __ اسم الأداة): اسم الأداة المراد التعامل معها مثل: (Button1)، والإجراء الخاص بالأداة

مثل: (Click)، والشرطة السفلية (__) تربط بين اسم الأداة والإجراء.

السطر الفراع بعد (Private Sub): لكتابة أوامر البرمجة وتنفيذ تلك الأوامر عند تنفيذ الإجراء.

(End Sub): نهاية الإجراء الفرعي.

(End Class): نهاية التعريف العام للطبقة.



شكل ٦-٤-٢: شاشة البرمجة ومكوناتها

ثالثاً/ كتابة أوامر النموذج الثاني (حسابات):

تُكتب أوامر هذا البرنامج كما في الشكل (٦-٤-٢) والتي تحقق:

ادخال البيانات في مربعي النص (Text1 و Text2).

تحديد عملية حسابية أو أكثر من خلال اختيار مربعات الاختيار

(CheckBox1, CheckBox2, CheckBox3, CheckBox4).

المعالجة التي تجري على العمليات لإظهار المخرجات.

الحصول على مخرجات البرنامج من خلال أدوات مربعات النصوص

(Text4 ,Text5 ,Text6 Text3)

```
Public Class Form2
```

```
Private Sub Button1__Click()
```

```
Dim A, B As Single
```

```
A = TextBox1.Text
```

```
B = TextBox2.Text
```

```
If CheckBox1.Checked = True Then TextBox3.Text = "حاصل الجمع = "& A + B
```

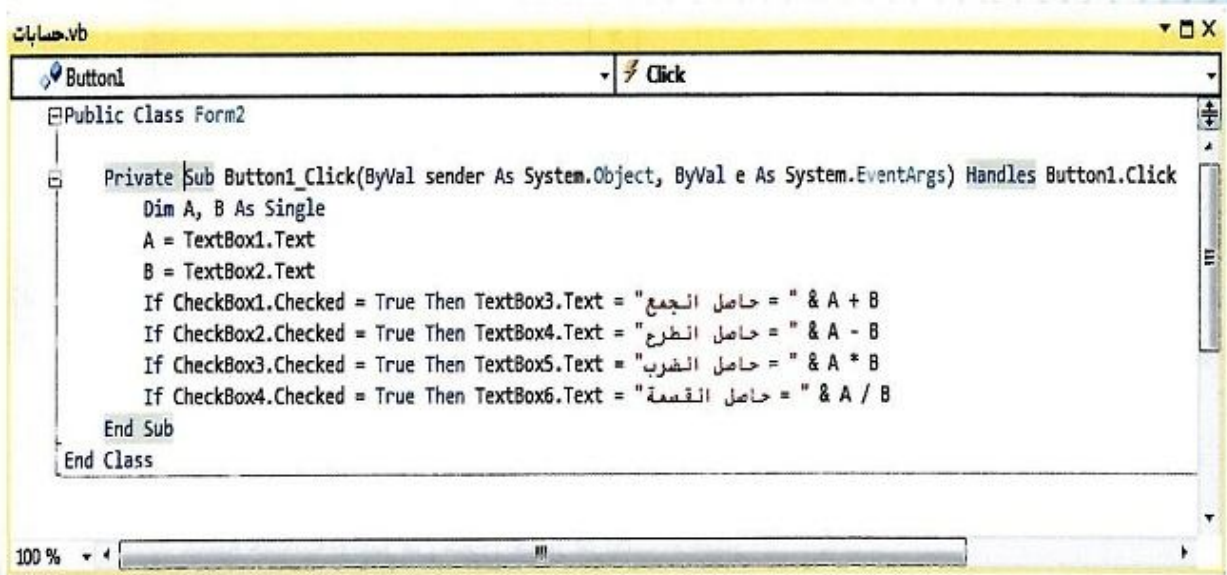
```
If CheckBox2.Checked = True Then TextBox4.Text = "حاصل الطرح = "& A - B
```

```
If CheckBox3.Checked = True Then TextBox5.Text = "حاصل الضرب = "& A * B
```

```
If CheckBox4.Checked = True Then TextBox6.Text = "حاصل القسمة = "& A / B
```

```
End Sub
```

```
End Class
```



شكل ٦-٤-٣: أوامر برنامج (حسابات)



شرح البرنامج :

السطر الثاني: عندما يتم النقر على زر الأمر نقرة واحدة نفذ الأسطر التالية:

السطر الثالث: تعريف المتغيرات المستخدمة في البرنامج وهي: (A للعدد الأول) و (B للعدد الثاني)، وهذا السطر اختياري.

السطر الرابع: الحصول على البيانات المدخلة في مربع النص (Textbox1) وتخزينها في المتغير (A).

السطر الخامس: الحصول على البيانات المدخلة في مربع النص (Textbox2) وتخزينها في المتغير (B).

السطر السادس: شرط يوضح إن كان المستخدم قد اختار مربع الاختيار الخاص بالجمع والمسمى (Checkbox1)، فعندئذ يقوم البرنامج بطباعة حاصل جمع العددين في أداة مربع النص المسمى (Textbox3).

السطر السابع: شرط يوضح إن كان المستخدم قد اختار مربع الاختيار الخاص بالطرح والمسمى (Checkbox2)، فعندئذ يقوم البرنامج بطباعة حاصل طرح العددين في أداة مربع النص المسمى (Textbox4).

السطر الثامن: شرط يوضح إن كان المستخدم قد اختار مربع الاختيار الخاص بالضرب والمسمى (Checkbox3)، فعندئذ يقوم البرنامج بطباعة حاصل ضرب العددين في أداة مربع النص المسمى (Textbox5).

السطر التاسع: شرط يوضح إن كان المستخدم قد اختار مربع الاختيار الخاص بالقسمة والمسمى (Checkbox4)، فعندئذ يقوم البرنامج بطباعة حاصل قسمة العددين في أداة مربع النص المسمى (Textbox6).

السطر العاشر: نهاية الإجراء.

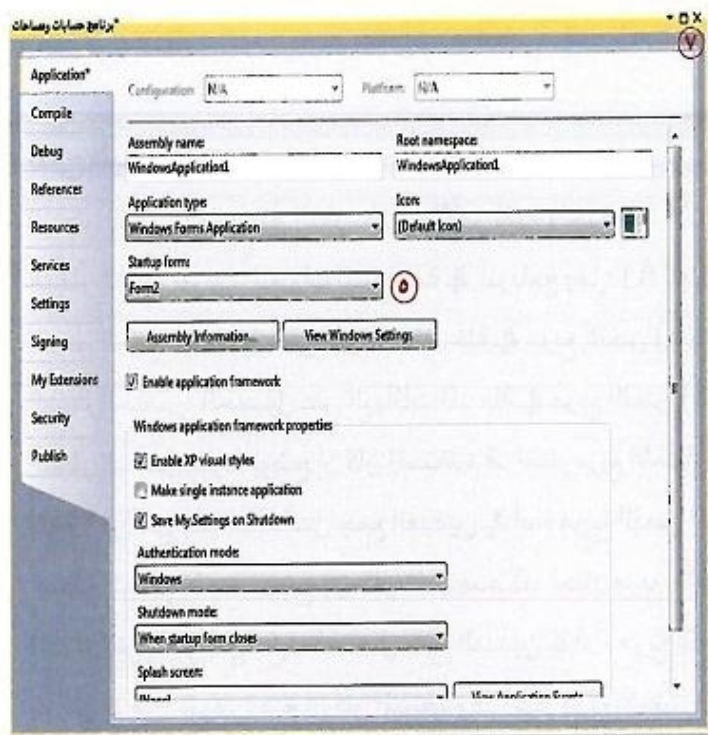
رابعاً/ تنفيذ برنامج (حسابات) :

فائدة

يمكن تنفيذ البرنامج باختيار الأمر (Start Debugging) من قائمة (Debug) أو من خلال الضغط على مفتاح (F5)

١ انقر على زر التنفيذ (▶) من شريط الأدوات القياسي، ماذا تلاحظ؟ (تم تنفيذ الشاشة الأولى وهي (Form1) شاشة المقدمة، ذلك لأن (فيجول بيسك) ينفذ أول شاشة تم إنشاؤها).

٢ أوقف تنفيذ البرنامج بالنقر على زر التوقف (■) في شريط الأدوات القياسي.



شكل ٤-٤-٦ : نافذة اختيار النموذج للتنفيذ

٣ اختر الأمر (Properties) من قائمة (Project).

٤ تظهر لك نافذة (Application) كما في الشكل (٤-٤-٦).

٥ انقر السهم المتجه لأسفل في قائمة (Startup Form) ليعرض كل النماذج.

٦ اختر النموذج الثاني (Form2).

٧ أغلق نافذة (Application) بالضغط على زر (X) الموجود بأعلىها، وبذلك تم تبديل التنفيذ للنموذج الثاني.

٨ نفذ البرنامج مرة أخرى ولاحظ أنه تم تنفيذ النموذج الثاني (حسابات).

خامساً / إيقاف تنفيذ برنامج (حسابات) :

انقر على زر التوقف () في شريط الأدوات القياسي.

سادساً / حفظ برنامج (حسابات) :

احفظ برنامج (حسابات) وأغلق مشروع (حسابات ومسابقات).

فائدة

يمكن إيقاف تنفيذ البرنامج باختيار الأمر (Stop Debugging) من قائمة (Debug) أو النقر على الرمز (X) في أعلى يمين نافذة تنفيذ نموذج (حسابات)



تمارين



س ١: ما المقصود بالأوامر التالية:

• End Sub • Dim grade As Integer • Private Sub Button5_Click ()

Private Sub Button5_click: الإجراء الفرعي داخل البرنامج، وهو ثابت لجميع الإجراءات.

Dim grade As Integer: تعريف المتغيرات المستخدمة في البرنامج.

End Sub: نهاية الإجراء الفرعي.

س ٢: ما الفرق بين الأمرين التاليين:

M = TextBox1.text

TextBox1.text = M

الأمر **M=Textbox1.text**

سوف يقوم بتخزين محتويات صندوق الإدخال في المتغير **M**

أما الأمر **TextBox1.text=M**

سوف يقوم باستدعاء محتويات المتغير **M** وإظهارها في صندوق الإدخال.



تمارينات



س٣: اشرح معنى كل سطر في البرنامج التالي:

Dim Grade As Integer, Mark As Single

Grade = 10

Mark = InputBox("أدخل درجة الطالب")

If mark >= 60 Then Grade = Grade + 1 Else Grade = Grade

MsgBox(Grade)

EndIf

Dim Grade As integer, Mark As single

Grade = 10

تخزين القيمة ١٠ في المتغير

Mark = InputBox ("أدخل درجة الطالب")

إظهار رسالة تطلب درجة الطالب وتخزينها في متغير

If mark >= 60 then Grade = Grade+1 Else Grade = Grade

إذا كانت الدرجة أكبر من أو تساوي ٦٠ سوف يقوم بزيادة الدرجة بمقدار ١ وغير

ذلك فتضل الدرجة كما هي

MsgBox(Grade)

إظهار رسالة بالدرجة

EndIf



تمارين



س٤ : اكتب الأوامر لبرنامج يقوم بالتحويل من الريال السعودي إلى الريال العماني .

```
Dim rs,ro As Long
```

```
rs = TextBox1.Text
```

```
ro = rs * 9.74
```

```
Label1.Text = ro
```


.....التدريب الخامس : مشروع برنامج (حسابات ومساحات) :

كتابة الأوامر البرمجية لنموذج (مساحات) واستكمال المشروع

في هذا التدريب ستتعلم :

- فتح النموذج الثالث (مساحات) وكتابة الأوامر الخاصة به.
- ربط النموذج الثاني (حسابات) والنموذج الثالث (مساحات) بالنموذج الأول (مقدمة).
- تنفيذ المشروع.
- بناء ملف ذاتي التنفيذ لمشروع (حسابات ومساحات).

متطلبات التدريب

- برنامج (فيجول بيسك ٢٠١٠).
- مشروع (windows application) الذي تم إنشاؤه في التدريب السابق.

مقدمة التدريب

في هذا التدريب ستقوم بكتابة أوامر نموذج (مساحات) لمشروع برنامج (حسابات ومساحات)، الذي سبق تصميم واجهة المستخدم له، وكذلك تنفيذ المشروع للتأكد من خلوّه من الأخطاء، بالإضافة إلى تحويل المشروع إلى ملف ذاتي التنفيذ بصيغة (.exe)، أي سيتم تنفيذ البرنامج مباشرة دون الحاجة إلى تشغيله بواسطة (فيجول بيسك).



خطوات التدريب

أولاً/ فتح النموذج الثالث (حسابات) :



شكل ١-٥-٦ : نافذة محتوى المشروع

- ١ افتح مشروع (حسابات ومساحات).
- ٢ اختر النموذج الثالث (مساحات)، بالنقر المزدوج عليه من محتوى المشروع كما في الشكل (١-٥-٦).
- ٣ تظهر شاشة نموذج (مساحات)، التي تتكون من أدوات أنشئت في التدريبات السابقة.
- ٤ انقر نقراً مزدوجاً على زر الأمر (Button1) المسمى (حساب) للدخول إلى شاشة البرمجة.
- ٥ اكتب الأوامر المتعلقة بهذا النموذج، التي تقوم بإدخال الطول والعرض للمستطيل في أداتي مربع النص، وباختيار زر الأمر (حساب) لتظهر النتيجة في أداة مربع النص الثالث.

ثانياً/ كتابة أوامر النموذج الثالث (مساحات) :

تُكتب أوامر هذا البرنامج كما في الشكل (٢-٥-٦)، التي تحقق:

- ١ الحصول على طول وعرض المستطيل من مربعي النص (TextBox1 و TextBox2).
- ٢ المعالجة التي تجري على العمليات لإظهار المخرجات بواسطة مربع النص (TextBox3) عند النقر على زر الأمر (Button1).


```
Public Class Form3
```

```
Private Sub Button1__Click()
```

```
Dim A, B As Single
```

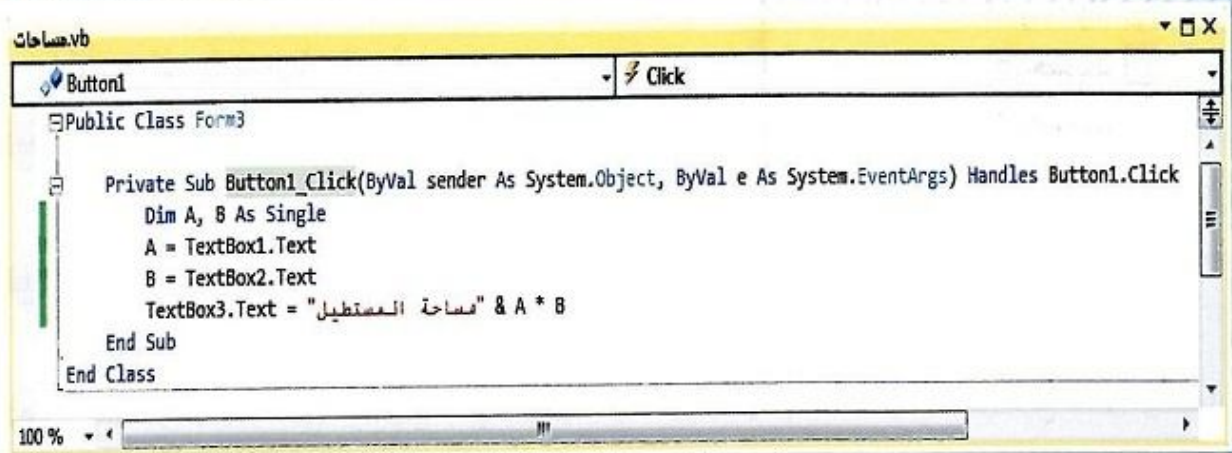
```
A = TextBox1.Text
```

```
B = TextBox2.Text
```

```
TextBox3.Text = "مساحة المستطيل" & A * B
```

```
End Sub
```

```
End Class
```



شكل ٦-٥-٢: أوامر برنامج (مساحات)

شرح البرنامج :

السطر الثاني: عندما يتم النقر على زر الأمر نقرة واحدة نفذ الأسطر التالية:

السطر الثالث: تعريف المتغيرات المستخدمة في البرنامج وهي: (A لطول المستطيل) و (B لعرض المستطيل).

السطر الرابع: الحصول على البيانات المدخلة في مربع النص (TextBox1) وتخزينها في المتغير (A).

السطر الخامس: الحصول على البيانات المدخلة في مربع النص (TextBox2) وتخزينها في المتغير (B).

السطر السادس: طباعة مساحة المستطيل، التي تساوي (الطول × العرض) في أداة مربع النص (TextBox3).

السطر السابع: نهاية الإجراء.



ثالثاً/ تنفيذ برنامج (مساحات) :

- ١- بدل التنفيذ للنموذج (مساحات) المسمى (Form3) من خلال الأمر (Properties) في قائمة (Project).
- ٢- نفذ البرنامج بالنقر على زر التنفيذ (▶) من شريط الأدوات القياسي.

رابعاً/ إيقاف تنفيذ برنامج (مساحات) :

انقر على زر التوقف (■) في شريط الأدوات القياسي.

خامساً/ حفظ برنامج (مساحات) :

احفظ برنامج (مساحات).

سادساً/ ربط النماذج بالنموذج الأول (مقدمة) :

عندما صممت نموذجك (حسابات) و (مساحات) ووضعت الأوامر الخاصة بهما قمت بتنفيذ كل برنامج على حدة، وكنت في بداية التخطيط للبرنامج وضعت شاشة مقدمة تجعل المستخدم يختار بين تنفيذ أحد البرنامجين وأسميتها (مقدمة). ولكن كيف يتم الربط بين هذا النموذج ونموذجي (حسابات) و (مساحات)؟

لعمل هذا الربط اتبع الخطوات التالية:

- ١- اختر النموذج الأول (مقدمة) بالنقر المزدوج عليه من محتوى المشروع.
- ٢- انقر نقراً مزدوجاً على زر الأمر (Button1) المسمى (موافق) للدخول إلى شاشة البرمجة.
- ٣- اكتب أوامر هذا النموذج كما في الشكل (٦-٥-٣)، التي تحقق عمليات المعالجة لإتمام عملية الربط.


```
Public Class Form1
```

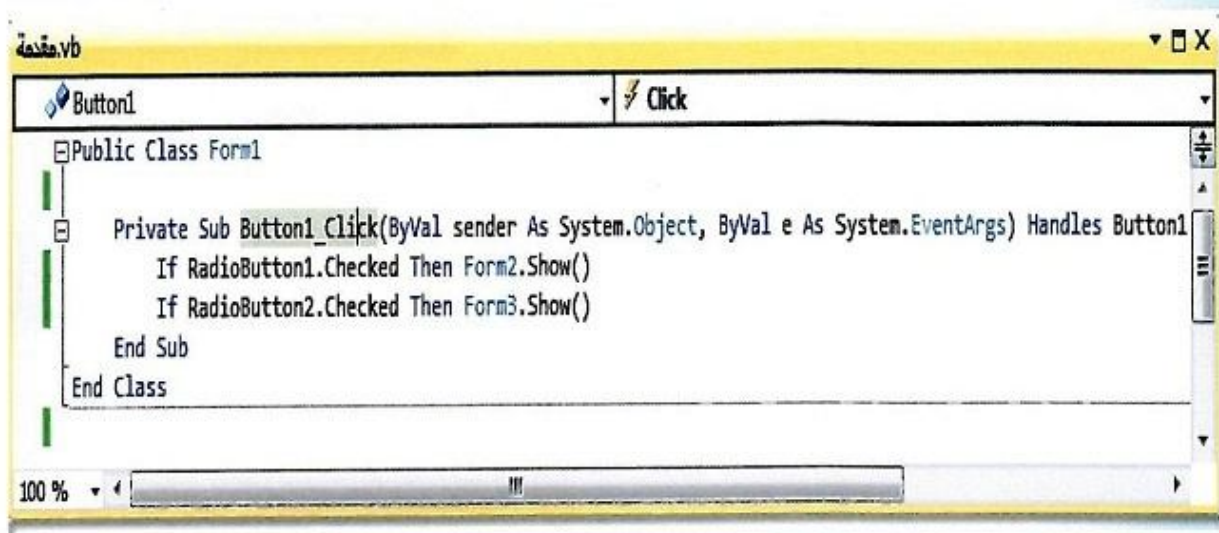
```
Private Sub Button1__Click() Handles Button1.Click
```

```
    If RadioButton1.Checked Then Form2.Show()
```

```
    If RadioButton2.Checked Then Form3.Show()
```

```
End Sub
```

```
End Class
```



شكل ٦-٥-٣: أوامر نموذج (المقدمة)

شرح البرنامج :

السطر الثاني : عندما يتم النقر على زر الأمر نقرة واحدة نفذ الأسطر التالية:

السطر الثالث: شرط أنه إذا اختار المستخدم زر الخيار (RadioButton1) الخاص ببرنامج (حسابات) ، فسيتم عرض النموذج الثاني (حسابات) المسمى (Form2).

السطر الرابع: شرط أنه إذا اختار المستخدم زر الخيار (RadioButton2) الخاص ببرنامج (مساحات) ، فسيتم عرض النموذج الثاني (مساحات) المسمى (Form3).

السطر الخامس: نهاية الإجراء.



سابعاً / حفظ برنامج المقدمة (مقدمة) :

احفظ برنامج (مقدمة).

ثامناً / تنفيذ مشروع (حسابات ومساحات) :

١) بديل التنفيذ للنموذج (مقدمة) المسمى (Form1) من خلال الأمر (Properties) في قائمة (Project).

٢) نفذ البرنامج بالنقر على زر التنفيذ (▶) من شريط الأدوات القياسي.

تاسعاً / تحويل مشروع (حسابات ومساحات) إلى برنامج ذاتي التنفيذ :

١) اختر الأمر (Properties) من قائمة (Project)، فتظهر نافذة خصائص المشروع كما في الشكل (٦-٥-٤).

٢) اختر التبويب (Compile).

٣) من القائمة المنسدلة (Configuration) اختر (Release)، وهذا يعني أن النسخة التنفيذية للبرنامج ستكون نسخة إصدار.

٤) حدد مكان تخزين ملف التشغيل بالنقر على زر (Browse...) الخاص بالخيار (Build output path).

واختار المجلد الذي سيتم تخزين الملف التشغيلي بداخله.

٥) أغلق نافذة خصائص المشروع.

٦) اختر الأمر (Build) من قائمة (Build) مع ملاحظة

وجود اسم المشروع مقابل الأمر.

فائدة

في حال عدم تغيير مكان المجلد فإن البرنامج سيخزن الملف في الموقع الافتراضي (bin\Release) الموجود في المجلد الخاص بالمشروع.



شكل ٦-٥-٤ : نافذة تحديد خيارات التنفيذ الذاتي

٧ سينشئ ملف تنفيذي يكون اسمه (WindowsApplication1.exe)، بالإضافة إلى ملفات أخرى داخل المجلد الذي حددته في الخطوة (٤).

٨ أغلق برنامج (فيجول بيسك) بالنقر على الرمز (X) أو اختر الأمر (Exit) من قائمة (File).

٩ استعرض المجلد الذي تم خزن فيه تشغيل برنامج (حسابات ومساحات) عن طريق مستكشف الملفات والمجلدات في نظام التشغيل.

١٠ انقر نقرًا مزدوج على الملف (WindowsApplication1.exe) ليتم تشغيل برنامج (حسابات ومساحات).

١١ أغلق برنامج (حسابات ومساحات).

وبهذا تكون قد انتهيت من مشروع برنامج (حسابات ومساحات).



تمارينات



س ١: إذا كان لدينا مشروع يحتوي على ثلاثة نماذج، النموذج الأول (F1)، والنموذج الثاني (F2)، والنموذج الثالث (F3)، ونريد أن يقوم برنامج (فيجول بيسك) بتنفيذ النموذج الثالث (F3) أولاً، ما الخطوات التي نتخذها لكي يقوم (فيجول بيسك) بذلك؟

- ١ - انقر على زر التنفيذ من شريط الأدوات القياسي
- ٢ - اوقف تنفيذ البرنامج بالنقر على زر التوقف في شريط الأدوات القياسي.
- ٣ - اختر الأمر **properties** من قائمة **Project**.
- ٤ - تظهر لك قائمة **Application**.
- ٥ - انقر السهم المتجه لأسفل في قائمة **start Form** ليعرض كل النماذج.
- ٦ - اختر النموذج الثالث **Form3**.
- ٧ - أغلق نافذة **Application**.
- ٨ - نفذ البرنامج مرة أخرى وستلاحظ أنه تم تنفيذ النموذج الثالث.



تمارينات



س ٢: اكتب أوامر برنامج بلغة (فيجول بيسك) لإدخال وقت بصيغة (٢٤) ساعة وتحويلها إلى صيغة (١٢) ساعة وإحافها بعبارة "صباحاً" أو "مساءً".

```
Dim a As Long
a = TextBox1.Text
If a < 12 Then
    Label1.Text = a.ToString + ":00 صباحاً"
Else
    a = a - 12
    Label1.Text = a.ToString + ":00 مساءً"
EndIf
```


تمرينات



س ٣: ما الخطوات اللازمة لتحويل مشروع (فيجول بيسك) إلى برنامج ذاتي التشغيل؟

- ١- اختر الأمر **Properties** من قائمة **Project**، فتظهر نافذة خصائص المشروع.
- ٢- اختر التبويب **Compile**
- ٣- من القائمة المنسدلة **Configuration** اختر **Release**، وهذا يعني أن النسخة التنفيذية ستكون نسخة إصدار.
- ٤- حدد مكان تخزين ملف التشغيل بالنقر على **Browse** الخاص بالخيار **Build out**
- ٥- اضغط زر **put ath** الخاص بالخيار المجلد الذي سيتم تخزين ملف التشغيل بداخله.
- ٥- أغلق نافذة خصائص المشروع.
- ٦- اختر الأمر **Build** من قائمة **Build** مع ملاحظة وجود اسم المشروع مقابل الأمر.

.....التدريب السادس : تطبيق تحويل درجة الحرارة

في هذا التدريب ستتعلم :

● تصميم برنامج بلغة (فيجول بيسك ٢٠١٠) درجة الحرارة من المئوي إلى (فهرنهايتي).



متطلبات التدريب

● برنامج (فيجول بيسك ٢٠١٠).

مقدمة التدريب

في هذا التدريب والتدريبات التالية سنتطرق إلى تطبيقات متنوعة حتى تتمرس على تصميم واجهة المستخدم وكتابة أوامر البرنامج، وقد بدأنا في التدريبات السابقة بإنشاء تطبيقات بسيطة (حسابات ومساحات)، ثم ننقل بصورة تدريجية إلى تطبيقات أكثر عمقاً وتتطلب فهماً أكثر.

وفي هذا التدريب ستقوم بكتابة برنامج باستخدام (فيجول بيسك) يقوم بتحويل درجة الحرارة التي يدخلها المستخدم بالمئوي إلى درجة الحرارة بالفهرنهايتي، ودرجة الحرارة بالفهرنهايتي إلى درجة الحرارة بالمئوي، علماً أن:

$$\begin{aligned} \text{درجة الحرارة بالفهرنهايتي} &= (9 \div 5 \times \text{درجة الحرارة بالمئوي}) + 32 \\ \text{درجة الحرارة بالمئوي} &= (5 \div 9 \times (\text{درجة الحرارة بالفهرنهايتي} - 32)) \end{aligned}$$



خطوات التدريب

أولاً/ التصميم الورقي :

١ صياغة حل المسألة :

حلّ عناصر المسألة، وذلك بتحديد التالي:

مخرجات البرنامج: درجة الحرارة بالفهرنهايت أو بالمئوي حسب اختيار المستخدم.

مدخلات البرنامج: درجة الحرارة بالمئوي، أو بالفهرنهايت ولنرمز لها بـ (D).

عمليات المعالجة: قانون التحويل:

$$\bullet \text{ درجة الحرارة بالفهرنهايت} = 32 + (D \times 5 \div 9)$$

$$\bullet \text{ درجة الحرارة بالمئوي} = (32 - D) \times 9 \div 5$$

٢ كتابة الخطوات الخوارزمية للمسألة :

اكتب الخطوات الخوارزمية كما يلي:

(أ) اختر عملية التحويل بالمئوي أو بالفهرنهايت.

(ب) أدخل درجة الحرارة (D).

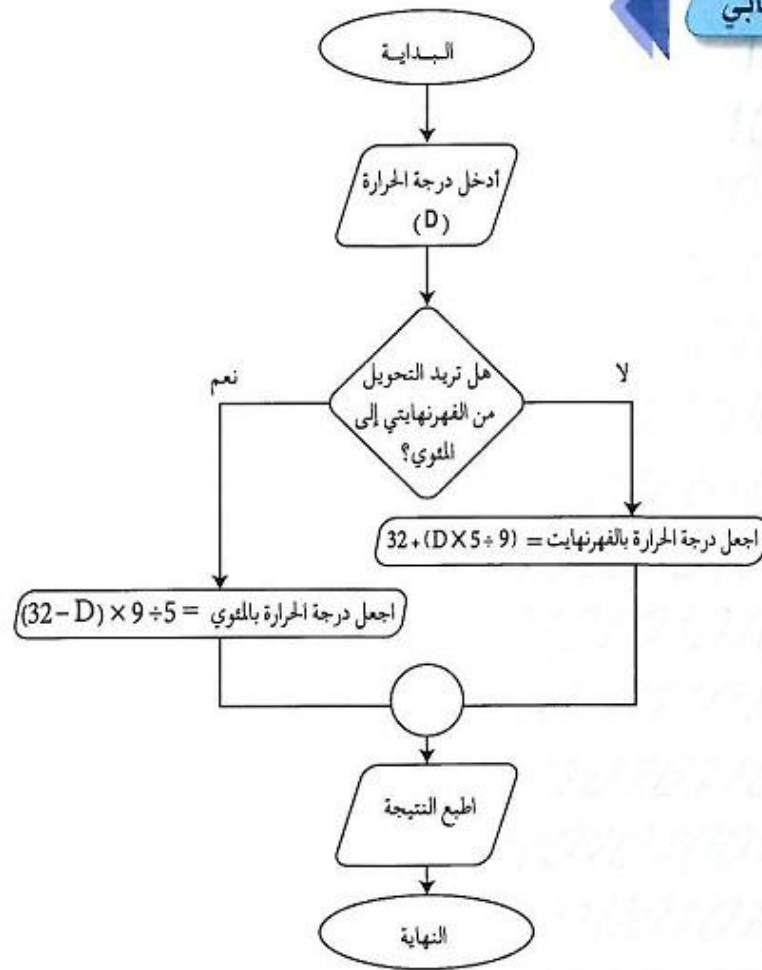
(ج) احسب عملية التحويل حسب اختيار المستخدم:

$$\bullet \text{ اجعل درجة الحرارة بالفهرنهايت} = 32 + (D \times 5 \div 9)$$

$$\bullet \text{ أو اجعل درجة الحرارة بالمئوي} = (32 - D) \times 9 \div 5$$

(د) اطبع النتيجة.

(هـ) النهاية.

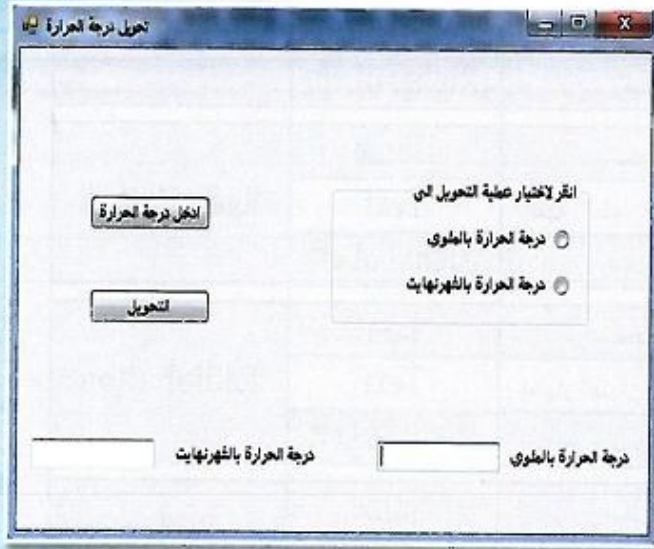


ثانياً/ أدوات وخصائص النموذج :

١- افتح برنامج (فيجول بيسك) وأنشئ مشروعاً جديداً.

٢- اضبط خصائص نموذج برنامج تحويل درجة الحرارة وفق الجدول التالي:

القيمة	الفرض من الخاصية	اسم الخاصية
Form1	تحديد اسم النموذج	Name
تحويل درجة الحرارة	إظهار عنوان النموذج	Text
Yes	اتجاه النص لليمين	RightToLeft



٣ أنشئ الأدوات على النموذج، كما في الشكل (٦-٦-١)، واضبط خصائص كل أداة كما في الجدول التالي:

شكل ٦-٦-١: نموذج تحويل درجة الحرارة

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
Label	Name	تحديد اسم الأداة	Label1
	Text	إظهار العنوان داخل الأداة على الشاشة	درجة الحرارة بالمئوي
	RightToLeft	اتجاه النص لليمين	Yes
Label	Name	تحديد اسم الأداة	Label2
	Text	إظهار العنوان داخل الأداة على الشاشة	درجة الحرارة بالفهرنهايت
	RightToLeft	اتجاه النص لليمين	Yes
Textbox	Name	تحديد اسم الأداة	Textbox1
	Text	إظهار العنوان داخل الأداة على الشاشة	(فارغ) احذف الموجود
	RightToLeft	اتجاه النص لليمين	Yes
TextBox	Name	تحديد اسم الأداة	Textbox2
	Text	إظهار العنوان داخل الأداة على الشاشة	(فارغ) احذف الموجود
	RightToLeft	اتجاه النص لليمين	Yes

القيمة	الغرض من الخاصية	اسم الخاصية	اسم الأداة
GroupBox1	تحديد اسم الأداة	Name	GroupBox
انقر لاختيار عملية التحويل إلى	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	
RadioButton1	تحديد اسم الأداة	Name	RadioButton
درجة الحرارة بالمتوي	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	
RadioButton2	تحديد اسم الأداة	Name	RadioButton
درجة الحرارة بالفهرنهايت	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	
Button1	تحديد اسم الأداة	Name	Button
أدخل درجة الحرارة	إظهار العنوان داخل الأداة على الشاشة	Text	
Button2	تحديد اسم الأداة	Name	Button
التحويل	إظهار العنوان داخل الأداة على الشاشة	Text	

ثالثاً/ كتابة أوامر البرنامج :

اكتب أوامر البرنامج بالنقر المزدوج على أداة زر الأمر (أدخل درجة الحرارة) للدخول على شاشة البرمجة وكتابة الأوامر الخاصة به، ثم انقر المزدوج على أداة زر الأمر (التحويل) للدخول على شاشة البرمجة وكتابة الأوامر الخاصة به، وأوامر هذا البرنامج كما يلي:

```
Public Class Form1
    Dim D As Single
    Private Sub Button1__Click()
        D = InputBox("أدخل درجة الحرارة")
    End Sub
    Private Sub Button2__Click()
        If RadioButton1.Checked = True Then
```




```

TextBox1.Text = D
TextBox2.Text = 5 / 9 * (D - 32)
RadioButton1.Checked = False
ElseIf RadioButton2.Checked = True Then
    TextBox2.Text = D
    TextBox1.Text = (9 / 5 * D) + 32
    RadioButton2.Checked = False
End If
End Sub
End Class

```

شرح البرنامج :

السطر الثاني: تعريف المتغيرات العامة التي تستخدم في أكثر من إجراء، وهو هنا درجة الحرارة (D).

السطر الثالث والرابع: عندما يتم النقر على زر الأمر (أدخل درجة الحرارة) نفذ التالي: تخزين القيمة المدخلة في مربع الإدخال في متغير درجة الحرارة (D).

السطر السادس: عندما يتم النقر على زر الأمر (التحويل) نفذ التالي:

السطر السابع إلى التاسع: عندما يقوم المستخدم بتحديد اختياره الأول (التحويل إلى درجة الحرارة بالمئوي) فإنه يتم طباعة الرقم المدخل من المستخدم (D) في مربع النص (Textbox1) والرقم بعد عملية التحويل في مربع النص (Textbox2).

السطر العاشر: إعادة قيمة زر الاختيار إلى القيمة الافتراضية وهي (صفر).

السطر الحادي عشر إلى الثالث عشر: عندما يقوم المستخدم بتحديد اختياره الثاني (التحويل إلى درجة الحرارة بالفهرنهايت) فإنه يتم طباعة الرقم المدخل من المستخدم (D) في مربع النص (Textbox2) والرقم بعد عملية التحويل في مربع النص (Textbox1).

السطر الرابع عشر: إعادة قيمة زر الاختيار إلى القيمة الافتراضية وهي (صفر).

السطر السادس عشر: نهاية الإجراء

رابعاً/ حفظ النموذج :

يُحفظ النموذج بإتباع الخطوات التالية:

- ١ اختيار الأمر (Save Form1 As...) من قائمة (File).
- ٢ كتابة اسم البرنامج (تحويل درجة الحرارة).

خامساً/ تنفيذ البرنامج :

يُنفذ البرنامج للتأكد من سلامته برمجياً، وذلك بالنقر على زر التنفيذ (▶) من شريط الأدوات القياسي.

تمرينات



س ١: اكتب برنامجاً باستخدام (فيجول بيسك) لإدخال وقت بصيغة (٢٤) ساعة وتحويلها إلى صيغة (١٢)

ساعة وإخراجها بعبارة "صباحاً" أو "مساءً"، وذلك عن طريق:

- ▶ تصميم نموذج واجهة البرنامج.
- ▶ تحديد الخصائص لكل أداة وضعتها على النموذج.
- ▶ كتابة أوامر البرنامج.
- ▶ تنفيذ البرنامج.



تمريعات



اسم الأداة	الخاصية	القيمة
Button1	Text	اضغط لإدخال الأعداد
Label1	Text	"فارغ"

أوامر البرنامج

```

Dim a As Long
a = TextBox1.Text
If a < 12 Then
    Label2.Text = a.ToString + ":00"
    ÖÇİÇ"
Else
    a = a - 12
    Label2.Text = a.ToString + ":00"
    ãÓÇÁ"
End If

```

التنفيذ:



..... التدريب السابع : تطبيق حساب الزكاة

في هذا التدريب ستتعلم :

🔴 تصميم برنامج بلغة (فيجول بيسك ٢٠١٠) لحساب زكاة مبلغ من المال.



متطلبات التدريب

🟡 برنامج (فيجول بيسك ٢٠١٠).

مقدمة التدريب

في هذا التدريب ستقوم بكتابة برنامج باستخدام (فيجول بيسك ٢٠١٠) يطلب من المستخدم ادخال مبلغ معين من المال، ويقوم البرنامج بحساب الزكاة للمبلغ إذا تجاوز (1000 ريال)، أما إذا كان المبلغ أقل من أو يساوي (1000 ريال) فإن البرنامج يطبع الرسالة التالية (البرنامج يقوم بحساب الزكاة لأكثر من 1000 ريال)، علماً أن الزكاة = $0.025 \times \text{المبلغ}$.

خطوات التدريب

أولاً/ التصميم الورقي :

١ صياغة حل المسألة :

حلّ عناصر المسألة، وذلك بتحديد التالي:

• مخرجات البرنامج: مقدار زكاة المال.

• مدخلات البرنامج: مبلغ المال ولنرمز له بـ (M).

• عمليات المعالجة: هذه المسألة تحتوي على عمليتي معالجة:

• الشرط في المسألة، وهو أن تحتسب الزكاة للمال الذي يتجاوز (1000 ريال).

• قانون حساب الزكاة هو: زكاة المال $(Z) = M \times 0.025$.

٢ كتابة الخطوات الخوارزمية للمسألة :

اكتب الخطوات الخوارزمية كما يلي:

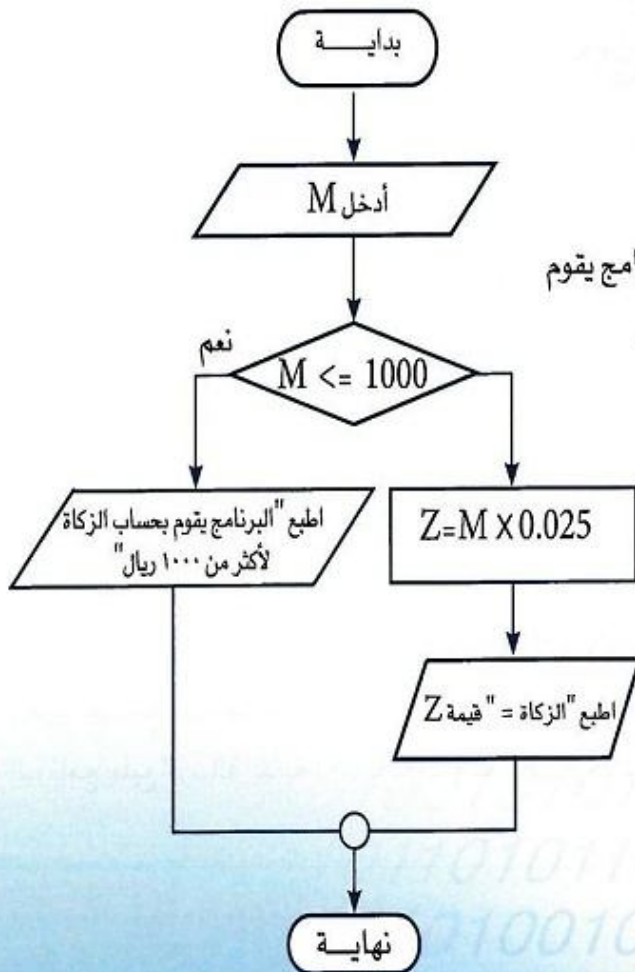
(أ) إدخال المبلغ M.

(ب) هل المبلغ $M \geq 1000$ ، اطبع عبارة "البرنامج يقوم بحساب الزكاة لأكثر من ١٠٠٠ ريال"، وتوقف.

(ج) احسب $Z = M \times 0.025$.

(د) اطبع Z.

(هـ) النهاية.



٣ المخطط الإنسيابي



ثانياً / تصميم نافذة النموذج :

١ افتح برنامج (فيجول بيسك) وأنشئ مشروعاً جديداً.

٢ اضبط خصائص نموذج برنامج حساب الزكاة وفق الجدول التالي:

القيمة	الغرض من الخاصية	اسم الخاصية
Form1	تحديد اسم النموذج	Name
برنامج حساب الزكاة	إظهار عنوان للنموذج	Text
Yes	اتجاه النص لليمين	RightToLeft

٣ أنشئ الأدوات التالية على النموذج، كما في

الشكل (٦-٧-١)، واضبط خصائص كل

أداة كما في الجدول التالي:

شكل ٦-٧-١ : نموذج برنامج حساب الزكاة

القيمة	الغرض من الخاصية	اسم الخاصية	اسم الأداة
Label1	تحديد اسم الأداة	Name	Label
أدخل المبلغ (أكثر من ١٠٠٠ ريال)	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	
Textbox1	تحديد اسم الأداة	Name	Textbox
(فارغ) احذف الموجود	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	
Textbox2	تحديد اسم الأداة	Name	TextBox
(فارغ) احذف الموجود	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	
Button1	تحديد اسم الأداة	Name	Button
الزكاة	إظهار العنوان داخل الأداة على الشاشة	Text	

Public Class Form1

Private Sub Button1_Click()

Dim M As Single, Z As Single

M = Textbox1.Text

If M <= 1000 Then

Textbox2.Text = "البرنامج يقوم بحساب الزكاة لأكثر من ١٠٠٠ ريال"

Else

Z = 0.025 * M

Textbox2.Text = "الزكاة = " & Z

End If

End Sub

End Class

ثالثاً/ كتابة أوامر البرنامج :

اكتب أوامر البرنامج بالنقر
المزدوج على أداة زر الأمر (زكاة)
للدخول على شاشة البرمجة وكتابة
الأوامر الخاصة به، وأوامر هذا
البرنامج كما يلي:



شرح البرنامج :

السطر الثاني: عندما يتم النقر على زر الأمر ضغطة واحدة ينفذ البرنامج ما يلي:
السطر الثالث: تعريف المتغيرات المستخدمة في البرنامج وهي (M)، التي تمثل المبلغ المدخل، والمتغير (Z) الذي يمثل قيمة الزكاة.
السطر الرابع: الحصول على البيانات المدخلة في مربع النص (Textbox1)، وتخزينها في المتغير (M).
السطر الخامس والسادس: اختبار قيمة المبلغ المدخل، فإذا كان المبلغ أقل من أو يساوي (1000) ريال فيتم طبع الرسالة التالية:
(البرنامج يقوم بحساب الزكاة لأكثر من ١٠٠٠ ريال)، ثم يتوقف البرنامج.
السطر السابع إلى التاسع: أما إذا كانت قيمة المبلغ أكبر من (1000) ريال فيتم حساب قيمة الزكاة (Z) ويتم طباعة العبارة (الزكاة =) مع قيمة Z.
السطر الحادي عشر: نهاية الإجراء.

رابعاً/ حفظ النموذج :

يُحفظ النموذج بإتباع الخطوات التالية:

- ١ اختيار الأمر (Save Form1 As...) من قائمة (File).
- ٢ كتابة اسم النموذج (زكاة).

خامساً/ تنفيذ البرنامج :

ينفذ البرنامج للتأكد من سلامته برمجياً، وذلك بالنقر على زر التنفيذ (▶) من شريط الأدوات القياسي.

تمرين التدريب

س ١: اكتب برنامجاً باستخدام (فيجول بيسك) يقوم لإدخال (١٠) أعداد ثم

طباعة العدد الأكبر منها، وذلك عن طريق:

- ▶ تصميم نموذج واجهة البرنامج.
- ▶ تحديد الخصائص لكل أداة وضعتها على النموذج.
- ▶ كتابة أوامر البرنامج.
- ▶ تنفيذ البرنامج.



تمرينات



القيمة	الخاصية	اسم الأداة
اضغط لإدخال الأعداد	Text	Button1
"فارغ"	Text	Label1

تمرينات

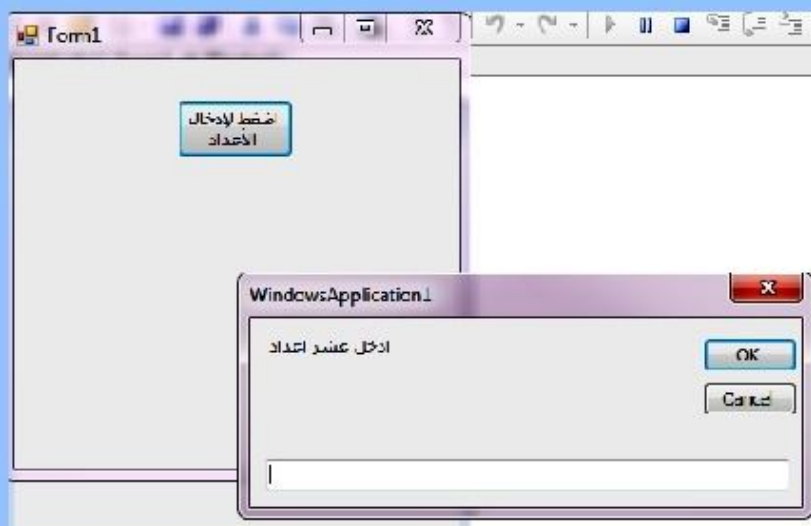


الكود

```
Dim g(10), count, max As Integer
For count = 0 To 9
    أدخل عشر أعداد") g(count) = InputBox("
Next
max = 0
For count = 0 To 9
    If g(count) > max Then
        max = g(count)
    End If
Next
Label1.Text = max
```

التنفيذ:

تمرينات



.....التدريب الثامن : تطبيق حساب درجات وتقدير طالب

في هذا التدريب ستتعلم :

🔴 تصميم برنامج بلغة (فيجول بيسك ٢٠١٠) لحساب درجات وتقدير طالب في مادة.



متطلبات التدريب

🟡 برنامج (فيجول بيسك ٢٠١٠).

مقدمة التدريب

في هذا التدريب ستقوم بكتابة برنامج باستخدام (فيجول بيسك) لحساب مجموع الدرجات التي يدخلها المستخدم لطالب في مادة الحاسب، والدرجات هي: (أعمال السنة، الحضور والغياب، الاختبار الفصلي، الاختبار النهائي)، ثم يطبع البرنامج التقدير، علماً أن التقدير يحسب كالتالي:

الدرجة	90 فأكثر	89-80	79-70	69-50	أقل من 50
التقدير	ممتاز	جيد جداً	جيد	مقبول	غير مجتاز



خطوات التدريب

أولاً/ التصميم الورقي :

صياغة حل المسألة :

١

حلّ عناصر المسألة، وذلك بتحديد التالي:

- مخرجات البرنامج: تقدير الطالب في مادة الحاسب.
- مدخلات البرنامج: اسم الطالب و(٤) مدخلات للدرجات هي:
 - (A0) أعمال السنة.
 - (A1) الحضور والغياب.
 - (A2) الاختبار الفصلي.
 - (A3) الاختبار النهائي.

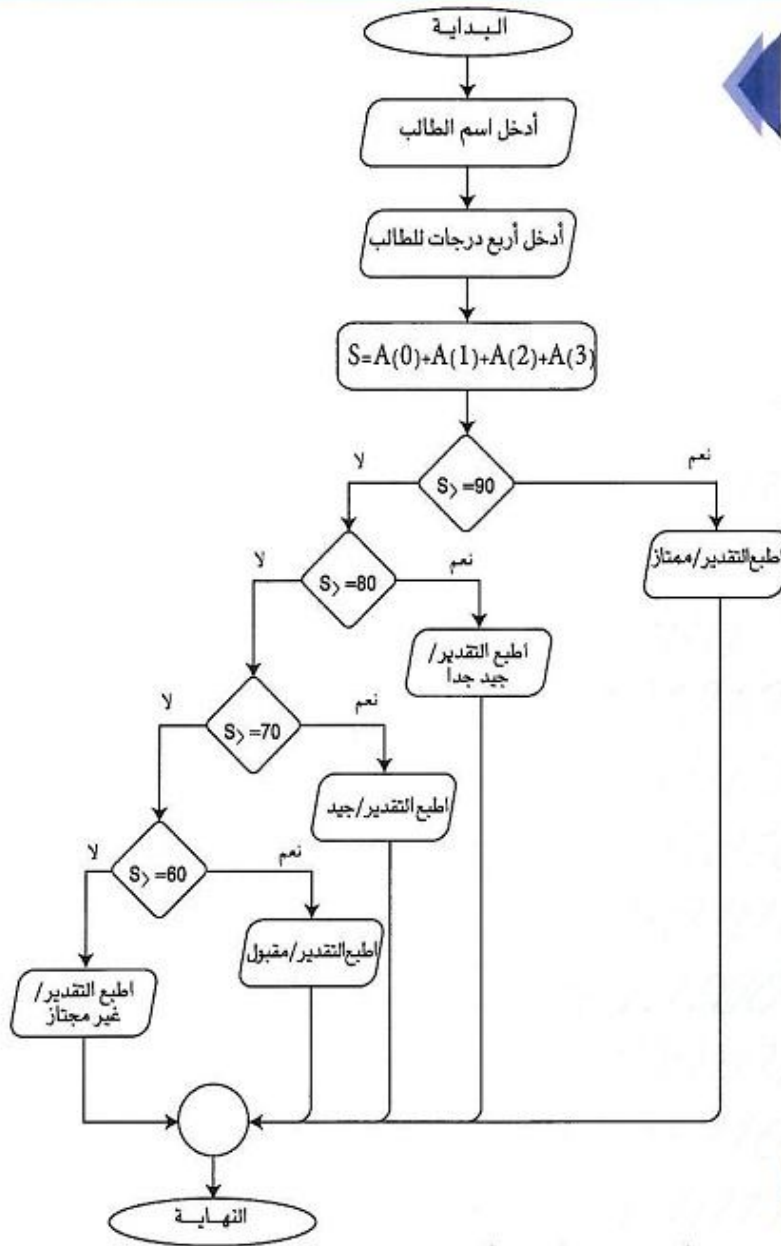
● عمليات المعالجة: جمع الدرجات الأربع، ويرمز له بالرمز (S) ومقارنته بالتقدير المناسب.

كتابة الخطوات الخوارزمية للمسألة :

٢

اكتب الخطوات الخوارزمية كما يلي:

- أ) أدخل اسم الطالب.
- ب) أدخل درجات الطالب الأربع على التوالي.
- ج) خزّن الدرجات في المصفوفة.
- د) جمع الدرجات الأربع: $S = A(0) + A(1) + A(2) + A(3)$
- هـ) قارن بين مجموع الدرجات (S) والتقدير المناسب.
- و) اطبع النتيجة مع بيانات الطالب.
- ز) النهاية.

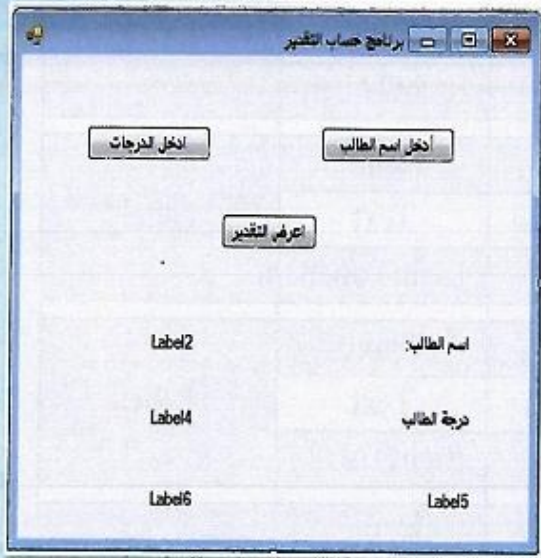


ثانياً/ تصميم نافذة النموذج :

١ افتح برنامج (فيجول بيسك) وأنشئ مشروعاً جديداً.

٢ اضبط خصائص نموذج برنامج حساب التقدير وفق الجدول التالي:

القيمة	الفرض من الخاصية	اسم الخاصية
Form1	تحديد اسم النموذج	Name
برنامج حساب التقدير	إظهار عنوان للنموذج	Text
Yes	اتجاه النص لليمين	RightToLeft



٣ أنشئ الأدوات التالية على النموذج، كما في الشكل (١-٨-٦)، واضبط خصائص كل أداة كما في الجدول التالي:

شكل ١-٨-٦: نموذج برنامج حساب التقدير

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
Button	Name	تحديد اسم الأداة	Button1
	Text	إظهار العنوان داخل الأداة على الشاشة	أدخل اسم الطالب
Button	Name	تحديد اسم الأداة	Button2
	Text	إظهار العنوان داخل الأداة على الشاشة	أدخل الدرجات
Button	Name	تحديد اسم الأداة	Button3
	Text	إظهار العنوان داخل الأداة على الشاشة	اعرض التقدير
Label	Name	تحديد اسم الأداة	Label1
	Text	إظهار العنوان داخل الأداة على الشاشة	اسم الطالب
	RightToLeft	اتجاه النص لليمين	Yes
Label	Name	تحديد اسم الأداة	Label2
	Text	إظهار العنوان داخل الأداة على الشاشة	احذف الموجود (فراغ)
	RightToLeft	اتجاه النص لليمين	Yes

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
Label	Name	تحديد اسم الأداة	Label3
	Text	إظهار العنوان داخل الأداة على الشاشة	درجات الطالب
	RightToLeft	اتجاه النص لليمين	Yes
Label	Name	تحديد اسم الأداة	Label4
	Text	إظهار العنوان داخل الأداة على الشاشة	احذف الموجود (فارغ)
	RightToLeft	اتجاه النص لليمين	Yes
Label	Name	تحديد اسم الأداة	Label5
	Text	إظهار العنوان داخل الأداة على الشاشة	احذف الموجود (فارغ)
	RightToLeft	اتجاه النص لليمين	Yes
Label	Name	تحديد اسم الأداة	Label6
	Text	إظهار العنوان داخل الأداة على الشاشة	احذف الموجود (فارغ)
	RightToLeft	اتجاه النص لليمين	Yes

ثالثاً/ كتابة أوامر البرنامج :

اكتب أوامر البرنامج بالنقر على أداة زر الأمر (أدخل اسم الطالب) للدخول على شاشة البرمجة، وكتابة الأوامر الخاصة بها، ثم انقر على أداة زر الأمر (أدخل الدرجات) للدخول على شاشة البرمجة، وكتابة الأوامر الخاصة بها، وأخيراً انقر على أداة زر الأمر (اعرض التقدير) للدخول على شاشة البرمجة، وكتابة الأوامر الخاصة بها، فيكون البرنامج كما يلي:

```
Public Class Form1
Dim I As Integer
Dim A (3) As Integer
Dim N As String
Private Sub Button1__Click ()
N = InputBox("فضلاً أدخل اسم الطالب")
EndSub
Private Sub Button2__Click()
For I = 0 To 3
```




```

A (I) = InputBox("أدخل درجة الطالب في مادة الحاسب الآلي (أعمال السنة ، الحضور والغياب، الاختبار)
                    ("الفصلي، الاختبار النهائي) على التوالي")
Next
EndSub
Private Sub Button3__Click()
    Dim S As Single
    Label2.Text = N
    Label4.Text = A(0) & "--" & A(1) & "--" & A(2) & "--" & A(3)
    S = A (0) + A (1) + A (2) + A (3)
    Label5.Text = "مجموع الدرجات"& S
    If S >= 90 Then
        Label6.Text = "التقدير / ممتاز"
    ElseIf S >= 80 Then
        Label6.Text = "التقدير / جيد جداً"
    ElseIf S >= 70 Then
        Label6.Text = "التقدير / جيد"
    ElseIf S >= 50 Then
        Label6.Text = "التقدير / مقبول"
    Else
        Label6.Text = "التقدير / غير مجتاز"
    EndIf
EndSub
EndClass

```

شرح البرنامج :

السطر الثاني إلى الثالث: تعريف المتغيرات العامة التي تستخدم في أكثر من إجراء. وهي:
 العدد (I) المستخدم مع المصفوفة، المصفوفة (A) وعدد عناصرها (4-1)، اسم الطالب (N).
 السطر الخامس والسادس: عندما يتم النقر على زر الأمر (أدخل اسم الطالب) نفذ التالي: أظهر مربع الإدخال، واحفظ الاسم في المتغير (N).
 السطر الثامن إلى الحادي عشر: عندما يتم النقر على زر الأمر (ادخل الدرجات) نفذ التالي: ابدأ العدد (تكرار) FOR بعدد مساو لعدد المصفوفة، وخلال ذلك أظهر مربع الإدخال (أدخل درجة الطالب في مادة الحاسب الآلي (أعمال السنة، الحضور والغياب، الاختبار الفصلي، الاختبار النهائي) لإدخال درجات الطالب، واحفظ الرقم المدخل في المصفوفة.

السطر الثالث عشر والرابع عشر: عندما يتم النقر على زر الأمر (اعرض التقدير) نفذ التالي: تعريف المتغيرات: تمثل (S) مجموع الدرجات المدخلة.
السطر الخامس عشر إلى الثامن عشر: أظهر المحتويات التالية في مربع العنوان: اسم الطالب في (Label2)، ودرجات الطالب في مادة الحاسب الآلي في (Label4)، ومجموع الدرجات بعد جمعها وتخزينها في (S) في (Label5)
السطر التاسع عشر إلى الثامن والعشرون: تبدأ هنا عملية المقارنة لمجموع الدرجات بالتقدير وإظهار النتيجة بعد عملية المقارنة في (Label6).
السطر الثلاثون: نهاية الإجراء.

رابعاً/ حفظ النموذج :

- يُحفظ النموذج بإتباع الخطوات التالية:
- ١ اختيار الأمر (Save Form1 As...) من قائمة (File).
 - ٢ كتابة اسم النموذج (درجات وتقدير الطالب).

خامساً/ تنفيذ البرنامج :

يُنفذ البرنامج للتأكد من سلامته برمجياً، وذلك بالنقر على زر التنفيذ (▶) من شريط الأدوات القياسي.

تمارين

س١ : اكتب برنامجاً باستخدام (فيجول بيسك) لإدخال أسماء (10) طلاب ودرجاتهم في مادة الحاسب الآلي ، ثم طباعة اسم الطالب الحاصل على أعلى درجة ، وطباعة درجته وتقديره، علماً أن التقدير يحسب كالتالي :

الدرجة	90 فأكثر	89-80	79-70	69-50	أقل من 50
التقدير	ممتاز	جيد جداً	جيد	مقبول	غير مجتاز





..... التدريب التاسع : تطبيق أجهزة الحاسب

في هذا التدريب ستتعلم :

● تصميم برنامج بلغة فيجول (بيسك ٢٠١٠) لعرض أسعار أجهزة الحاسب .



متطلبات التدريب

● برنامج (فيجول بيسك ٢٠١٠).

مقدمة التدريب

في هذا التدريب ستقوم بكتابة برنامج باستخدام (فيجول بيسك) لعرض عدد من أجهزة الحاسب والشركات المصنعة لها، وعندما يحدد المستخدم اسم الجهاز والشركة المصنعة، يعرض البرنامج سعره، حسب الجدول التالي:

اسم الجهاز	الشركة المصنعة	السعر
حاسب محمول	DELL	6127
	LG	6131
	HP	5175
حاسب لوحي	SAMSUNG	2737
	APPLE	2906
طابعة متعددة المهام	EPSON	1631
	HP	1871
حاسب مكتبي	LG	2850
	DELL	4050
جهاز عرض	LG	7687
	HP	8625

خطوات التدريب

أولاً/ التصميم الورقي :

١ صياغة حل المسألة :

حلّ عناصر المسألة، وذلك بتحديد التالي:

- مخرجات البرنامج: سعر جهاز الحاسب.
- مدخلات البرنامج: اسم الجهاز (D)، واسم الشركة المصنعة للجهاز (K).
- عمليات المعالجة: تحديد اسم جهاز الحاسب والشركة المصنعة ثم عرض السعر.

٢ كتابة الخطوات الخوارزمية للمسألة :

اكتب الخطوات الخوارزمية كما يلي:

(أ) اعرض أسماء أجهزة الحاسب وأنواعها.

(ب) حدد اسم الجهاز (D)، واسم الشركة المصنعة للجهاز (K).

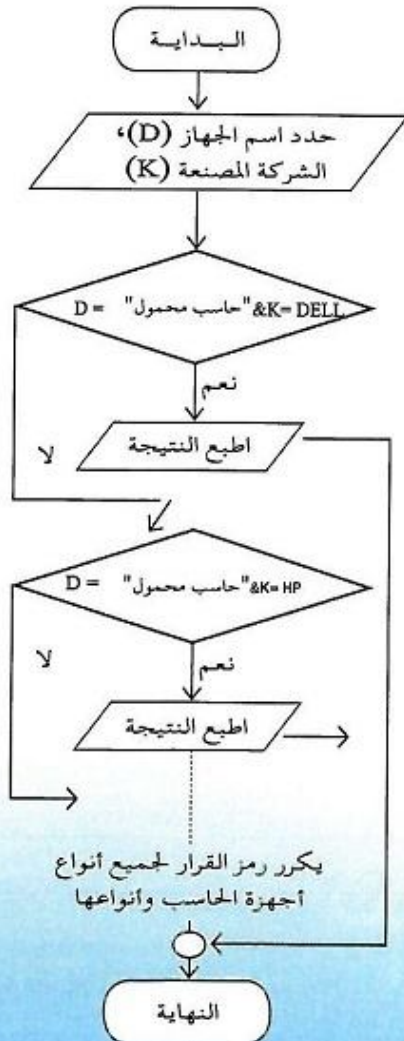
(ت) قارن بين اسم الجهاز (D)، واسم الشركة المصنعة للجهاز (K)

الذين تم اختيارهما لتحديد السعر.

(ث) اطبع النتيجة.

(ج) النهاية.

٣ المخطط الإنسيابي





ثانياً/ تصميم نافذة النموذج :

١ افتح برنامج (فيجول بيسك) وأنشئ مشروعاً جديداً.

٢ اضبط خصائص نموذج برنامج أجهزة الحاسب وفق الجدول التالي:

القيمة	الفرض من الخاصية	اسم الخاصية
Form1	تحديد اسم النموذج	Name
برنامج أجهزة الحاسب	إظهار عنوان للنموذج	Text
Yes	اتجاه النص لليمين	RightToLeft



٣ أنشئ الأدوات التالية على النموذج، كما في الشكل (٦-٩-١)، واضبط خصائص كل أداة كما في الجدول التالي:

شكل ٦-٩-١: نموذج برنامج معرض أجهزة الحاسب

القيمة	الفرض من الخاصية	اسم الخاصية	اسم الأداة
Label1	تحديد اسم الأداة	Name	Label
اسم الجهاز	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	
Label2	تحديد اسم الأداة	Name	Label
الشركة المصنعة	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	

القيمة	الغرض من الخاصية	اسم الخاصية	اسم الأداة
TextBox1	تحديد اسم الأداة	Name	TextBox
احذف الموجود (فارغ)	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	
ListBox1	تحديد اسم الأداة	Name	ListBox
اكتب أسماء الأجهزة، وكل اسم في سطر	إظهار العنوان داخل الأداة على الشاشة	Items	
Yes	اتجاه النص لليمين	RightToLeft	
ComboBox1	تحديد اسم الأداة	Name	ComboBox
اكتب اسم الشركة المصنعة للأجهزة، وكل شركة في سطر	إظهار العنوان داخل الأداة على الشاشة	Items	
Yes	اتجاه النص لليمين	RightToLeft	
Button1	تحديد اسم الأداة	Name	Button
عرض السعر	إظهار العنوان داخل الأداة على الشاشة	Text	

ثالثاً/ كتابة أوامر البرنامج :

اكتب أوامر البرنامج بالنقر على أداة زر الأمر (عرض السعر) للدخول على شاشة البرمجة، وكتابة الأوامر الخاصة بها، فيكون البرنامج كما يلي:

```
Public Class Form1
Private Sub Button1__Click()
Dim D As String, K As String
D = ListBox1.Text
K = ComboBox1.Text
If D = "حاسب محمول" And K = "DELL" Then
TextBox1.Text = "6127"
ElseIf D = "حاسب محمول" And K = "LG" Then
TextBox1.Text = "6131"
ElseIf D = "حاسب محمول" And K = "HP" Then
TextBox1.Text = "5175"
ElseIf D = "حاسب لوحي" And K = "SAMSUNG" Then
TextBox1.Text = "2737"
```




```

ElseIf D = "حاسب لوحي" And K = "APPLE" Then
    TextBox1.Text = "2906"
ElseIf D = "طابعة متعددة المهام" And K = "EPSON" Then
    TextBox1.Text = "1631"
ElseIf D = "طابعة متعددة المهام" And K = "HP" Then
    TextBox1.Text = "1871"
ElseIf D = "حاسب مكتبي" And K = "LG" Then
    TextBox1.Text = "2850"
ElseIf D = "حاسب مكتبي" And K = "DELL" Then
    TextBox1.Text = "4050"
ElseIf D = "جهاز عرض" And K = "LG" Then
    TextBox1.Text = "7687"
ElseIf D = "جهاز عرض" And K = "HP" Then
    TextBox1.Text = "8625"
Else
    TextBox1.Text = "لا يوجد جهاز بهذه المواصفات"
EndIf
EndSub
EndClass

```

شرح البرنامج :

السطر الثاني والثالث: عندما يتم النقر على زر الأمر (عرض السعر) نفذ التالي: تعريف المتغيرات المستخدمة في البرنامج وهي: (D) وتمثل اسم الجهاز، و(K) تمثل اسم الشركة المصنعة للجهاز .

السطر الرابع: يخزن قيمة العنصر الذي اختاره المستخدم في مربع القائمة في (D).

السطر الخامس: يخزن قيمة العنصر الذي اختاره المستخدم في مربع الخانة المركبة في (K).

السطر السادس إلى السابع والعشرين: يقوم باختبار قيمة (D) و(K) التي اختارها المستخدم وهما اسم الجهاز ونوعه ثم يطبع السعر في مربع النص، ويطبق ذلك على جميع الأسماء والأنواع.

السطر التاسع والعشرون: إذا لم يكن هناك تطابق يتم طباعة رسالة بعدم وجود ذلك الجهاز في مربع النص (Textbox1).

السطر الواحد والثلاثون: نهاية الإجراء.

رابعاً/ حفظ النموذج :

يُحفظ النموذج باتباع الخطوات التالية:

١ اختيار الأمر (Save Form1 As...) من قائمة (File).

٢ كتابة اسم النموذج (درجات وتقدير الطالب).

خامساً/ تنفيذ البرنامج :

يتم تنفيذ البرنامج للتأكد من سلامته برمجياً، وذلك بالنقر على زر التنفيذ (▶) من شريط الأدوات القياسي.

تمارين

لديك معرض سيارات يحتوي عدة أنواع وعدة موديلات من السيارات، اكتب برنامجاً باستخدام (فيجول بيسك) لعرض سعر السيارة عندما يحدد المستخدم النوع والموديل، وذلك حسب الجدول التالي:



النوع	الموديل	السعر
كامري	2012	67000
كامري	2013	85000
ماكسيما	2012	60000
ماكسيما	2013	70000
أكورد	2012	65000
أكورد	2013	83000
سوناتا	2012	59000
سوناتا	2013	71000