

قررت وزارة التعليم تدريس  
هذا الكتاب وطبعه على نفقتها

# الحاسب



## وتقنية المعلومات

المستوى الثاني  
النظام الفصلي للتعليم الثانوي  
الإعداد العام

قام بالتأليف والمراجعة  
فريق من المتخصصين

يُوزع مجاناً للإتباع

طبعة تجريبية  
١٤٣٧ - ١٤٣٨ هـ  
٢٠١٦ - ٢٠١٧ م

## ح) وزارة التعليم ١٤٣٥ هـ

فهرسة مكتبة الملك فهد الوطنية أثناء النشر

وزارة التعليم

الحاسب وتقنية المعلومات للمستوى الثاني: النظام الفصلي للتعليم الثانوي -

كتاب الطالب والتدريبات العملية - /وزارة التعليم - الرياض. ١٤٣٥ هـ

١٥٢ ص؛ ٢١ × ٢٥,٥ سم

ردمك : ٩ - ٤٩٦ - ٥٠٢ - ٦٠٣ - ٩٧٨

١- الحواسيب - كتب دراسية ٢- التعليم الثانوي - السعودية - كتب دراسية

أ.العنوان

١٤٣٥ / ٣٧٣٨

ديوي ٠٠٤,٧١٢

رقم الإيداع : ١٤٣٥ / ٣٧٣٨

ردمك : ٩ - ٤٩٦ - ٥٠٢ - ٦٠٣ - ٩٧٨

لهذا المقرر قيمة مهمة وفائدة كبيرة فلنحافظ عليه، ولنجعل نظافته تشهد على حسن سلوكنا معه.

إذا لم نحفظ بهذا المقرر في مكتبتنا الخاصة في آخر العام للاستفادة، فلنجعل مكتبة مدرستنا تحتفظ به.

حقوق الطبع والنشر محفوظة لوزارة التعليم - المملكة العربية السعودية

وزارة التعليم

موقع

[www.moe.gov.sa](http://www.moe.gov.sa)

البريد الإلكتروني :

لقسم الحاسب - الإدارة العامة للمناهج

[computer.cur@moe.gov.sa](mailto:computer.cur@moe.gov.sa)





بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



# الفهرس

## الوحدة الرابعة : مقدمة في البرمجة

- (١-٤) مقدمة .. ص ٩
- (٢-٤) مفهوم البرمجة .. ص ٩
- (٣-٤) أهمية البرمجة .. ص ١١
- مشروع الوحدة .. ص ١٦
- خارطة المفاهيم .. ص ١٧
- دليل الدراسة .. ص ١٨
- تمارينات .. ص ١٩
- اختبار .. ص ٢٠

## الوحدة الخامسة : صياغة حل المسائل

- (١-٥) مقدمة .. ص ٢٣
- (٢-٥) الهدف من تعلم صياغة حل المسائل .. ص ٢٣
- (٣-٥) خطوات حل المسائل .. ص ٢٣
- (٤-٥) فهم المسألة وتحليل عناصرها .. ص ٢٤
- (٥-٥) كتابة الخطوات الخوارزمية .. ص ٢٥
- (٦-٥) مخططات الانسياب .. ص ٢٧
- (٧-٥) أمثلة على صياغة حل المسائل .. ص ٣١
- مشروع الوحدة .. ص ٣٥
- خارطة المفاهيم .. ص ٣٦
- دليل الدراسة .. ص ٣٧
- تمارينات .. ص ٣٨
- اختبار .. ص ٤٠

## الوحدة السادسة : البرمجة بلغة (فيجول بيسك ستوديو)

- (١-٦) مقدمة .. ص ٤٥
- (٢-٦) مراحل كتابة البرنامج بلغة (فيجول بيسك ستوديو) .. ص ٤٥
- (٣-٦) طريقة تعامل البرنامج مع البيانات .. ص ٤٧
- (٤-٦) العمليات الحسابية والمنطقية .. ص ٥٠
- (٥-٦) أدوات البرمجة بلغة (فيجول بيسك ستوديو) .. ص ٥٤
- (٦-٦) بعض الأوامر الأساسية للغة (فيجول بيسك ستوديو) .. ص ٦٤
- مشروع الوحدة .. ص ٧٧
- خارطة المفاهيم .. ص ٧٨
- دليل الدراسة .. ص ٧٩
- تمارين .. ص ٨٠
- اختبار .. ص ٨٢

### تدريبات الوحدة السادسة:

- التدريب الأول : التعرف على بيئة العمل .. ص ٨٦
- التدريب الثاني : مشروع برنامج (حسابات ومساحات) :
- تصميم نموذج المقدمة والنموذج الثاني .. ص ٩٤
- التدريب الثالث : مشروع برنامج (حسابات ومساحات) :
- تصميم النموذج الثالث .. ص ١٠٣
- التدريب الرابع : مشروع برنامج (حسابات ومساحات) :
- كتابة الأوامر البرمجية لنموذج (حسابات) .. ص ١٠٩
- التدريب الخامس : مشروع برنامج (حسابات ومساحات) : كتابة الأوامر البرمجية لنموذج (مساحات واستكمال المشروع .. ص ١١٦
- التدريب السادس : تطبيق تحويل درجة الحرارة .. ص ١٢٣
- التدريب السابع : تطبيق حساب الزكاة .. ص ١٣٠
- التدريب الثامن : تطبيق حساب درجات وتقدير طالب .. ص ١٣٦
- التدريب التاسع : تطبيق أجهزة الحاسب .. ص ١٤٢
- المصطلحات .. ص ١٥٠





## الوحدة الرابعة

# مقدمة في البرمجة

### موضوعات الوحدة :

- مفهوم البرمجة.
- أهمية البرمجة.
- أقسام لغات البرمجة.

## بعد دراستك لهذه الوحدة سوف تحقق الأهداف التالية :

- ◀ توضّح أهمية البرمجة.
- ◀ تشرح الفرق بين المبرمج ومستخدم البرنامج.
- ◀ تعدّد طرق الحصول على البرامج.
- ◀ تعرّف البرنامج.
- ◀ تعدّد بعض لغات الحاسب المستخدمة.
- ◀ تشرح مفهوم لغات البرمجة بالكائنات.
- ◀ تفرّق بين البرمجة بالكائنات ولغات البرمجة الإجرائية.

## الأهمية :

تعتبر هذه الوحدة مدخلاً أساسياً للتعرف على لغات البرمجة التي يتعامل معها الحاسب الآلي، حيث يتم التعرف على مفهوم البرمجة وأهميتها والتفريق بين عمل المبرمج ومستخدم البرنامج مع دراسة أنواع لغات البرمجة وخاصة لغات البرمجة بالكائنات.

والغرض الأساسي من تدريس هذه الوحدة معرفة الطالب للحقائق العلمية والمعرفية في مجال البرمجة ولغات الحاسب المختلفة.



## مقدمة

١-٤

جهاز الحاسب الآلي آلة تنفذ ما يأتيها من أوامر بدقة؛ حيث تكون هذه الأوامر مكتوبة فيما يسمى بـ (برنامج)، فـجهاز الحاسب الآلي دون برامج لا فائدة منه. وجميع البرامج تكون مكتوبة على هيئة سلسلة من الأوامر اليسيرة التي ينفذها الحاسب الآلي لتخرج لنا بالشكل الذي نراه.

## مفهوم البرمجة

٢-٤

إثراء علمي

### أنواع البرمجيات

- أنظمة التشغيل.
- برامج التشغيل المساعدة.
- البرامج التطبيقية.

إن هذه الأوامر تكتب بلغة معينة يفهمها جهاز الحاسب الآلي؛ حيث يوجد لدينا عديد من لغات البرمجة التي سوف نتطرق لبعض منها في هذه الوحدة، كذلك سوف نستعرض الخطوات اللازم معرفتها لحل المسائل عند كتابة أي برنامج. وكما قلنا سابقاً، إذا أردنا من جهاز الحاسب الآلي أن ينفذ عملاً معيناً فعلياً أن نعطيه الأوامر اللازمة لتنفيذ هذا العمل، وهذه الأوامر مكتوبة على شكل برنامج.

### ولكن كيف نكتب الأوامر؟

لو أردت أن تطلب من شخص ما أن يجمع رقمين لقلت له : ما هو حاصل جمع ٢+٣؟ ولكن لنفترض أنك ستطلب ذلك من شخص لا يعرف العربية فكيف ستطلب منه أن يجمع الرقمين؟ يجب أن تعرف اللغة التي يفهمها وتحديثه بها. أليس كذلك؟ كذلك جهاز الحاسب فإنه لا يفهم عبارة «اجمع رقمين»، ولذا فعليك أن تتعلم لغة البرمجة حتى تستطيع أن توجه له الأوامر. وهناك كثير من اللغات التي يفهمها جهاز الحاسب الآلي وتسمى لغات البرمجة.

## الفرق بين المبرمج ومستخدم البرنامج :

١-٢-٤

المبرمج هو من يكتب البرامج، حيث يقوم بالبداية بفهم وتحليل المشكلة التي قدمها المستخدم، ثم يشرع في كتابة سلسلة من الأوامر لحل المشكلة، يلي ذلك اختبار البرنامج والتأكد من صحة عمله. وعند التأكد من صحة عمل البرنامج يحوّل البرنامج إلى صيغة تنفيذية تمثل الشكل النهائي الذي يحتوي على الواجهة التي يراها المستخدم. وفي النهاية فإن مستخدم البرنامج سوف تظهر له واجهة البرنامج ولن تظهر له الأوامر التي كتبها المبرمج.



## كيفية الحصول على البرامج :

٢-٢-٤

هنالك طريقتان لحصول الشركات أو الأفراد على البرامج:

- شراء برنامج جاهز (مكتوب سابقاً).
- تطوير برنامج جديد خاص بالشركة أو الفرد.

### نشاط



لو طُلب منك جمع رقمين، فما الخطوات التي تقوم بها عقلك للحصول على الناتج؟

- ١- .....
- ٢- .....
- ٣- .....
- ٤- .....

والبرامج الجاهزة هي برامج عامة مكتوبة سابقاً يمكن شراؤها مباشرة من السوق، وفي كثير من الأحيان تقي هذه البرامج باحتياجاتك الخاصة، ولكن أحياناً قد لا تكون هذه البرامج مطابقة لما تحتاجه، وهنا سوف تضطر إلى تطوير البرامج بنفسك، ومع أن هذا يحتاج إلى وقت وجهد، إلا أن البرنامج الناتج سوف يلبي احتياجاتك تماماً.

## أهمية الأوامر في البرمجة :

٣-٢-٤

لا يستطيع الحاسب فهم ما يريد المبرمج القيام به؛ حيث يجب أن يكتب المبرمج سلسلة دقيقة من الأوامر تحدد بالتفصيل جميع الخطوات اللازمة لحل المسألة، ففي النشاط السابق يجب القيام بالخطوات التالية لحل المسألة:

- ١- استقبال الرقم الأول واحفظه.
- ٢- استقبال الرقم الثاني واحفظه.
- ٣- اجمع الرقم الأول مع الثاني واحفظه.
- ٤- أعط الناتج.

وعلياً أن نعطي الحاسب هذه الأوامر على هيئة برنامج حتى يتم تنفيذه. ولو أردنا كتابة برنامج يجمع رقمين فإنه سوف يحتوي على هذه الأوامر، ولكن تختلف التسميات قليلاً، فعمليات استقبال الرقمين تسمى إدخال (input) و عملية إعطاء الناتج تسمى إخراج (output).

### نشاط



لو طُلب منك جمع رقمين زوجيين فقط، فما الخطوات التي تقوم بها عقلك للحصول على الناتج؟

- ١- .....
- ٢- .....
- ٣- .....
- ٤- .....



### نشاط

لو طُلب منك جمع خمسة أزواج من الأرقام، فما الخطوات التي يقوم بها عقلك للحصول على النواتج؟

- ١ - .....
- ٢ - .....
- ٣ - .....
- ٤ - .....

هنا سوف تستقبل الرقم الأول وتحفظه، ثم تستقبل الرقم الثاني وتحفظه. يلي ذلك التأكد من تحقق الشرط، فإن تحقق الشرط فسوف تجمع الرقمين وتعطي النتيجة، وإن لم يتحقق الشرط فسوف تطلب استقبال رقمين آخرين يحققان الشرط.

هنا سنقوم بتكرار الأوامر الأربعة في كل مرة حتى يحسب مجموع الخمسة أزواج، وفي كل مرة يقوم الحاسب بتكرار عمليات ( الإدخال - الجمع - الإخراج )، ولكن بدلاً من كتابة هذه الأوامر خمس مرات لحساب مجموع الخمسة أزواج فإنه سوف يحوي على أمر رابع يوضح لجهاز الحاسب الآلي أن عليه أن يقوم بهذه الأوامر الأربعة خمس مرات.

من النشاط السابق نستطيع أن نستنتج أهم الأوامر في أي برنامج:

**الإدخال:** وهي عملية استقبال البيانات التي سوف يعالجها البرنامج.

**الإخراج:** وهي عملية عرض المعلومات التي نتجت عن تنفيذ البرنامج.

**الحساب:** وهي العمليات الحسابية التي ينفذها البرنامج.

**التحقق من الشرط:** وهي عملية التحقق من صحة شرط ما وتنفيذ بعض الأوامر بناء على مصداقية الشرط.

**التكرار:** وهي عملية تكرار مجموعة من الأوامر.

وتسمى أوامر الحساب والتحقق من الشرط والتكرار بعمليات المعالجة.

وتوجد أوامر أكثر تعقيداً من هذه، ولكن هذه أيسر الأوامر التي يجب

أن يحتوي أي برنامج على واحد منها في أقل تقدير.

### إثارة التفكير

ما أهم البرامج المستخدمة في مدرستك ؟

## ٣-٤ أهمية البرمجة

نستطيع القول إن جهاز الحاسب الآلي دون برمجة لا فائدة منه إطلاقاً؛ لأن البرامج هي التي تخبر الجهاز بما عليه القيام به وكيفية القيام بالعمل. ولو نظرت إلى أغلب الأجهزة الإلكترونية التي تستخدمها لوجدت أن كثيراً منها تعمل أيضاً بواسطة برامج مخزنة داخلها.

**أمثلة على أجهزة ونظم تعمل باستخدام برامج خاصة بها :**

**في المنزل:** جهاز الميكرويف، والهاتف المحمول، وبعض الساعات الإلكترونية، وأجهزة الاستقبال الفضائية الرقمية، وجهاز التلفزيون والفيديو.



في الشارع: إشارات المرور، والسيارات الحديثة، وأجهزة الصراف الآلي.  
في المستشفيات: الأجهزة الطبية المختلفة، ونظام المواعيد وملفات المرضى.  
في المحلات التجارية: نظام البيع، وقارئ الأكواد.

### تعريف البرنامج :

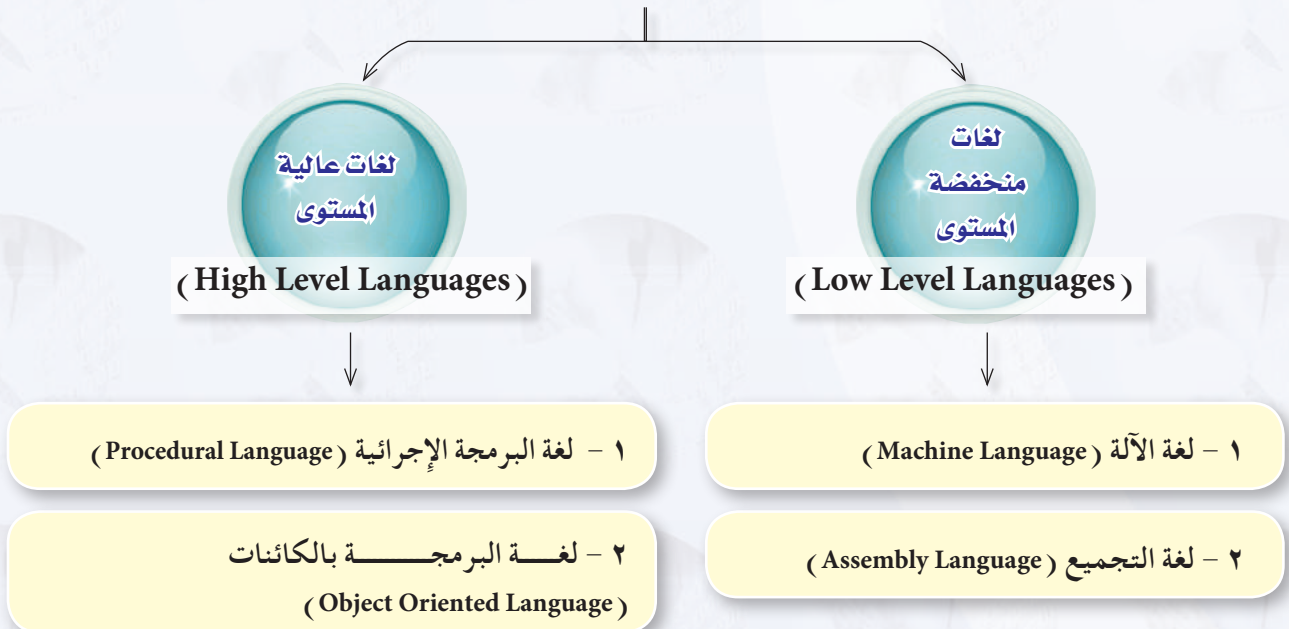
١-٣-٤

**البرنامج** هو سلسلة من الأوامر مكتوبة بلغة برمجة معينة تكتب لتنفيذ مهمة محددة في جهاز الحاسب، ويوجد كثير من لغات البرمجة المستخدمة في كتابة البرامج، وكما نعلم فلغة الآلة هي اللغة التي ينفذها الحاسب مباشرة وهي مجموعة من التعليمات تنفذ مباشرة من قبل وحدة المعالجة المركزية للحاسب (CPU)، حيث تُكتب مجموعة من التعليمات لتنفيذ مهمة محددة للغاية. وتُكتب كافة برامج الحاسب اليوم بلغات متعددة ومختلفة لها قواعدها ومفرداتها تُترجم إلى لغة الآلة بواسطة برنامج المترجم.

### أقسام لغات البرمجة :

٢-٣-٤

تنقسم لغات البرمجة إلى



اللغات المنخفضة المستوى :

أولاً

اللغات منخفضة المستوى هي اللغات القريبة جداً من جهاز الحاسب تظهر التركيب الداخلي للحاسب في مجموعة أوامر وتعليمات هذه اللغات، أي أنها لا توفر تجريداً أو فصلاً بين مجموعة أوامر أو تعليمات هذه اللغات والتركيب الداخلي للحاسب.

وتنقسم لغات البرمجة منخفضة المستوى إلى:

١ - لغة الآلة :

تكتب الأوامر في لغة الآلة على شكل سلسلة من الأرقام الثنائية (الصففر والواحد) حتى يفهمها جهاز الحاسب الآلي مباشرة ، وترمز كل مجموعة من الأرقام إلى عملية محددة يسيرة يمكن للحاسب تنفيذها، وتعامل مباشرة مع ما هو مخزن في الذاكرة الرئيسة للجهاز.

٢ - لغة التجميع :

هي اختصارات للغة الآلة؛ حيث تُكتب الأوامر على هيئة اختصارات مفهومة باللغة الانجليزية يسهل تذكرها. ويوجد مترجم يحول أوامر لغة التجميع إلى الصففر والواحد حتى يفهمها الجهاز وينفذها. وتعامل أوامر لغة التجميع مع مكونات جهاز الحاسب الداخلية مباشرة.

اللغات العالية المستوى :

ثانياً

اللغات عالية المستوى هي لغة برمجة تخفي تفاصيل تركيب الحاسب الداخلية عن المبرمج، وهي بعكس لغات البرمجة منخفضة المستوى، ولا تتوافق أوامر اللغات عالية المستوى وتعليماتها مع التركيب الداخلي للحاسب. وتعد هذه اللغات أسهل في الاستخدام وعملية كتابة البرامج من اللغات منخفضة المستوى. ويوجد كثير من لغات البرمجة عالية المستوى تستخدم لبرمجة تطبيقات في مجالات مختلفة، كما تسمى اللغات العالية المستوى أيضاً باللغات المتقدمة.

وتنقسم لغات البرمجة العالية المستوى إلى:

١ - لغة البرمجة الإجرائية :

تعتمد لغة البرمجة الإجرائية على قيام المستخدم بكتابة البرنامج مفصلاً إجراءً بعد إجراء، ومن ثمَّ تحديد التعليمات التي تؤدي إلى قيام الحاسب بالمهام المطلوبة للبرنامج خطوة بعد أخرى، وفي هذه اللغات تُجمَع الأوامر التي تنفذ مهمة واحدة تحت إجراء واحد يعطى اسماً معيناً، ثم في كل مرة نريد تنفيذ هذه المهمة يُستدعى هذا الاسم الذي اخترناه بدلاً من كتابة جميع هذه الأوامر مرة أخرى، وتمتاز البرامج المكتوبة باللغات الإجرائية بأنها قصيرة وسهلة القراءة والفهم.





### أمثلة لبعض اللغات الإجرائية :

**لغة (بيسك) (BASIC):** وهي من أسهل اللغات من حيث التعليم والاستخدام، وقد طُوِّرت عام ١٩٦٤م، وانتشرت بشكل واسع في الثمانينيات من القرن السابق. وتم تطويرها حالياً للغة فيجول بيسك (Visual Basic) التي صارت تصنف ضمن لغات البرمجة بالكائنات.

**لغة (سي) (C):** وهي لغة برمجة عامة طُوِّرت عام ١٩٧٣م، وتستخدم لكتابة برمجيات نظم التشغيل، كما تستخدم لكتابة تطبيقات مختلفة أخرى.

### ٢ لغة البرمجة بالكائنات :

وتسمى أيضاً البرمجة بالعناصر، وفي هذه اللغات تُجَمَّع البيانات مع الأوامر الخاصة بها تحت كائن واحد، يعطى اسماً معيناً، ويمكن استخدام هذا الكائن عدة مرات داخل البرنامج، ويتم في لغة البرمجة بالكائنات تعديل وإعداد البرنامج بالرؤية المباشرة للكائنات التي يتألف منها البرنامج مثل : الجداول أو الشاشات أو الرموز الصورية، ولذا فإن البرمجة بالكائنات لا تتطلب من المستخدم كتابة البرنامج على شكل خطوات إجرائية محددة أو كتابة أوامر وتعليمات متتابعة، وإنما يستخدم جهاز الفأرة باختيار وتحريك وتجميع كائنات البرنامج، ويقوم الحاسب بناءً على ذلك بإنشاء التعليمات والأوامر تلقائياً وتنفيذها؛ مما يعني أن البرامج المصممة من هذا النوع ليست ملزمة بالتقيد بقيود البرمجة الإجرائية، فهي تتألف من عديد من الكائنات تمثل مجموعة من البرامج الفرعية المجزأة كالجداول والشاشات التي تستجيب لأحداث معينة (مثل حدث النقر أو النقر المزدوج أو التحميل أو الفتح أو تنشيط أداة معينة)؛ بحيث تستجيب هذه البرامج الفرعية إلى وقوع ذلك الحدث وتقوم بتنفيذ الأوامر الخاصة بالكائن.

### أمثلة للغات البرمجة بالكائنات :

**لغة (سي بلس بلس) (C++):** وهي امتداد للغة (سي) الإجرائية، وقد طورت عام ١٩٧٩م، ولكن أضيف عليها بعض الخصائص الجديدة والبرمجة بالعناصر. وتمتاز هذه اللغة بأنها يمكن استخدامها للبرمجة الإجرائية أو البرمجة بالعناصر.

**لغة (جافا) (Java):** وهي لغة برمجة بالعناصر، أي أن جميع البرامج بهذه اللغة تستخدم العناصر. وجميع هذه اللغات العالية المستوى يجب تحويلها إلى لغة الآلة حتى يتم تنفيذها بالحاسب الآلي؛ حيث إن لكل لغة مترجماً خاصاً يقوم بتحويلها للغة الآلة.

ويوضّح الجدول التالي مقارنة بين البرمجة بالكائنات و البرمجة الإجرائية :

م	البرمجة الإجرائية	البرمجة بالكائنات
١-	لغة إجرائية، أي أن البرنامج يتحكم في مسار تنفيذ البرنامج أمراً بعد أمر .	المستخدم ونظام التشغيل والبرنامج جميعها تتحكم في مسار تنفيذ البرنامج.
٢-	صعوبة إنشاء واجهة للمستخدم.	سهولة إنشاء واجهة للمستخدم.
٣-	صعوبة الربط مع قواعد البيانات المختلفة.	سهولة الربط مع قواعد البيانات المختلفة.
٤-	يتم كتابة الأوامر والتعليمات من المستخدم لتنفيذ البرنامج.	يتم استخدام الكائنات لتنفيذ البرنامج.

وختاماً نود الإشارة إلى أن لغات البرمجة بالكائنات ليس خاتمة المطاف؛ حيث يسعى علماء الحاسب مستقبلاً إلى إيجاد جيل جديد من لغات البرمجة ستكون لغات طبيعية كالتي يتخاطب بها البشر فيما بينهم مثل : اللغة العربية أو اللغة الإنجليزية، بحيث يمكنك مخاطبة الحاسب وإعطاؤه الأوامر والتعليمات مباشرة، وسوف يكون قادراً على إدراك وتفهم عباراتك وأوامرك كأن تقول له : " اطبع " أو " اعمل جدولاً " أو " احسب نتيجة الضرب لمعادلة جبرية " .



## مشروع الوحدة

### المشروع الأول :

تتعدد لغات البرمجة وتختلف من حيث استخداماتها وإمكاناتها، من خلال محركات البحث أجري بحثاً للمقارنة بين لغات البرمجة المختلفة يشتمل على الآتي:

- ١ تعريف بلغات البرمجة.
- ٢ أهم لغات البرمجة المستخدمة حالياً.
- ٣ اللغات البرمجية المناسبة لطلاب التعليم العام.
- ٤ مقارنة بين هذه اللغات من حيث السهولة والاستخدام والإمكانات.
- ٥ مراجع البحث.
- ٦ تصميم عرض تقديمي لتلخيص أهم بنود البحث.

### المشروع الثاني :

سكراتش (scratch) لغة برمجية تجعلك وبكل سهولة تنشئ قصصاً تفاعلية، رسوماً متحركة، ألعاباً، وقد تم تصميم هذه البرمجية لأهداف تربوية تعليمية تمكن الطلاب من التعلم والتعبير عن قدراتهم وتساعدهم على التغلب على الصعوبات التي قد تواجههم في فهم البرمجة وخاصة برمجة الكائنات. قم بزيارة موقع سكراتش العربي [scratch.uae.ac.ae](http://scratch.uae.ac.ae) وتحميل نسخة من البرنامج مع دليل الاستخدام والإطلاع على المشروعات والأفكار التي قام بتصميمها وبرمجتها مجموعة من الشباب المهتم بهذه البرمجية، ومن ثم قم بتصميم برمجية ، وتقديمها للمعلم وفق البنود التالية:

عنوان البرمجة	
وصف البرمجة	
الهدف من البرمجة	

## خارطة مفاهيم الوحدة

أكمل خارطة المفاهيم أدناه باستخدام العبارات والمصطلحات التي تعلمتها في الوحدة :







## دليل الدراسة



مفردات الوحدة	المفاهيم الرئيسة
مفهوم البرمجة	<ul style="list-style-type: none"> <li>لغات البرمجة هي اللغات التي يفهمها الحاسب.</li> <li>البرامج تكتب على شكل سلسلة من الأوامر التي ينفذها الحاسب.</li> </ul>
الفرق بين المبرمج ومستخدم البرنامج	<ul style="list-style-type: none"> <li>المبرمج يكتب البرنامج وفق خطوات محددة (فهم وتحليل المشكلة-كتابة سلسلة الأوامر-اختبار البرنامج - تحويل البرنامج لصيغة تنفيذية).</li> <li>مستخدم البرنامج تظهر له واجهة البرنامج دون ظهور الأوامر التي كتبها المبرمج.</li> </ul>
طرق الحصول على البرامج	<ul style="list-style-type: none"> <li>١ - شراء برنامج جاهز.</li> <li>٢ - تطوير برنامج.</li> </ul>
أهم الأوامر في البرمجة	<ul style="list-style-type: none"> <li>١ - الإدخال. ٣- الحساب.</li> <li>٢-الإخراج. ٤- التحقق من الشرط. ٥-التكرار.</li> </ul>
تعريف البرنامج	<ul style="list-style-type: none"> <li>البرنامج سلسلة من الأوامر مكتوبة بلغة برمجة معينة تكتب لتنفيذ مهمة محددة في جهاز الحاسب.</li> </ul>
أقسام لغات البرمجة	<ul style="list-style-type: none"> <li>١ - لغات عالية المستوى.</li> <li>٢ - اللغات منخفضة المستوى</li> </ul>
اللغات منخفضة المستوى	<ul style="list-style-type: none"> <li>١ - لغة الآلة.</li> <li>٢- لغة التجميع.</li> </ul>
اللغات عالية المستوى	<ul style="list-style-type: none"> <li>١ - لغة البرمجة الإجرائية ( البيسك - السي).</li> <li>٢ - لغة البرمجة بالكائنات ( الفيجول بيسك - السي بلس بلس - الجافا).</li> </ul>



## تمارين



س ١

أي من العبارات التالية غير صحيح ؟ ولماذا ؟

- ١ - يُعدّ نظام التشغيل (DOS) من أنظمة التشغيل الرسومية التي تعمل بها لغات البرمجة بالكائنات .
- ٢ - لا يفهم الحاسب الآلي إلا لغة واحدة فقط هي لغة الآلة .
- ٣ - لا يوجد فرق بين المبرمج ومستخدم البرنامج .
- ٤ - جميع لغات البرمجة العالية لها مترجم واحد يقوم بتحويل برامجها .
- ٥ - لغة الآلة يسيرة جداً وسهلة الاستخدام .
- ٦ - يتحكم مستخدم البرنامج في تنفيذ مسار البرنامج في اللغات الإجرائية .

س ٢

عرّف برنامج الحاسب . وبأي لغة تكتب هذه البرامج ؟

س ٣

تُعدّ اللغات البسيطة سهلة الفهم بالنسبة للحاسب الآلي ، فما الأسباب التي حالت دون انتشارها ؟

س ٤

ما البرمجة بالكائنات ؟ وما مزايا البرمجة بالكائنات عن البرمجة الإجرائية ؟

س ٥

حدّد ما إذا كانت اللغات التالية إجرائية أو برمجة بالكائنات :

- فيجول بيسك . - سي . - بيسك . - جافا .

س ٦

اضرب أمثلة للأحداث التي يقوم مستخدم الحاسب بإثارتها في لغة البرمجة بالكائنات .

س ٧

عدّد أهم الأوامر في البرمجة .



## اختبار



اختر رمزا لإجابة الصحيحة فيما يلي :

١ يسر يقوم بالتأكد من صحة البرنامج ويحول البرنامج لصيغة تنفيذية:

أ - البرنامج. ب - المبرمج.

ج - مستخدم البرنامج. د - جهاز الحاسب.

٢ يسر ( مساحة المستطيل = العرض "ع" × الطول "ل" ) تعد هذه العبارة من أوامر الحاسب وهو أمر :

أ - الإدخال. ب - الإخراج.

ج - الحساب. د - التكرار.

٣ يسر ( اطبع مساحة المستطيل ) تعد هذه العبارة من أوامر الحاسب وهو أمر :

أ - الإدخال. ب - الإخراج.

ج - الحساب. د - التكرار.

٤ يسر يتم فيها كتابة الأوامر على هيئة اختصارات مفهومة باللغة الإنجليزية :

أ - لغة التجميع. ب - لغة الآلة.

ج - لغة البيسك. د - لغة الجافا.

٥ يسر لغة يتم فيها تحكم المستخدم ونظام التشغيل في مسار تنفيذ البرنامج :

أ - البيسك. ب - الآلة.

ج - التجميع. د - الجافا.

٦ يسر تعد من اللغات التي لا تحتاج إلى مترجم :

أ - الفيغول بييسك. ب - الجافا.

ج - الآلة. د - سي بلس بلس.

٧ يسر لغة تمتاز بقصر أوامرها إضافة إلى سهولة قراءتها وفهمها:

أ - الفيغول بييسك. ب - التجميع.

ج - الآلة. د - البيسك.



## الوحدة الخامسة

# صياغة حل المسائل

### موضوعات الوحدة :

- الهدف من تعلم صياغة حل المسائل.
- خطوات حل المسائل.
- فهم المسألة وتحليل عناصرها.
- كتابة الخطوات الخوارزمية.
- مخططات الانسياب.



## بعد دراستك لهذه الوحدة سوف تحقق الأهداف التالية :

- تعرف الهدف من تعلم صياغة حل المسائل.
- تعدّد الخطوات الأساسية لصياغة حل المسائل .
- تشرح الخطوات الأساسية لكتابة البرنامج وتنفيذه.
- تكتب الخطوات الخوارزمية للمسائل البرمجية.
- ترسم مخططات الانسياب للمسائل البرمجية.
- تحل مجموعة من المسائل البرمجية باتباع طريقة صياغة حل المسائل.

## الأهمية :

لعلك تشاهد برامج الحاسب المختلفة وتتساءل كيف يمكن إنتاج هذه البرامج، وكيف يمكنني أن اصمم برنامج حاسوبي، وماذا علي القيام به؟  
في هذه الوحدة سنجيب عن هذه التساؤلات، وستتعرف بمشيئة الله على خطوات مرتبة ومحددة لحل المسائل أو المشكلات، والتي إذا اتبعتها ستصبح قادراً على إنتاج وتصميم برامج حاسوبية مميزة، وستساعدك على اكتساب القدرات العقلية والإبداعية مع التفكير المنطقي الاستقرائي والاستنباطي.

## مقدمة

١-٥



عندما تواجهنا مسائل أو مشكلات نحتاج إلى حلها في حياتنا اليومية فإننا نحتاج إلى فهم المشكلة وتحديد تفاصيلها، ومن ثم جمع المعلومات عنها، واختيار أفضل طرق الحل لاتباعها، فالتخطيط والتفكير لحل المشكلات والتعامل معها يحقق لنا النجاح والتغلب على المشكلات في حياتنا اليومية.

إن الحاسب لا يستطيع حل جميع المسائل أو المشكلات التي تواجهنا في حياتنا اليومية، وإن كانت يسيرة في نظرنا، كتحديد الوجبة المفضلة لديك أو لدى زميلك، أو اتخاذ قرار ما، فمثل هذه المسائل لا يمكن للحاسب أن يحلها، ولكنه عندما يقوم بحل المسائل المتعلقة بالأرقام ومعالجتها، أو تحرير النصوص، أو معالجة الصور والأصوات، فهذا يعد مجالاً خصباً للحاسب لا تجاريه في ذلك أية آلة أخرى.

## الهدف من تعلم صياغة حل المسائل

٢-٥

إننا عندما نتعلم ونتدرب على صياغة حل المسائل بواسطة الحاسب، فإن هذا لا يعني أن الفائدة تقتصر على المسائل الحسابية والمنطقية فحسب، بل إننا نهدف من تعلم هذا الموضوع إلى :

٣

القدرة على التفكير لحل المشكلات

٢

التخطيط لحياتك اليومية

١

القدرة على كتابة برامج للحاسب

## خطوات حل المسائل

٣-٥

حل المسائل مهارة تكتسب كغيرها من المهارات الأخرى بالتدرب عليها. ولحل المسائل أو المشكلات بواسطة الحاسب لتكتمل في النهاية على شكل برنامج يستطيع الحاسب فهمه والتعامل معه، فإن هناك خطوات ومراحل يجب اتباعها في حل تلك المسائل وهي:



### صياغة حل المسألة :

أولاً

المقصود بصياغة الحل هو تحديد الخطوات اللازمة لضمان الوصول للحل الصحيح. وتتكون هذه الصياغة من ثلاث خطوات أساسية، هي:

- ١ فهم المسألة وتحديد عناصرها.
- ٢ كتابة الخوارزم والخطوات المنطقية للحل.
- ٣ التمثيل البياني للخوارزم عن طريق مخططات الانسياب.

#### إثارة التفكير

لماذا علينا التخطيط لكتابة برنامج بواسطة الحاسب الآلي؟

### كتابة البرنامج وتنفيذه :

ثانياً

تتكون هذه المرحلة من ثلاث خطوات أساسية، هي:

- ١ كتابة البرنامج بواسطة إحدى لغات البرمجة من قبل المبرمج.
  - ٢ ترجمة البرنامج إلى لغة الآلة وتنفيذه، وهذا هو دور الحاسب الآلي.
  - ٣ اختبار البرنامج وإصلاح الأخطاء.
- وسوف نتطرق في هذه الوحدة من الكتاب للشرح التفصيلي للمرحلة الأولى، أما المرحلة الثانية فسوف يتم التطرق إليها بالتفصيل في الوحدة التالية.

### ٤-٥ فهم المسألة وتحليل عناصرها

لا يمكن للمرء حل مسألة ما لم يكن متأكداً من فهمها بشكل كامل وكما يقال : " فهم المسألة ثلث الحل ". والمقصود بفهم المسألة وتحليل عناصرها أن نعرف ماذا نريد بالضبط من البرنامج، وأن نحدد العناصر الأساسية لحل المسألة، وهي:

- ١ مخرجات البرنامج: النتائج والمعلومات المراد التوصل إليها عند حل المسألة.
- ٢ مدخلات البرنامج: المدخلات والبيانات اللازم الحصول عليها لمعرفة النتائج والمخرجات.
- ٣ عمليات المعالجة: العمليات الحسابية والخطوات المنطقية التي نقوم بإجرائها على مدخلات البرنامج حتى تؤدي في النهاية إلى المخرجات والنتائج.



وهذه العناصر الثلاثة هي العمليات الأساسية المبني عليها جهاز الحاسب، وبتحديد هذه العناصر نستطيع كتابة خوارزم البرنامج أو رسم مخطط الانسياب بشكل سليم. وفهم طريقة تحليل المسألة، إليك المثال التالي:

مثال ٥

نفترض أننا نريد حساب مساحة المستطيل بمعلومية الطول والعرض، حلّ عناصر المسألة إذا علمت أن مساحة المستطيل = الطول × العرض.

الحل:

لتحليل عناصر المسألة نحدّد التالي:

- ١ المخرجات، وهي مساحة المستطيل.
- ٢ المدخلات، وهي الطول والعرض.
- ٣ عمليات المعالجة، وهي قانون مساحة المستطيل.

## ٥-٥ كتابة الخطوات الخوارزمية

**الخوارزمية:** مشتقة من اسم عالم الرياضيات المسلم أبي جعفر محمد بن موسى الخوارزمي (المتوفى سنة ٨٢٥م) وصاحب كتاب (الجبر والمقابلة)، وهو أول من استعمل الطريقة الخوارزمية لحل المعادلات الجبرية.

**الخوارزمية:** هي مجموعة من الأوامر المكتوبة بصورة واضحة ومبسطة ومتتابعة منطقياً لحل مسألة.

## ١-٥-٥ كتابة الخطوات الخوارزمية لمسألة ما :

للتعرف على كيفية كتابة خطوات خوارزمية لمسألة ما، لنستعرض المثالين التاليين :

مثال ١

اكتب الخطوات الخوارزمية لقراءة عدد وتحديد ما إذا كان سالباً أو موجباً .

الحل:

- ١ أدخل العدد (A).
- ٢ إذا كان العدد  $0 < (A)$ ، اطبع عبارة (العدد موجب) وانتقل للخطوة رقم (٥).
- ٣ إذا كان العدد  $0 > (A)$ ، اطبع عبارة (العدد سالب) وانتقل للخطوة رقم (٥).
- ٤ اطبع عبارة (العدد مساوٍ للصفر).
- ٥ النهاية.



مثال ٢

اكتب الخطوات الخوارزمية لإيجاد مجموع الأعداد من ١ إلى ١٠

**الحل:** الخطوات الخوارزمية هي:

١. ضع قيمة  $(S) = 1$  ، المجموع  $(M) = 0$
٢. أضف  $(S)$  على  $(M)$  أي :  $S + M = M$
٣. قم بزيادة  $S$  بواحد صحيح أي  $1 + S = S$
٤. إذا كانت  $(S) < 10$  اطبع  $(M)$  وتوقف.
٥. ارجع إلى الخطوة رقم (٢).

### خواص الخوارزمية السليمة :

٥-٥-٢

لكي تكون الخطوات الخوارزمية سليمة لابد أن تحتوي على خواص أساسية، وهي:

١. أن تكون كل خطوة معرّفة جيداً ومحددة بعبارات دقيقة.
٢. أن تتوقف العمليات بعد عدد محدد من الخطوات.
٣. أن تؤدي العمليات في مجملها إلى حل المسألة.

مثال ١

تأكد من أن المسألة التالية تتبع الخطوات الخوارزمية السليمة لحساب وزنك المثالي بالكيلو جرام.

١. احسب الوزن المثالي  $(Y) = 100 - X$

٢. اطبع  $(Y)$

**الحل:**

الخطوات الخوارزمية بمجملها لا تؤدي إلى حل المسألة. من المفترض إضافة الخطوة التالية في البداية :

١. أدخل قيمة الطول  $(X)$  بالسنتيمتر.



تأكد من أن المسألة التالية تتبع الخطوات الخوارزمية السليمة لعرض التخصصات الجامعية المناسبة لك بناءً على تخصصك في المرحلة الثانوية (علمي - أدبي):

- ١ - أدخل التخصص (X).
- ٢ - اطبع عبارة (علوم الحاسب والمعلومات - العمارة والتخطيط - علوم الأغذية والزراعة - العلوم الهندسية - كليات صحية)، ثم توقف.
- ٣ - اطبع عبارة (الآداب - التربية - اللغة العربية - اللغات والترجمة - الحقوق - السياحة والآثار) ثم توقف.

**الحل:**

الخطوات الخوارزمية بمجملها لا تؤدي إلى حل المسألة. من المفترض إضافة عبارة (إذا كان التخصص (X) = علمي) في بداية الخطوة الثانية.

## ٦-٥ مخططات الانسياب

بعد أن نتأكد من أن الخطوات الخوارزمية سليمة، وقبل ترجمة الخطوات إلى إحدى لغات البرمجة علينا أن نرسم مخطط الانسياب لهذه الخطوات الخوارزمية، فما مخططات الانسياب؟

**مخططات الانسياب: هي تمثيل بياني أو رسمي للخطوات الخوارزمية، وتكمن الفائدة من رسم هذه المخططات في النقاط التالية:**

- ١ - توضيح الطريق التي يمر بها البرنامج ابتداء من المدخلات أو البيانات، ومن ثم المعالجة، وأخيراً مخرجات البرنامج ونتائجه.
- ٢ - توثيق منطق البرنامج للرجوع إليه عند الحاجة، وذلك بغرض إجراء أي تعديلات على البرنامج، أو اكتشاف الأخطاء التي تقع عادة في البرامج وخاصة الأخطاء المنطقية.



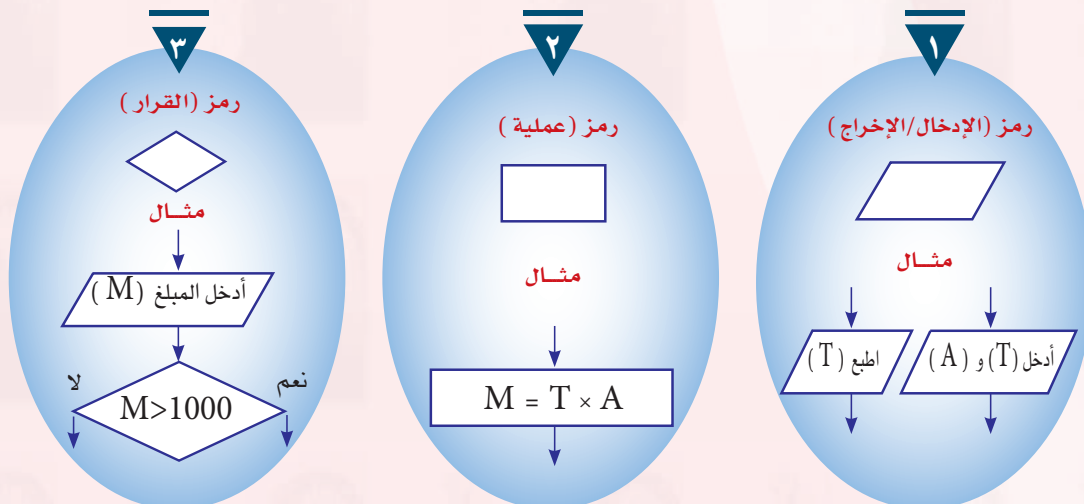
## رموز تمثيل مخططات الانسياب ١-٦-٥

لتمثيل مخططات الانسياب بيانياً نحتاج إلى مجموعة من الرموز التي يستخدم كل شكل منها للدلالة على وصف معين كما يوضح الشكل (١-٥)، وفي داخل هذه الأشكال تكتب عمليات الإدخال أو المعالجة أو الإخراج للدلالة على العملية المطلوبة.

الرمز	الاسم	المعنى
	بداية / نهاية	يمثل بداية أو نهاية البرنامج.
	إدخال / إخراج	يمثل إدخال البيانات أثناء البرنامج أو إخراجها.
	عملية	يمثل عملية معالجة للبيانات.
	قرار	يمثل اتخاذ قرار أو تعبير منطقي يحتاج إلى جواب.
	خط انسياب	يمثل اتجاه الانسياب المنطقي للبرنامج.
	توصيلة	لتوصيل الأجزاء المختلفة في المخطط.

شكل (١-٥) : رموز تمثيل مخططات الانسياب ومعانيها

### أمثلة لرسم بعض النماذج في مخططات الانسياب

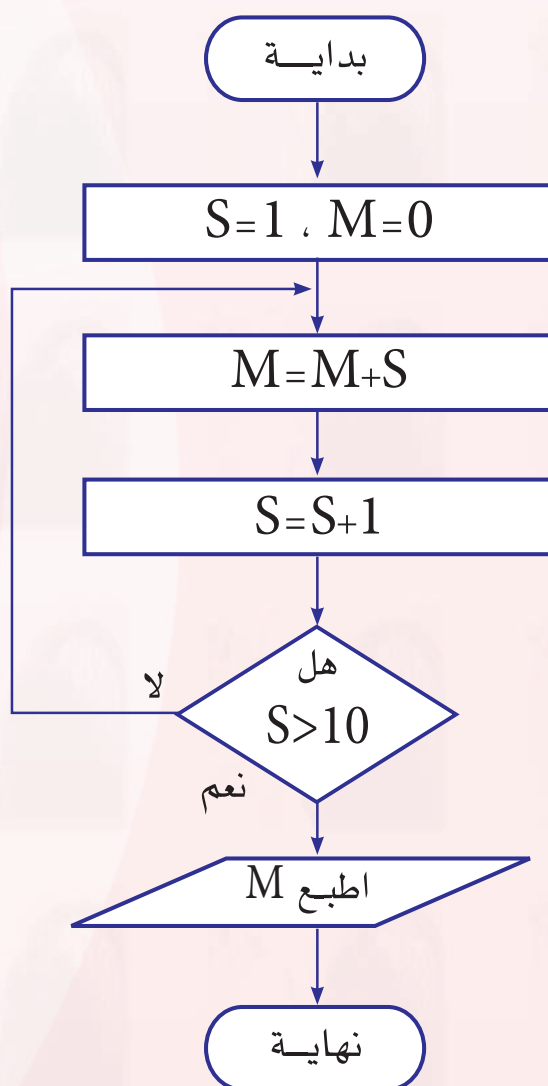


## ٢-٦-٥ رسم مخطط انسياب لمسألة ما :

### إثارة التفكير

لماذا تستخدم المخططات الانسيابية  
دون الاكتفاء بالخوارزميات؟

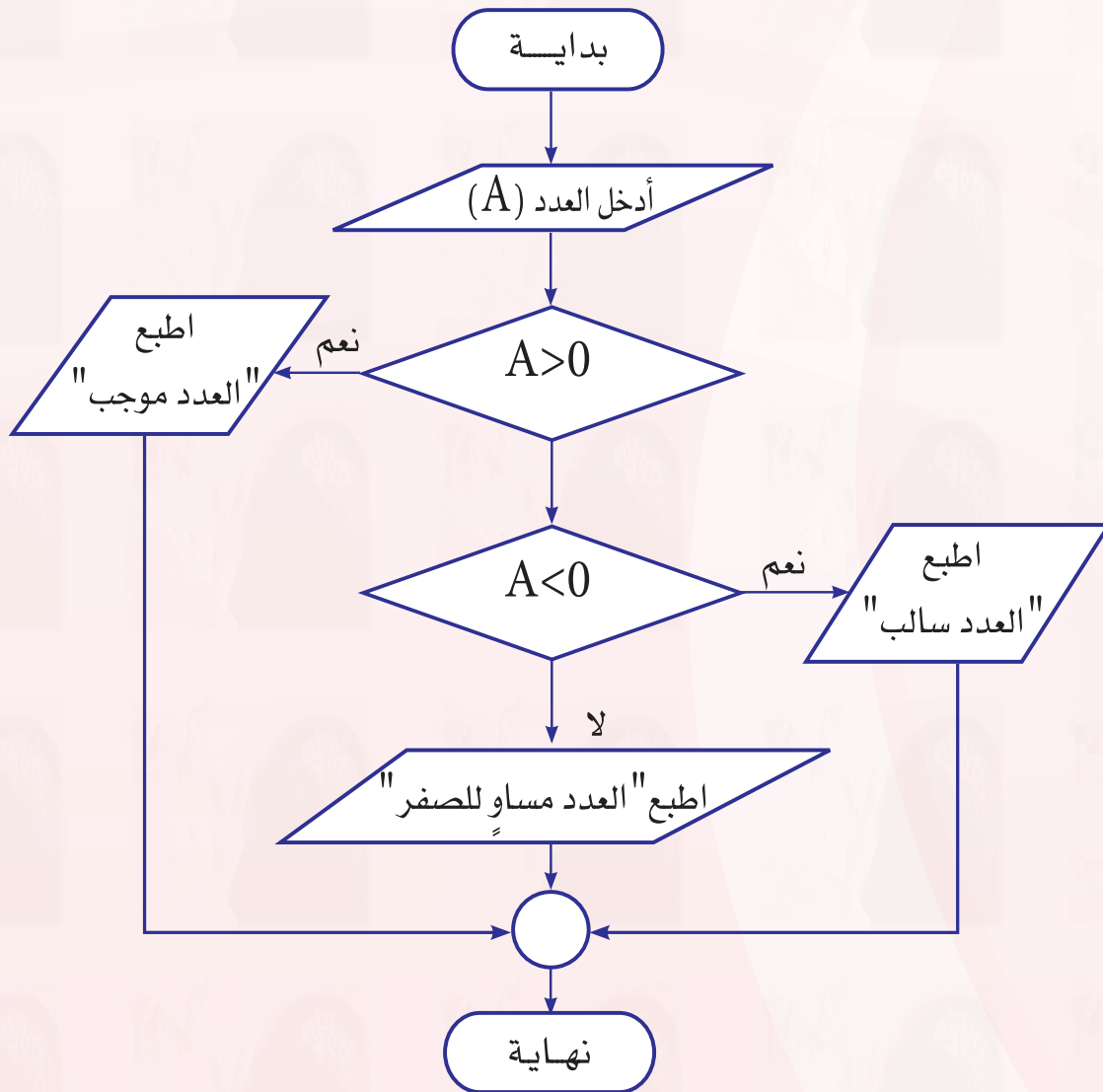
يوضح الشكل (٢-٥) طريقة استخدام مخططات الانسياب للخطوات  
الخوارزمية لحل مسألة مجموع الأعداد من (1) إلى (10).



شكل (٢-٥) : مخطط انسياب مجموع الأعداد من (1) إلى (10)



يوضح الشكل (٣-٥) طريقة استخدام مخططات الانسياب للخطوات الخوارزمية لحل مسألة قراءة عدد وتحديد ما إذا ما كان سالباً أو موجباً.



شكل (٣-٥) : مخطط انسياب لتحديد العدد سالب أم موجب

## ٧-٥ أمثلة على صياغة حل المسائل

### المثال الأول :

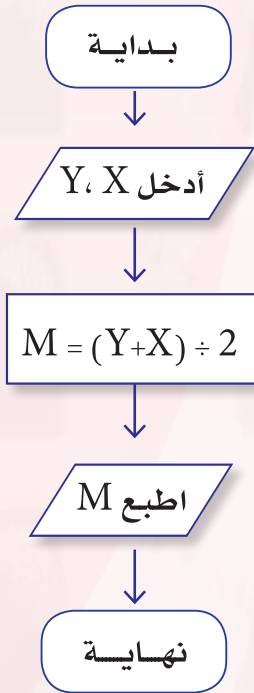
ما صياغة الحل لإيجاد متوسط عددين؟

الحل :

أولاً : فهم المسألة وتحليل عناصرها، وذلك بتحديد الآتي :

- ١ مخرجات البرنامج: متوسط العددين المدخلين، ولنرمز له بـ (M).
- ٢ مدخلات البرنامج: عدداً مدخلان، ولنرمز لهما بـ (Y ، X).
- ٣ عمليات المعالجة: قانون متوسط عددين = (العدد الأول + العدد الثاني) ÷ 2 أو بمعنى (Y + X) ÷ 2

ثانياً : كتابة الخطوات الخوارزمية للمسألة، وهي :



١ أدخل العددين (Y) و (X)

٢ اجعل  $M = (Y + X) \div 2$

٣ اطبع (M)

٤ نهاية البرنامج.

ثالثاً : رسم مخطط الانسياب لمسألة :

كما في الشكل (٥-٤).



تفكر

ليس هناك طريقة محددة لصياغة حل جميع المسائل.

شكل (٥-٤) : مخطط انسياب طباعة متوسط عددين



### المثال الثاني :

ما صياغة الحل لطباعة الأعداد الزوجية من (2) إلى (50) ؟

الحل :

أولاً : فهم المسألة وتحليل عناصرها، وذلك بتحديد الآتي :

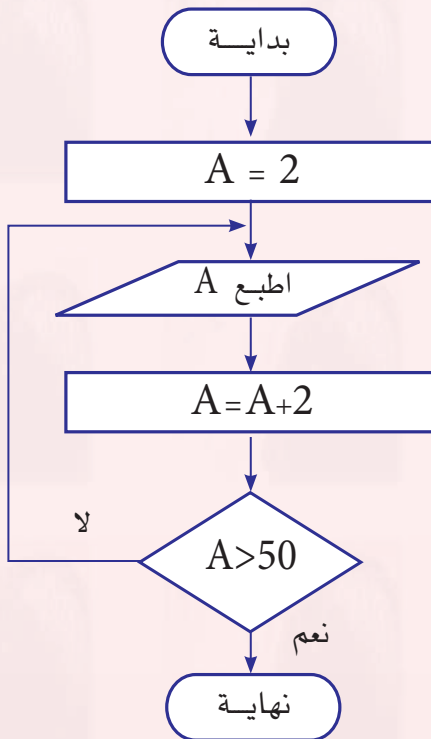
- ١ - مخرجات البرنامج: طباعة الأعداد الزوجية من (2) إلى (50).
- ٢ - مدخلات البرنامج: لا توجد مدخلات.
- ٣ - عمليات المعالجة: الانتقال من عدد زوجي إلى عدد زوجي آخر.

ثانياً : كتابة الخطوات الخوارزمية للمسألة، وهي :

- ١ - اجعل  $A = 2$
- ٢ - اطبع  $A$
- ٣ - اجعل  $A = A + 2$
- ٤ - إذا كانت  $A < 50$  توقف، وإلا اذهب إلى الخطوة رقم (٢).

ثالثاً : رسم مخطط الانسياب لمسألة :

كما في الشكل (٥-٥) .



شكل (٥-٥) : مخطط انسياب طباعة الأعداد الزوجية

## المثال الثالث :

ما صياغة الحل لإيجاد متوسط درجات طلاب فصل في مادة الحاسب الآلي، إذا علمت أن عددهم (10) طلاب؟

الحل :

فهم المسألة وتحليل عناصرها، وذلك بتحديد الآتي :

أولاً

- ١ - مخرجات البرنامج: متوسط الدرجات، ولنرمز له بـ (S).
- ٢ - مدخلات البرنامج: درجات (10) طلاب ولنضعها في متغير رمزه (D).
- ٣ - عمليات المعالجة: إيجاد المتوسط  $S = \text{مجموع الدرجات} (M) \div 10$

ونلاحظ أننا في هذا المثال لا يمكن أن نحسب المتوسط إلا بعد إيجاد مجموع الدرجات، كما أننا نحتاج إلى عدّاد يقوم بعدّ الطلاب حتى يصل عددهم إلى (10) طلاب، ولذا فإن هذه المسألة تحتوي على عمليات معالجة وشروط لا يمكن لنا أن نكتب البرنامج بدونها وهي:

أ - وجود العدّاد (I)؛ حيث نضع شرطاً على هذا العداد وهو: إذا كان العدّاد  $= 10$  استمر في البرنامج، وإلا قم بتزويد العدّاد.

ب - إيجاد مجموع درجات الطلاب وتخزينه في مكان (M)، حيث نضع صفرًا كقيمة أولى للمكان (M)، بعد ذلك نقوم بإضافة درجات الطلاب من المكان (D) إلى المكان (M) وذلك عن طريق المعادلة التالية:

$$M = (M \text{ القديمة}) + D$$

كتابة الخطوات الخوارزمية للمسألة، وهي :

ثانياً

- ١ - ضع المجموع  $0 = (M)$
- ٢ - اجعل عدّاد  $0 = (I)$
- ٣ - اجعل عداد  $1 + I = I$
- ٤ - أدخل درجة الطالب رقم (I) و خزنها في (D).





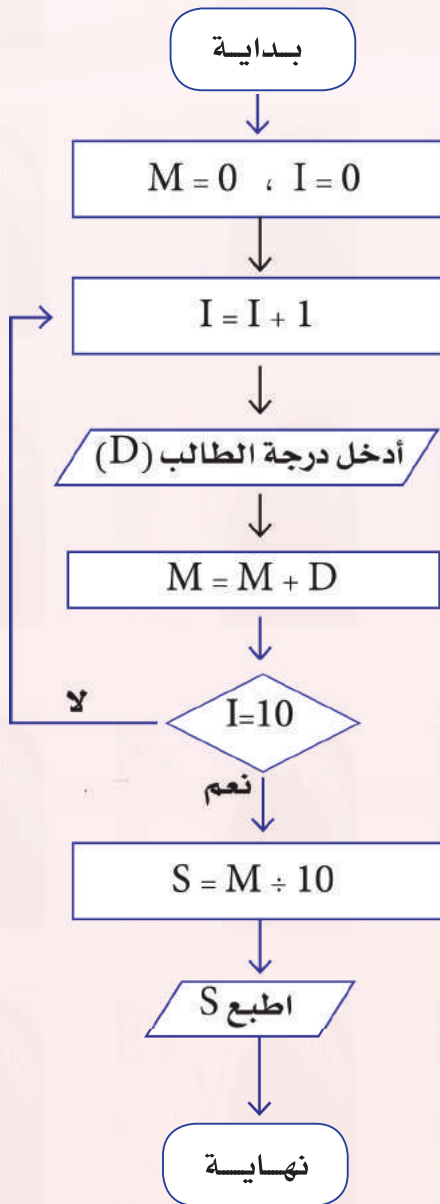
اجعل  $D+M = M$  ٥

إذا كان العداد  $(I) = 10$  استمر، وإلا اذهب إلى الخطوة (٣). ٦

احسب المتوسط  $10 \div M = S$  ٧

اطبع المتوسط  $S$  ٨

نهاية البرنامج ٩



رسم مخطط الانسياب لمساواة :

ثالثاً

كما في الشكل (٥-٦).

شكل (٥-٦) : مخطط انسياب لحساب متوسط درجات الطلاب



## مشروع الوحدة

### المشروع الأول :

ما صياغة الحل لإيجاد القاسم المشترك الأكبر لعددین طبيعيين وذلك وفق الخطوات التالية :

- ١ تحديد المدخلات والمخرجات والعمليات.
- ٢ كتابة الخوارزمية للمسألة.
- ٣ رسم مخطط الانسياب للمسألة.
- ٤ تصميم عرض تقديمي للخطوات الخوارزمية وخريطة الانسياب.

### المشروع الثاني :

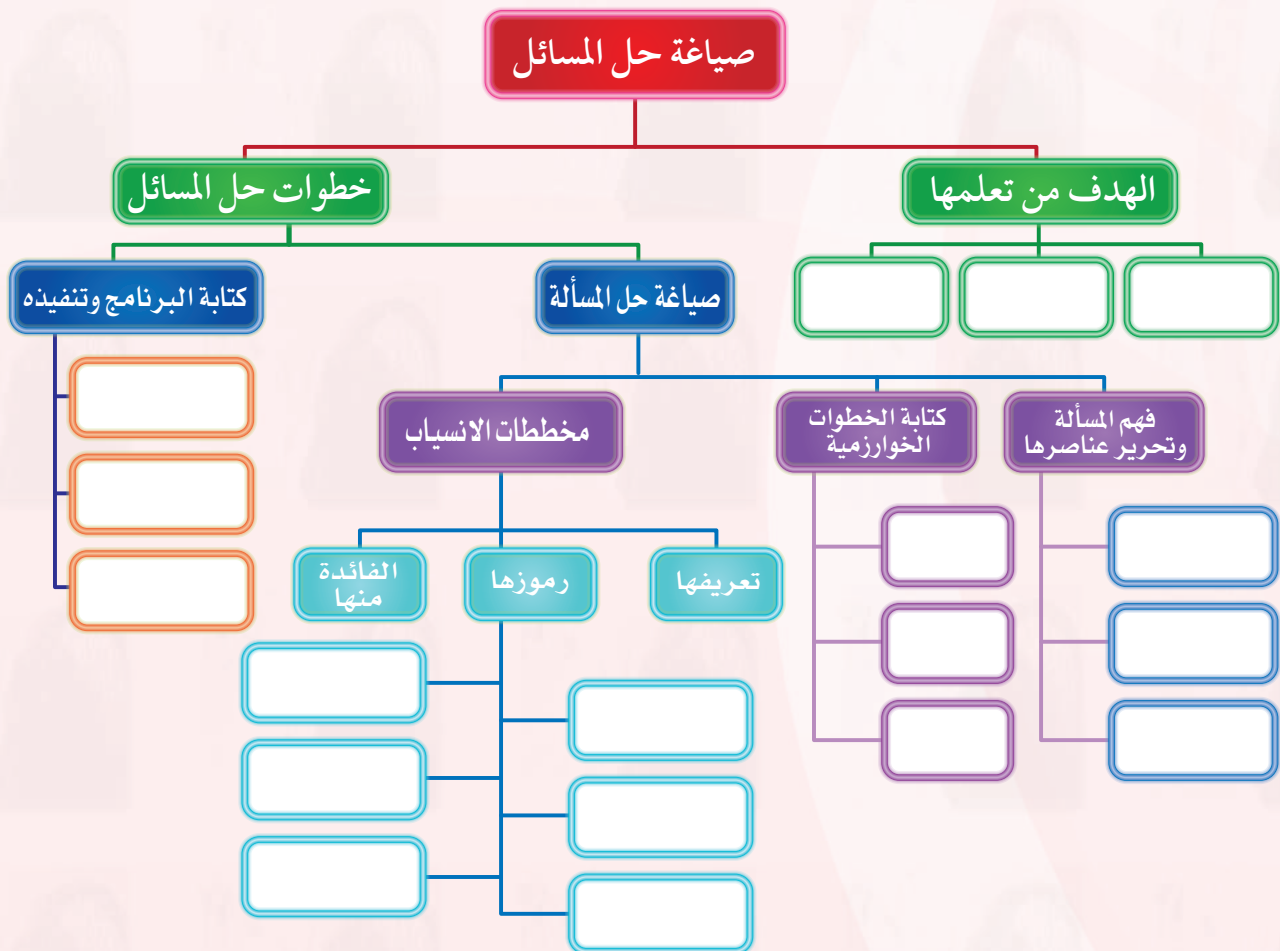
ما صياغة الحل لإيجاد زكاة بهيمة الأنعام وفق ما تعلمت في مادة الفقه، وذلك وفق الخطوات التالية :

- ١ تحديد المدخلات والمخرجات والعمليات.
- ٢ كتابة الخوارزمية للمسألة.
- ٣ رسم مخطط الانسياب للمسألة.
- ٤ تصميم عرض تقديمي للخطوات الخوارزمية وخريطة الانسياب.



## خارطة مفاهيم الوحدة

أكمل خارطة المفاهيم أدناه باستخدام العبارات والمصطلحات التي تعلمتها في الوحدة :



## دليل الدراسة



المفاهيم الرئيسية	مفردات الوحدة
<ul style="list-style-type: none"> <li>■ صياغة حل المسائل.</li> <li>■ كتابة البرنامج وتنفيذه.</li> <li>■ مخرجات البرنامج.</li> <li>■ مدخلات البرنامج.</li> <li>■ عمليات المعالجة.</li> <li>■ تعريف الخوارزمية.</li> <li>■ خواص الخوارزمية السليمة.</li> <li>■ تعريف مخططات الانسياب.</li> <li>■ الفائدة من مخططات الانسياب.</li> <li>■ رموز تمثيل مخططات الانسياب.</li> </ul>	<ul style="list-style-type: none"> <li>■ أهداف صياغة حل المسائل.</li> <li>■ خطوات حل المسائل.</li> <li>■ فهم المسألة وتحليل عناصرها.</li> <li>■ كتابة الخطوات الخوارزمية.</li> <li>■ مخططات الانسياب.</li> </ul>



## تمريعات



عدّد اثنين من أهداف تعلم صياغة حل المسائل .



ما المراحل اللازمة لحل المسائل بواسطة الحاسب الآلي؟ اشرح الخطوات الأساسية لكل مرحلة .



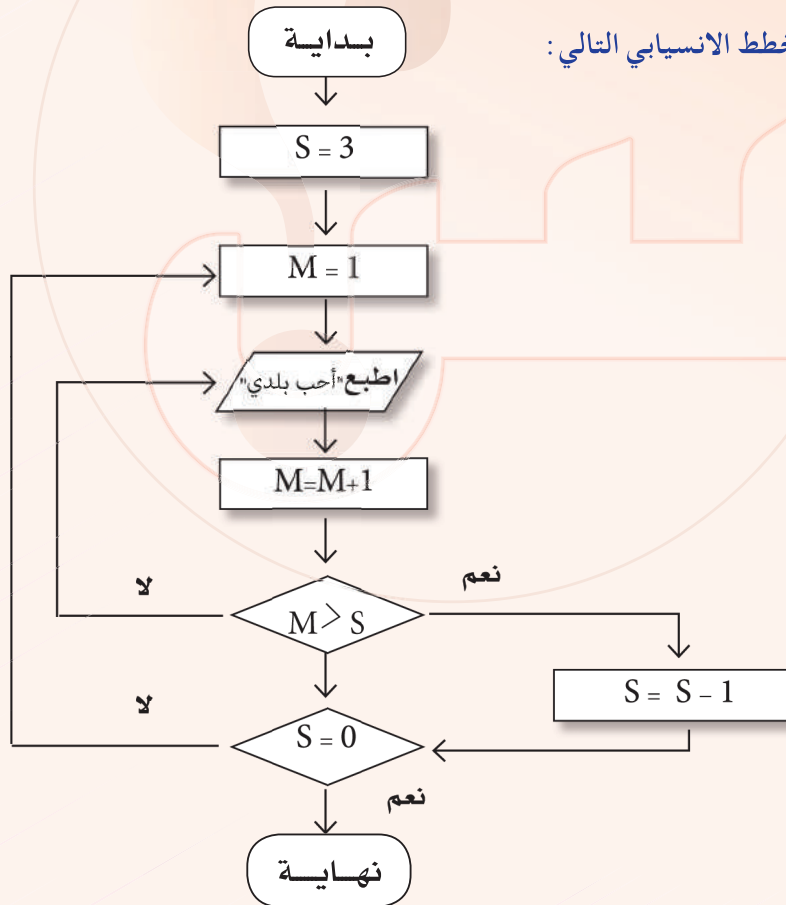
لتعريف وتحليل المسألة لا بد من تحديد عناصرها، فما هذه العناصر؟



ما الخوارزمية؟ وما خواصها الأساسية؟



حدّد مخرجات المخطط الانسيابي التالي:



٦ ما صياغة الحل لحساب عدد الأعداد الفردية ما بين 1 - 100 ؟

س٦

٧ ما صياغة الحل لإيجاد وطباعة المبلغ الإجمالي لخمس سلع بقيم مختلفة وكميات مختلفة . يتم قراءة كل سلعة وكميتها على حدة ؟

س٧

٨ ما صياغة الحل لتحديد نجاح أو رسوب طالب في مادة . علماً بأن الطالب يعد ناجحاً إذا كان مجموع أعمال السنة والامتحان النهائي > 50 ؟

س٨

٩ ما صياغة الحل لتحويل درجة الحرارة من النظام المئوي إلى الفهرنهايت ، إذا علمت أن : درجة الحرارة بالفهرنهايت =  $(9 \div 5) \times$  درجة الحرارة بالمئوي + 32 ؟

س٩

١٠ ما صياغة الحل لقراءة وطباعة تقدير طالب في مادة ما حسب الجدول التالي :

س١٠

العلامة	100-90	89-80	79-70	69 - 50	أقل من 50
التقدير	ممتاز	جيد جداً	جيد	مقبول	غير مجتاز



## اختبار

اختر رمز الإجابة الصحيحة فيما يلي :

١ القدرة على كتابة البرامج والتخطيط لحياتك اليومية والتفكير لحل المشكلات هي أهداف:

- أ - لصياغة حل المسائل.      ب- للتدريب على البرنامج.  
ج- لفتح البرامج.      د- لتحليل أنظمة التشغيل.

٢ تحديد الخطوات المتبعة للوصول إلى الحل لضمان صحة الحل هو :

- أ - صياغة حل المسألة.      ب- كتابة البرنامج.  
ج- تنفيذ البرنامج.      د- تطبيق البرنامج.

٣ النتائج والمعلومات المراد التوصل إليها عند حل المسألة هي

- أ - مخرجات البرنامج.      ب- مدخلات البرنامج.  
ج- عمليات المعالجة.      د- عمليات التنفيذ.

٤ مجموعة من الأوامر المكتوبة بصورة واضحة ومبسطة ومتراكبة منطقياً لحل المسألة هي:

- أ - مخططات الانسياب.      ب- الخوارزمية.  
ج- كتابة البرنامج.      د- تنفيذ البرنامج.

٥

١- ضع قيمة  $S = 1$  ، المجموع  $(M) = (0)$

٢- أضف  $(S)$  على  $(M)$  أي  $S+M=M$

٣- قم بزيادة  $(S)$  بواحد صحيح أي  $1+S=S$

٤- إذا كانت  $(S) < 10$  أطلع  $(M)$  وتوقف .

٥- ارجع إلى الخطوة رقم (2)

خطوات الخوارزمية السابقة لإيجاد :

- أ - ترتيب الأعداد من (١) إلى (١٠).      ب- مجموع الأعداد من (١) إلى (١٠).  
ج- الأعداد الزوجية من (١) إلى (١٠).      د- الأعداد الفردية من (١) إلى (١٠).



٦ يسعد بالرمز ( ) من رموز تمثيل مخططات الانسياب بـ

أ - بداية/نهاية. ب - عملية.

ج - إدخال/إخراج. د - قرار.

٧ يسعد بالرمز ( ) من رموز تمثيل مخططات الانسياب بـ

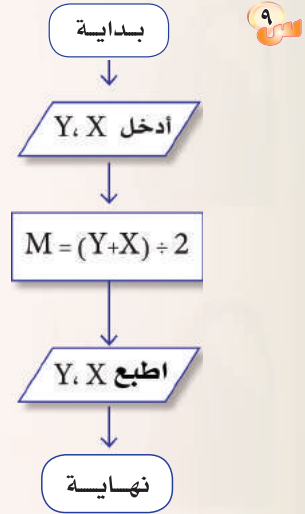
أ - بداية/نهاية. ب - عملية.

ج - إدخال/إخراج. د - قرار.

٨ يسعد بالرمز ( ) من رموز تمثيل مخططات الانسياب بـ

أ - بداية/نهاية. ب - عملية.

ج - إدخال/إخراج. د - توصيلة.



مخطط الانسياب السابق هو لطباعة :

أ - متوسط عددين. ب - مجموع عددين.

ج - حاصل ضرب عددين. د - الأعداد الزوجية.





Microsoft®  
Visual Studio®2010

## الوحدة السادسة

# البرمجة بلغة (فيجول بيسك ستوديو)

### موضوعات الوحدة :

- 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).

## بعد دراستك لهذه الوحدة سوف تحقق الأهداف التالية :

- تُعدُّ مراحل كتابة البرنامج بلغة فيجول بيسك ستديو.
- تُوضَّح طريقة تعامل برنامج فيجول بيسك ستديو مع البيانات.
- تُجرى العمليات الحسابية والمنطقية ببرنامج الفيجول بيسك ستديو.
- تستخدم أدوات البرمجة بلغة فيجول بيسك ستديو.
- تُعدُّ الأوامر الأساسية في لغة فيجول بيسك ستديو.

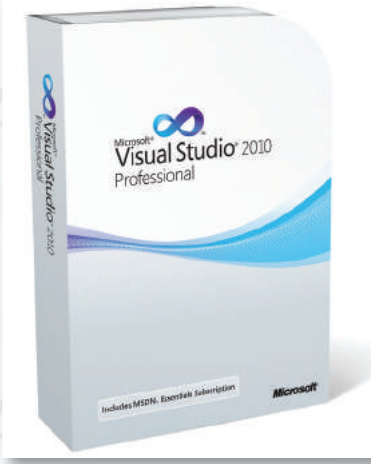
## الأهمية :

الهدف الرئيس لجهاز الحاسب هو القيام بالعمليات الحسابية ومعالجة البيانات وهذه المهمة تتطلب وجود برامج تقوم بها ، هذه البرامج يقوم بنائها المبرمجون ويحتاجون إلى منصات عمل توفر لهم أدوات قوية عند التنفيذ.

وتعد لغة فيجول بيسك ستديو من لغات المستوى العالي سهولة التعلم والتي غالباً ما ينصح المبتدئين في عالم البرمجة بتعلمها وذلك لخلوها من التعقيد واعتمادها على البرمجة بالكائنات «البرمجة الشيئية» مع مناسبتها لتطبيقات قواعد بيانات والتطبيقات المخصصة للشركات الصغيرة.

## مقدمة

١-٦



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.

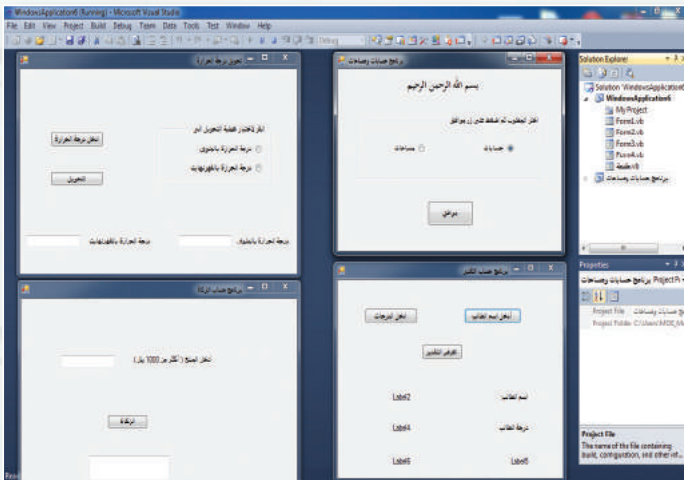
## ٢-٦ مراحل كتابة البرنامج بلغة (فيجول بيسك ستوديو)

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.



### تصميم الواجهات :

أولاً

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

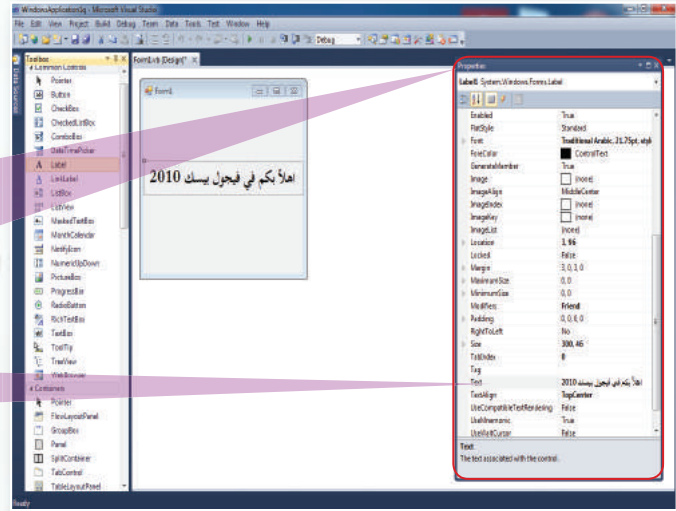




## ضبط خصائص الأدوات :

ثانياً

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 .

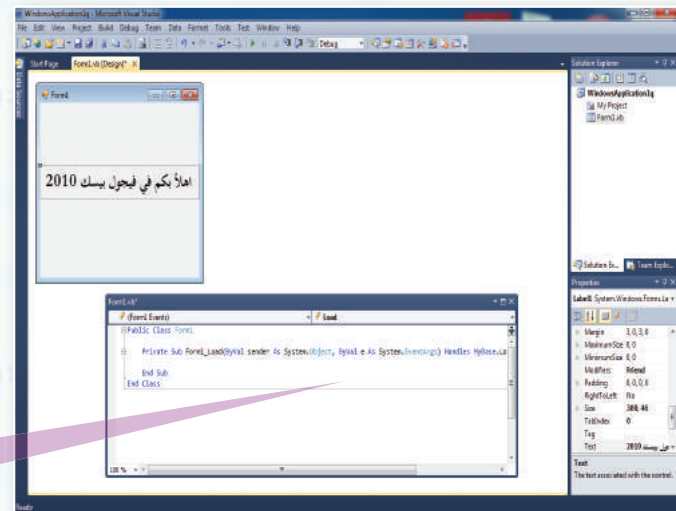
مثلاً لتغيير خاصية عنوان أداة تجد في نافذة الخصائص خاصية (Text) وهكذا لبقية الخصائص.

شكل (٢-٦) : نافذة البرنامج وخصائص الأدوات

## كتابة أوامر البرمجة :

ثالثاً

بعد أن ننهي 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



## طريقة تعامل البرنامج مع البيانات

٣-٦

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.

## الثوابت وأنواعها :

أولاً

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.**

## إثراء علمي

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 حقيقي.

## طريقة تعريف الثوابت :

تُعرف الثوابت باستخدام الأمر (Const)

**Const Const1 = Value**

حيث إن :

- ١ **Const**: الأمر الذي نستخدمه لتعريف الثابت.
- ٢ **Const1**: 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**



## المتغيرات وأنواعها :

### ثانياً

**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 يكون 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@

إثراء علمي

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 المتغير :

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, ...*

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"

## ٤-٦ العمليات الحسابية والمنطقية

### 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.

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

الحل:

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

لليمين:

$$M = 12 / 3$$

نفذ عملية الضرب أولاً:

$$M = 4$$

ثم عملية القسمة ثانياً:

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:

$$M = 2 * 6 + 9$$

نفذ عملية الأس أولاً:

$$M = 12 + 9$$

ثم عملية الضرب ثانياً:

$$M = 21$$

وأخيراً عملية الجمع:



مثال ٢٠

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، لكن يوجد أقواس حول عملية الجمع لذلك نبدأ بها:

$$M = 2 * 9^2$$

نفذ عملية الجمع أولاً:

$$M = 2 * 81$$

ثم عملية الأس ثانياً:

$$M = 162$$

وأخيراً عملية الضرب:

**العمليات المنطقية في البرمجة :**

**ثانياً**

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).

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

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



**تنبيه**

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

مثال ٢١

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

**الحل:**

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

مثال ٢٠

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

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

الحل:

مثال ٢١

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

نفذ العمليات الحسابية أولاً:

$$4 + 15 < 24$$

$$19 < 24$$

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

الحل:

## تحويل المعادلات الجبرية إلى الصيغة المستخدمة في البرمجة :

ثالثاً

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.

مثال ٢٢

حوّل المعادلة الجبرية الآتية إلى معادلة بصيغة برمجية.

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

$$Num = X^2 / (A + B)$$

الحل:

مثال ٢٣

حوّل المعادلة الجبرية الآتية إلى معادلة بصيغة برمجية.

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

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

الحل:



## ٥-٦ أدوات البرمجة باغة (فيجول بيسك ستوديو)

### أدوات البرمجة :

### أولاً

**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.

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.

### خصائص الأدوات :

### ثانياً

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.

### الخصائص المشتركة بين الأدوات :

١

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.



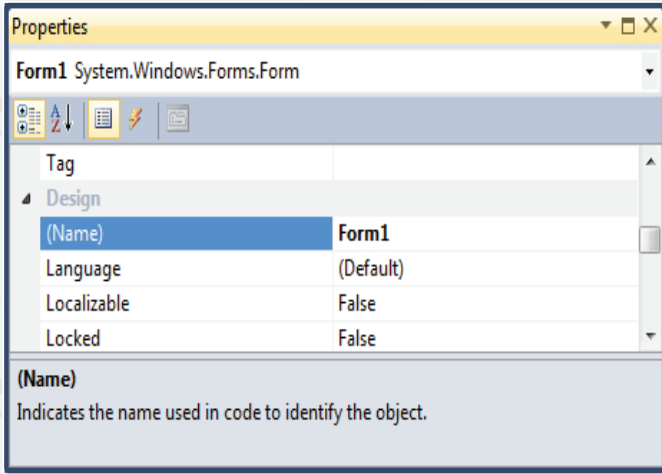
هناك طريقتان للتحكم بخصائص الأدوات، هما:

### أ ضبط خصائص الأدوات أثناء تصميم البرنامج :

نستخدم إحدى الطرق التالية:

#### ١- كتابة القيمة :

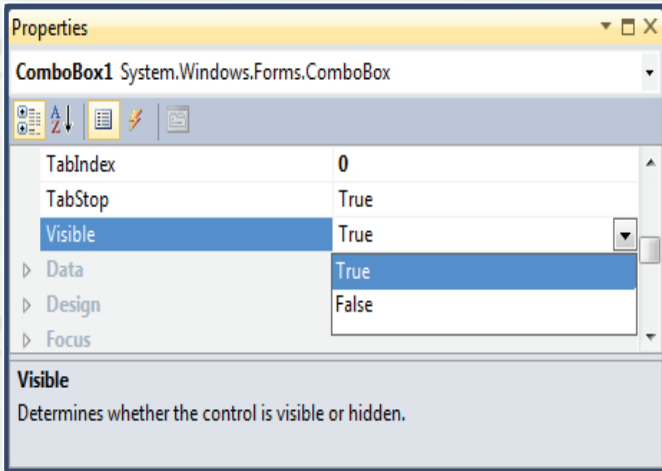
نكتب القيمة بواسطة لوحة المفاتيح في الخانة المخصصة داخل إطار الخصائص. كما في خاصية الاسم (Name) في الشكل (٤-٦)، حيث كتبنا القيمة (Form1) داخل خانة (الاسم).



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).

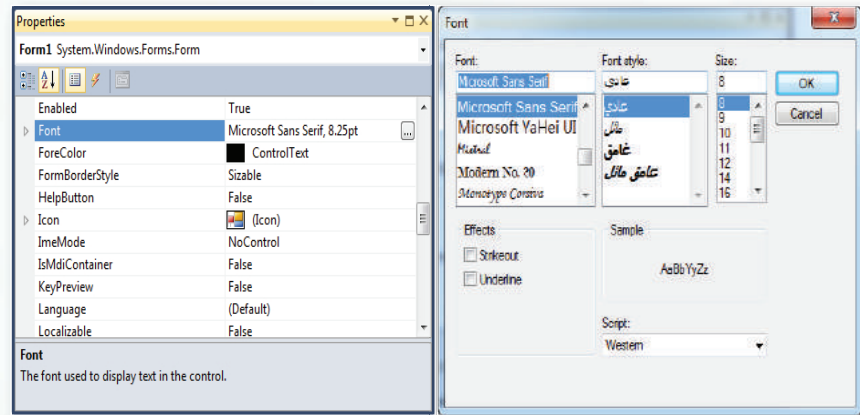


شكل (٥-٦) : خاصية المشاهدة



### 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

### إثراء علمي

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"

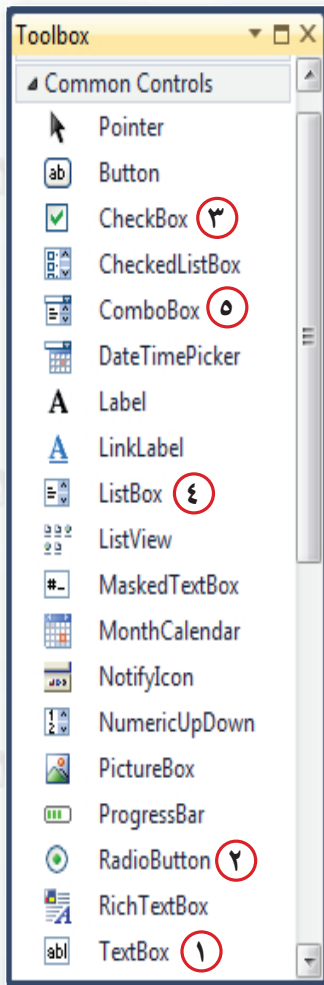
يجب وضع نقطة بين اسم الأداة والخاصية التي نريد الوصول إليها.

### أدوات إدخال البيانات :

### ثالثاً

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.



شكل (٦-٧): مربع الأدوات

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 وتخزينها في الخاصية (Text).

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

#### ١ مربع النص (Text Box):

إن البيانات التي نحصل عليها من مربع النص تختلف بحسب ما يدخله المستخدم، فقد تكون أرقامًا أو حروفًا.

للحصول على البيانات من مربع النص نستخدم الصيغة التالية:

**Var1=TextBox.Text**

حيث إن :

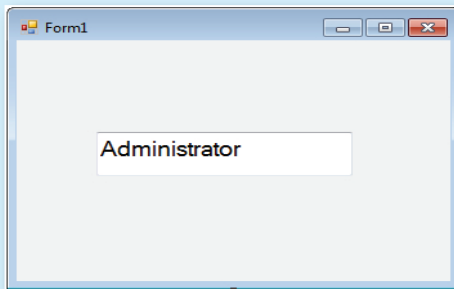
**Var1** : متغير لتخزين البيانات فيه أيًا كان نوعها عددية أم حرفية.

**TextBox** : اسم أداة مربع النص على النموذج.

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



مثال :



شكل (٨-٦) : أداة مربع النص

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"`

أداة زر الخيار (RadioButton) :

٢

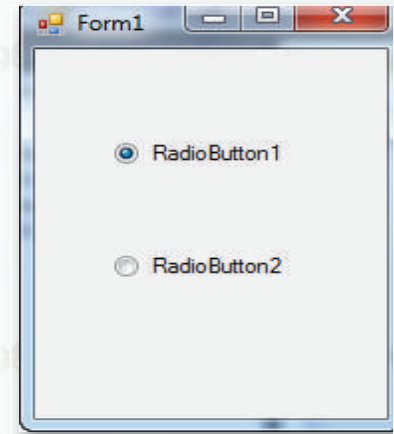
إن البيانات التي نحصل عليها من زر الخيار كما في الشكل (٩-٦) هي بيانات منطقية تخزن في الخاصية (checked) وهي إحدى قيمتين، إما:

• أن الزر قد تم اختياره فقيمه عندئذ هي (True).

• أن الزر لم يتم اختياره فقيمه عندئذ هي (False).

والصيغة العامة للحصول على البيانات هي:

`Var1=RaidoButton.Checked`



شكل (٩-٦) : أداة زر الخيار

حيث إن :

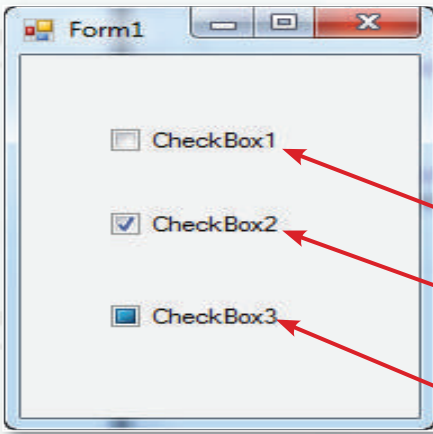
**Var1** : متغير لتخزين البيانات من نوع منطقي.

**RaidoButton** : اسم أداة زر الخيار على النافذة.

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

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.

### أداة مربع الاختيار (CheckBox) :



شكل (١٠-٦) : أداة مربع الاختيار

- 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 يختَر المربع فقيمته عندئذ هي (0).
  - أن المستخدم اختار المربع فقيمته عندئذ هي (1).
  - أن المستخدم لا يستطيع اختيار المربع فقيمته عندئذ هي (2).

والصيغة العامة للحصول على البيانات هي :

$Var1 = CheckBox.Checked$

حيث إن :

**Var1** : متغير لتخزين البيانات فيه من نوع عددي.  
**CheckBox** : اسم أداة مربع الاختيار على النافذة.  
**Checkedstate** : 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.

أداة مربع القائمة (ListBox) :

٤

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) .

الصيغة 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.

الصيغة العامة للحصول على قيمة العنصر (Text) هي:

`Var1=ListBox.Text`

حيث إن :

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

**ListBox** : اسم أداة مربع القائمة على النافذة.

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

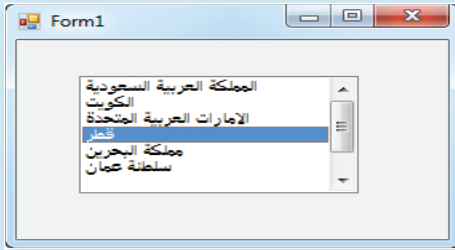
مثال ١

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

$C = \text{ListBox1.SelectedIndex}$

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

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



شكل (١١-٦) : أداة مربع القائمة

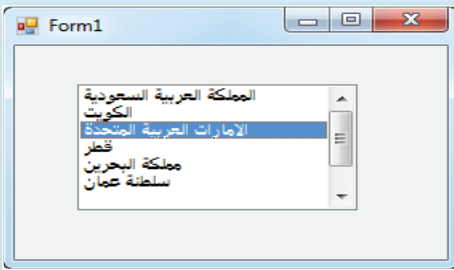
مثال ٢

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 = "الإمارات العربية المتحدة" ، كما في الشكل (١٢-٦).



شكل (١٢-٦) : أداة مربع القائمة

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 :

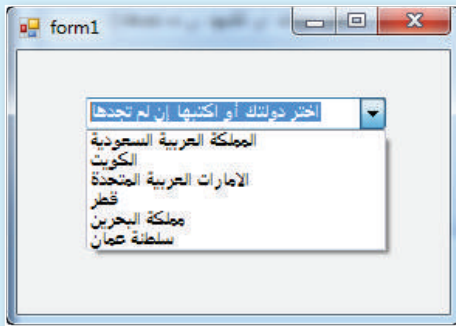
**Varl** : متغير لتخزين البيانات فيه أيًا كان نوعها عددية أم حرفية.  
**ComboBox** : اسم أداة الخانة المركبة على النافذة.  
**Text** : خاصية أداة الخانة المركبة التي تستقبل البيانات من المستخدم سواء باختياره من القائمة أو بإدخاله للبيانات بالكتابة مباشرة.

مثال :

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).



شكل (٦-١٣) : أداة الخانة المركبة

أدوات إخراج المعلومات :

رابعاً

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).

طريقة إخراج المعلومات إلى مربع النص (TextBox) :



لإخراج المعلومات إلى مربع النص نستخدم الصيغة التالية :

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

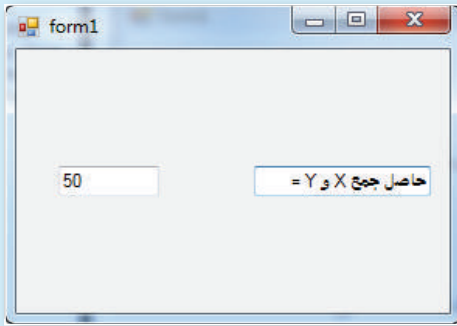
حيث إن :

**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 :



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.

شكل (١٤-٦) : استخدام أداة (Text)

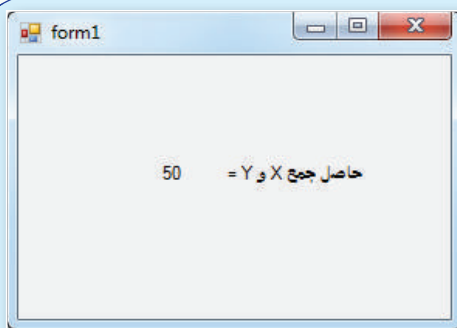
طريقة إخراج المعلومات إلى أداة التسمية (Label) :

٢

لإخراج المعلومات إلى أداة التسمية نستخدم الصيغة التالية :

**Label.Text = Var1**

مثال :



لإظهار حاصل جمع عددين (X+Y) في الأداة المسماة (Lable1) نكتب السطر التالي :

**Label1.Text = X+Y**

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

شكل (١٥-٦) : استخدام أداة (Label)



## ٦-٦ بعض الأوامر الأساسية للغة (فيجول بيسك ستوديو)

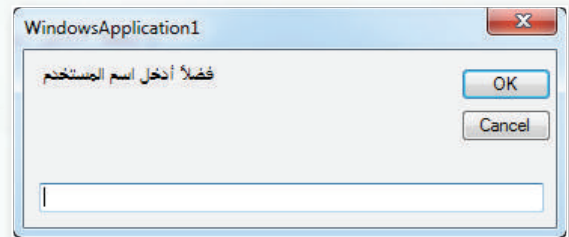
هناك أوامر داخلية في (فيجول بيسك ستوديو) تظهر للمستخدم نافذة مصممة سابقاً من قبل الشركة المنتجة للغة البرمجة، سواء لإدخال البيانات أو إخراج المعلومات، ومن هذه الأوامر:

أوامر إدخال البيانات وإخراج المعلومات :

أولاً

### إدخال البيانات بواسطة الأمر ( 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 يريدتها في مربع النص ثم يضغط على زر الأمر. لإنشاء هذه النافذة يجب أن نكتب الأمر الخاص بها، وصيغته كالتالي:



شكل (٦-١٦) : نافذة أمر (InputBox)

***Var1 = InputBox (message)***

**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 " " .

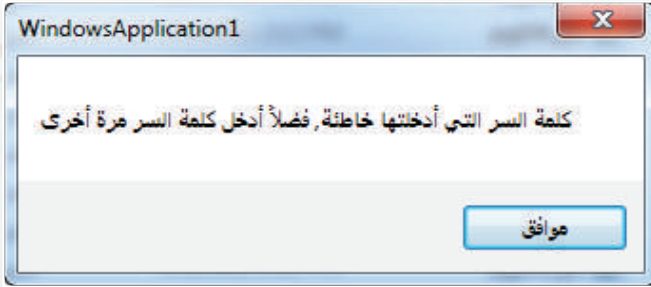
حيث إن :

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

مثال :



## 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 (٦-١٧)، ولإنشاء هذه النافذة يجب أن نكتب الأمر الخاص بها:

وصيغته 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)

أمر الإسناد :

ثانياً

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.



تنبيه

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



مثال

لو أردنا تخزين اسم في متغير حريف 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:  $X = V + 20 \times I$

الجملة الشرطية :

ثالثاً

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

الجملة 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 (٦٠) فهو ناجح، وإن كانت أقل من (٦٠) فهو غير مجتاز.

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

أ صيغة (IF-THEN) :

أ

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  $\geq$  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.

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

مثال :

## ج 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 صحيح 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

```

## الجملة الشرطية (Select Case) :

٢

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 لقيمتها أكثر من احتمال.

وصيغتها :

```

SELECT CASE expression
CASE prob1
...
statements1
...
CASE prob2
...
statements2
...
[ CASE ELSE
...
statements3
... ]
END SELECT

```





حيث إن :

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.



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.



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



لو أردنا 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
```

## حلقات التكرار :

## رابعاً

هو 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:

## الأمر ( For .. Next ) :

١

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

```
.....
statements
```

```
.....
NEXT
```

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

صيغته:



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
```

## الأمر ( DO WHILE ) :

٢

## إثارة التفكير

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

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 ١

```
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 ٢

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

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

المصفوفات :

خامساً

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؟

## فوائد المصفوفات :

١

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.

## التعامل مع المصفوفات :

٢

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

وخرنا 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:

أن تبدأ العداد من الصفر وتنتهيه  
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
```

## مشروع الوحدة

## المشروع الأول :

قم بتصميم برنامج لإيجاد القاسم المشترك الأكبر لعددتين باستخدام نظرية اقليدس.

## المشروع الثاني :

قم بتصميم برنامج لمفصلة ملابس تقوم فيه بإدخال اسم العميل ثم اختيار نوع الملابس وأسعارها ثم عرض اسم المستخدم وقائمة ملابس مع أسعارها وإجمالي فاتورته انظر الشكل للنموذج المطلوب عرضه :

اسم العميل : محمد أحمد عبدالله

الملابس :

القطعة	العدد	السعر الفردي	السعر الإجمالي
١. ثوب	٥	٣	١٥
٢. غترة	٢	٢	٤

إجمالي القطع : ٧ إجمالي السعر : ١٩

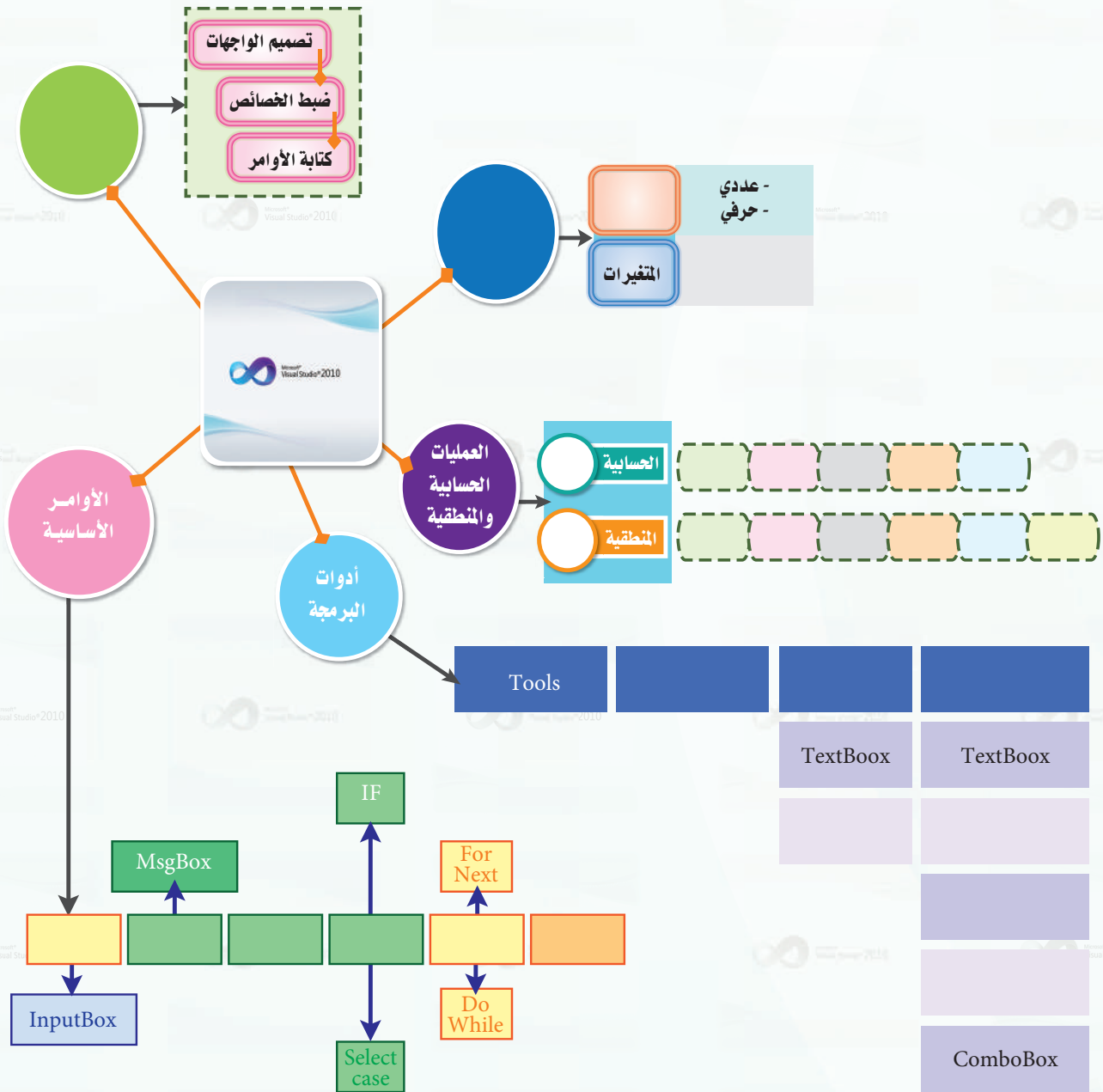
باستخدام برنامج فيجول بيسك ستوديو قم باختيار أحد المشروعات أعلاه، وكتابة تقرير عن المشروع يشمل:

- ١ مقدمة عن التطبيق (الفكرة - الهدف).
- ٢ خطوات حل المسألة.
- ٣ خوارزم البرنامج.
- ٤ صور الواجهات المصممة وعمل مكونات كل واجهة.
- ٥ النص البرمجي للبرنامج.



## خارطة مفاهيم الوحدة

أكمل خارطة المفاهيم أدناه باستخدام العبارات والمصطلحات التي تعلمتها في الوحدة :



## دليل الدراسة



مفاهيم الرئيسة	مفردات الوحدة
<ul style="list-style-type: none"> <li>■ تصميم الواجهات.</li> <li>■ ضبط الخصائص.</li> <li>■ كتابة الأوامر البرمجية.</li> </ul>	<ul style="list-style-type: none"> <li>■ مراحل كتابة البرنامج بلغة (فيجول بيسك ستديو).</li> </ul>
<ul style="list-style-type: none"> <li>■ الثوابت وأنواعها وتعريفها.</li> <li>■ المتغيرات أنواعها وتعريفها.</li> <li>■ شروط تسمية المتغيرات.</li> <li>■ أنواع البيانات.</li> </ul>	<ul style="list-style-type: none"> <li>■ تعامل (فيجول بيسك ستديو) مع البيانات.</li> </ul>
<ul style="list-style-type: none"> <li>■ العمليات الحسابية: الجمع - الطرح - الضرب - القسمة - التربيع.</li> <li>■ العمليات المنطقية: يساوي - لا يساوي - أكبر من - أقل من - أكبر من أو يساوي - أصغر من أو يساوي.</li> </ul>	<ul style="list-style-type: none"> <li>■ العمليات الحسابية والمنطقية.</li> </ul>
<ul style="list-style-type: none"> <li>■ الأدوات «Tools».</li> <li>■ الخصائص «Properties».</li> <li>■ أدوات إدخال البيانات.</li> <li>■ أدوات إخراج المعلومات.</li> </ul>	<ul style="list-style-type: none"> <li>■ أدوات البرمجة بلغة (فيجول بيسك ستديو).</li> </ul>
<ul style="list-style-type: none"> <li>■ إدخال البيانات.</li> <li>■ إخراج المعلومات.</li> <li>■ أوامر الإسناد.</li> <li>■ الجمل الشرطية.</li> <li>■ حلقات التكرار.</li> </ul>	<ul style="list-style-type: none"> <li>■ الأوامر الأساسية.</li> </ul>





## تمارين



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



٢ ماذا نسمي أماكن تخزين البيانات في الذاكرة الرئيسة ؟



٣ ماذا يعني الأمر التالي : Dim Number As Integer ؟



٤ ما الفرق بين الثوابت والمتغيرات ؟



٥ هل الأسماء التالية يمكن استخدامها لتسمية المتغيرات :



2ABC, 123, AB2, AB\_2, Num one, While, aBxY, Case

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



ما نواتج العمليات الحسابية التالية :

$$X+Z*A^2$$



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



$$X*5^A$$



٧ حوّل العمليات الجبرية التالية إلى صيغة برمجية :



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



$$z x + 4 + y$$



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



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



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



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



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. ( )
- ج) نتائج العمليات المنطقية هي دائماً أرقام. ( )
- د) ننفذ عمليات الضرب والقسمة قبل عمليات الجمع والطرح. ( )
- هـ) إذا أردنا أن يدخل المستخدم رقم هاتفه نستخدم أداة مربع الاختيار. ( )

اذكر ثلاث طرق لإخراج معلومات للمستخدم.

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

ما الذي يحدث بعد تنفيذ الإجراء التالي :

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

```
Num=0
```

```
If Num<1 Then Name=InputBox(«أهلا بك الرجاء إدخال اسمك»)
```

```
MsgBox (Name+«أهلا بك يا»)
```

```
ENDIF
```

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
```

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



## اختبار

اختر رمز الإجابة الصحيحة فيما يلي :

١. لكتابة برنامج هناك :

- أ - ثلاث مراحل.
- ب - مرحلتان.
- ج - أربع مراحل.
- د - خمس مراحل.

٢. تحديد عدد الواجهات والأدوات المستخدمة لكل واجهة نقصد به :

- أ - تصميم الواجهات.
- ب - برمجة الواجهات.
- ج - تعديل الواجهات.
- د - ربط الواجهات.

٣. قبل كتابة الأوامر البرمجية نحتاج إلى :

- أ - تصميم الواجهات فقط.
- ب - تصميم الواجهات وضبط الخصائص أولاً.
- ج - ضبط الخصائص فقط.
- د - كتابة خوارزم البرنامج.

٤. تصنف البيانات إلى :

- أ - نوع واحد.
- ب - نوعين.
- ج - أربعة أنواع.
- د - ثلاثة أنواع.

٥. إعطاء اسم لقيمة معينة واستخدامها داخل البرنامج هو تعريف :

- أ - الثابت.
- ب - المتغير.
- ج - التاريخ.
- د - الحروف.

٦ الجملة الصحيحة لتعريف متغير فيما يلي هي :

- أ - Dim x = int .  
 ب - Dim 2DF As long  
 ج - Dim x = If .  
 د - Dim x As string

٧ ناتج العملية الحسابية  $M = 2 \times 6 + 3^2$  هو :

- أ - 13  
 ب - 20  
 ج - 12  
 د - 21

٨ العملية التي نتيجتها True فيما يلي هي :

- أ -  $6 \times 4 = 5 \times 3 + 4$   
 ب -  $6 \times 4 < 5 \times 3 + 4$   
 ج -  $6 \times 4 > 5 \times 3 + 4$   
 د -  $6 + 4 > 5 \times 3 + 4$

٩ من أدوات إخراج المعلومات :

- أ - RadioButton  
 ب - ListBox  
 ج - ChekBox  
 د - TextBox

١٠ لتنفيذ أمر معين طالما كان الشرط صحيحاً فإننا نستخدم :

- أ - If .. Then .. ElseIf  
 ب - For.. Next  
 ج - Do .. While  
 د - Select Case







Microsoft®  
Visual Studio®2010

## تدريبات الوحدة السادسة

البرمجة بلغة (فيجول بيسك ستوديو)



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

في هذا التدريب ستتعلم:

تشغيل برنامج (فيجول بيسك ستوديو ٢٠١٠) والبدء في مشروع جديد.

مكونات برنامج (فيجول بيسك ستوديو ٢٠١٠).

مربع الأدوات.

## متطلبات التدريب

برنامج (فيجول بيسك ستوديو ٢٠١٠).

## مقدمة التدريب

تعد البرمجة بلغة (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، وستتعرف في التدريبات القادمة على البرمجة باستخدام (فيجول بيسك ستوديو)، لتكتمل لديك المهارات اللازمة في إنشاء برامج متكاملة.

## خطوات التدريب

### تشغيل برنامج (فيجول بيسك ستوديو)؛

أولاً

لتشغيل برنامج (فيجول بيسك ستوديو) :

١ انقر على زر (ابدأ).

٢ افتح قائمة (كافة البرامج).

٣ اختر المجلد (Microsoft Visual Studio 2010)

حتى تسدل القائمة الموجودة بداخله.

٤ انقر على برنامج (Microsoft Visual Studio 2010)،

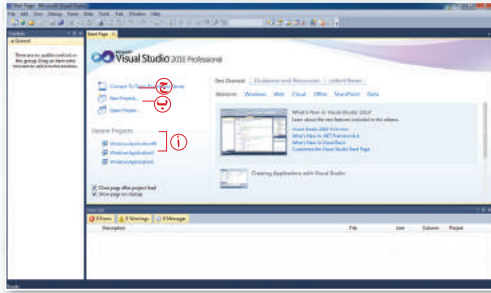
كما في الشكل (١-١-٦).

٥ سيشتغل برنامج (فيجول بيسك ستوديو ٢٠١٠)، وتظهر نافذة

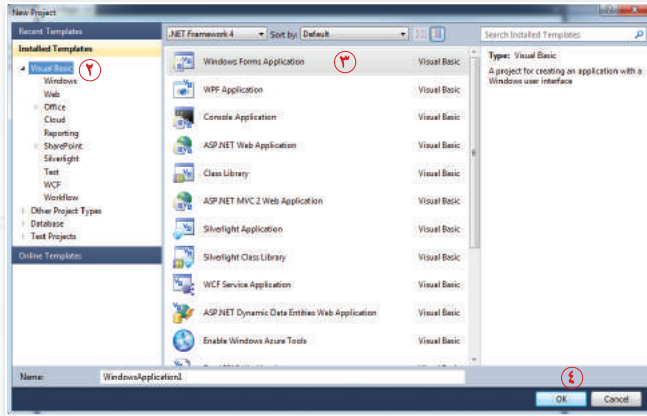
(StartPage) كما في الشكل (٢-١-٦)، التي من خلالها نستطيع عمل التالي:



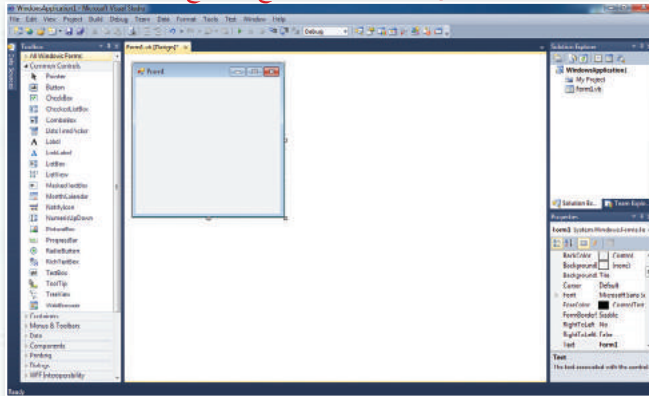
شكل ١-١-٦: تشغيل برنامج (فيجول بيسك ستوديو ٢٠١٠)



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



شكل ٦-١-٣: نافذة أنواع المشاريع



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

## البدء في مشروع جديد:

## ثانياً

ولإنشاء مشروع جديد اتبع الخطوات التالية:

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 ستوديو) بإطلاق اسم مشروع على مجموعة الملفات التي تمثل البرنامج.

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 ستوديو).

## مكونات برنامج (فيجول بيسك ستوديو):

## ثالثاً

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

### ١- شريط القوائم:

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

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



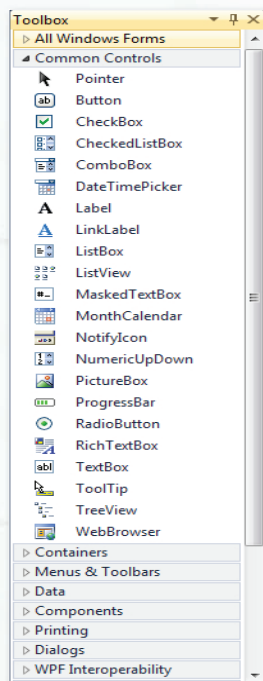
## ٢ شريط الأدوات القياسي:

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



## ٣ مربع الأدوات (Toolbox):

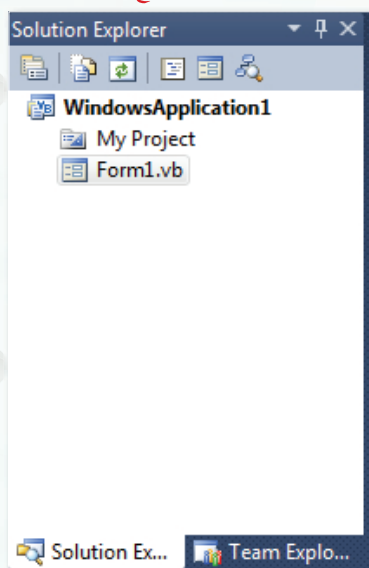
هي الأدوات التي نستخدمها في بناء واجهة البرامج كما في الشكل (٥-١-٦)، ولإظهارها اختر الأمر (Toolbox) من قائمة (View).



شكل ٥-١-٦: مربع الأدوات

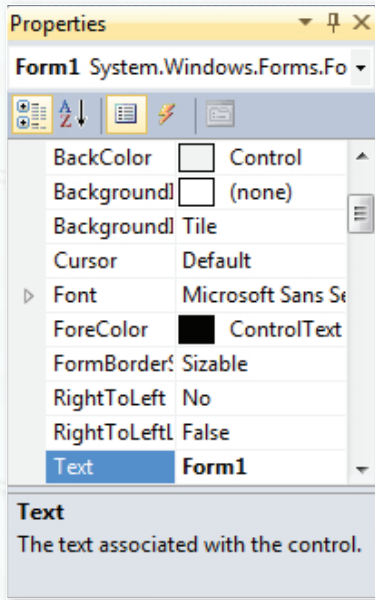
## ٤ محتويات المشروع:

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



شكل ٦-١-٦: محتويات المشروع

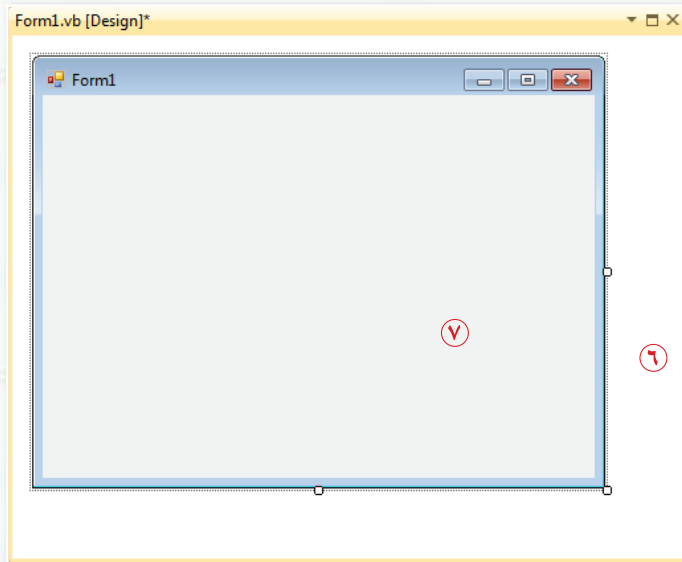




شكل ٦-١-٧: إطار الخصائص

### ٥ إطار الخصائص (Properties):

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



شكل ٦-١-٨: إطار المشروع ونافذة النموذج

### ٦ إطار المشروع:

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

### ٧ 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.

اسم الأداة	رمز الأداة	وظيفة الأداة
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.



وضع الأدوات على النموذج:

خامساً

إثراء علمي

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

توضع 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- ولتعديل 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 (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

إغلاق برنامج (فيجول بيسك ستوديو):

سادساً

يُغلق برنامج (فيجول بيسك ستوديو) بإحدى الطريقتين التاليتين:

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

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

## تمرينات



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



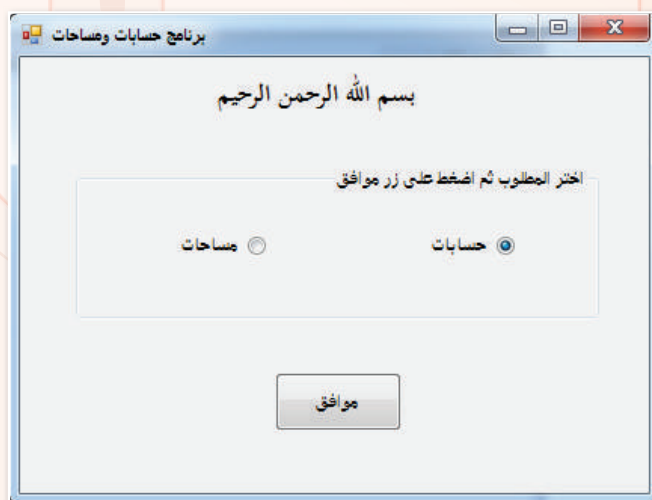
ما الفرق بين ( ListBox ) و ( ComboBox ) ؟



كيف يتم إظهار مربع الأدوات ( Toolbox ) ؟



ما الأدوات التي نحتاجها لتصميم نافذة النموذج التالي ؟





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

في هذا التدريب ستتعلم:

إنشاء مشروع باستخدام (فيجول بيسك ستوديو ٢٠١٠).

تصميم النماذج.

حفظ النماذج.



## متطلبات التدريب

برنامج (فيجول بيسك ستوديو ٢٠١٠).

## مقدمة التدريب

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

- ١ - صياغة حل المسألة وتصميم النماذج على الورق، ويتم الصياغة حسبما تعلمته في الجزء النظري، أما التصميم على الورق فمعناه رسم النموذج على الورق تمهيداً لرسمه على الجهاز باستخدام (فيجول بيسك ستوديو).
- ٢ - تصميم واجهة المستخدم باستخدام (فيجول بيسك ستوديو).
- 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): تحسب مساحة المستطيل وذلك بمعلومية الطول والعرض.

## خطوات التدريب

### التخطيط للنموذج الأول (مقدمة) :

أولاً

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).



شكل ١-٢-٦: الواجهة الرئيسية للبرنامج



## أدوات وخصائص النموذج الأول :

ثانياً

١ عدّل خصائص نافذة النموذج الأول لتصبح كالتالي:

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:

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

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

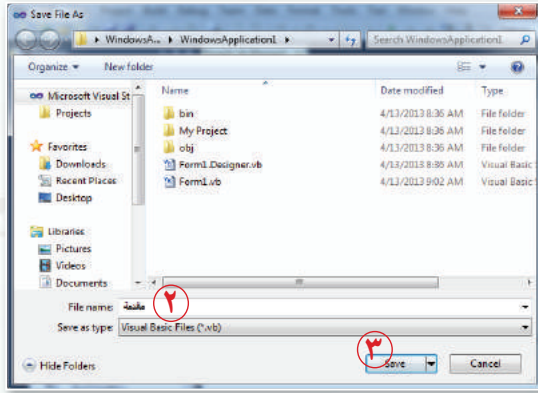
القيمة	الغرض من الخاصية	اسم الخاصية	اسم الأداة
GroupBox1	تحديد اسم الأداة	Name	GroupBox
اختر المطلوب ثم اضغط على زر (موافق)	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	

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

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 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	موافق



شكل ٢-٦: حفظ النموذج

### حفظ النموذج الأول :

ثالثاً

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

٢ تسمية النموذج (مقدمة) كما في الشكل (٢-٦-٢).

٣ انقر على زر (Save).

### التخطيط النموذج الثاني (حسابات) :

رابعاً

١ تحليل عناصر المسألة:

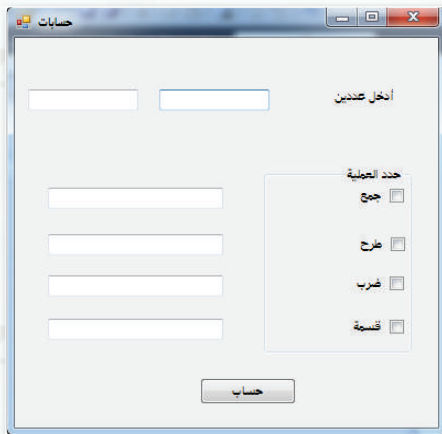
مخرجات البرنامج: ناتج العملية الحسابية.

مدخلات البرنامج: عدنان 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 (٢-٦-٣).

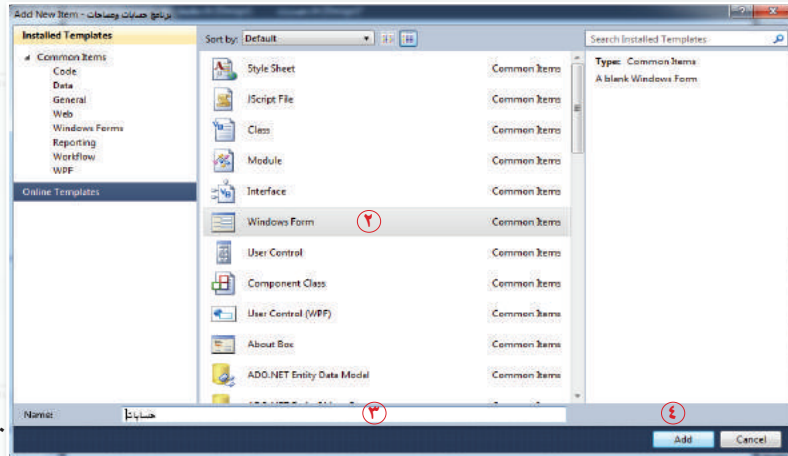


شكل ٢-٦-٣: تصميم واجهة (حسابات)



#### خامساً

#### البدء في تصميم النموذج الثاني (برنامج حسابات) :



شكل ٤-٢-٦ : إنشاء نموذج جديد

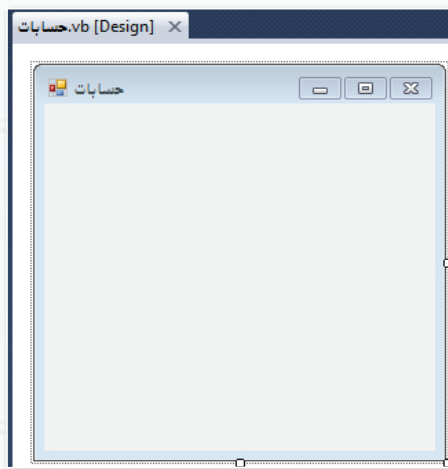
لإنشاء شاشة (حسابات) نحتاج إلى إنشاء نموذج جديد، وخطوات الإنشاء كالتالي:

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 ٥-٢-٦ : نموذج حسابات

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

#### 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

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	أدخل عددين

أنشئ أداتي مربع النص ( TextBox abt ) وغيّر 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

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

#### إثراء علمي

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:

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

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

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

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

٦ 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	حساب

٧ أنشئ أربع أدوات مربع النص (  TextBox ) وغيّر 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



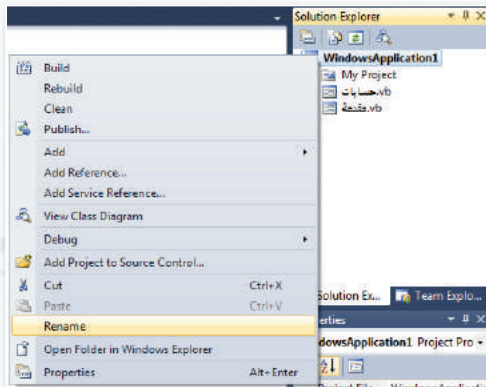
اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
TextBox	Name	تحديد اسم الأداة	TextBox4
	Text	إظهار العنوان داخل الأداة على الشاشة	(فارغ)
	RightToLeft	اتجاه النص لليمين	Yes
اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
TextBox	Name	تحديد اسم الأداة	TextBox5
	Text	إظهار العنوان داخل الأداة على الشاشة	(فارغ)
	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

#### إثراء علمي

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

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

View	Project	Build	Debug	Team	Data	Tools
	Solution Explorer	Ctrl+Alt+L				
	Team Explorer	Ctrl+\\, Ctrl+M				
	Server Explorer	Ctrl+Alt+S				



شكل ٦-٢: إعادة تسمية المشروع

#### حفظ النموذج الثاني (حسابات) :

سابعاً

اختر الأمر (Save) من قائمة (File).

#### إعادة تسمية المشروع :

ثامناً

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

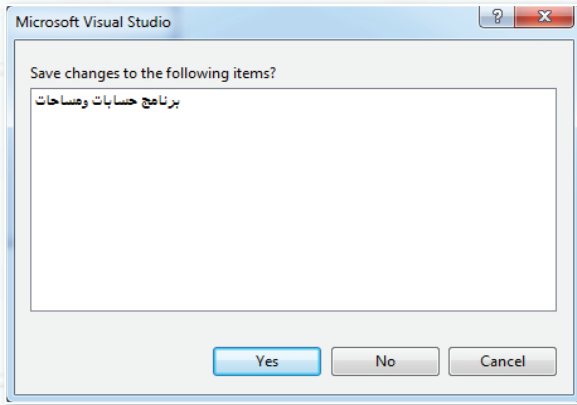
٢ - اختر الأمر (Rename).

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



حفظ المشروع وإغلاقه :

تأسعاً



شكل ٧-٢-٦: حفظ المشروع

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

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

٣ انقر على زر (Yes) لحفظ المشروع.

إثراء علمي

سيُحفظ المشروع باسم

(WindowsApplication1)

## تمريبات

ما الغرض من الخواص التالية :

- Font
- Text
- Name
- RightToLeft

ما الفرق بين الأداة (CheckBox) والأداة (RadioButton) ؟

- 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.

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



## .....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 ستوديو 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.

## خطوات التدريب

### أولاً: التخطيط للنموذج الثالث (مساحات) :

تحليل عناصر المسألة:

- مخرجات البرنامج: مساحة المستطيل :  
مدخلات البرنامج: عدنان مدخلان ولنرمز لهما (أ) (الطول)، ب (العرض) .  
عمليات المعالجة: قانون إيجاد مساحة المستطيل = الطول × العرض.
- التصميم على ورق خارجي:  
يُصمم هذا النموذج كما في الشكل (١-٣-٦) .

شكل ١-٣-٦ : تصميم واجهة (مساحات)

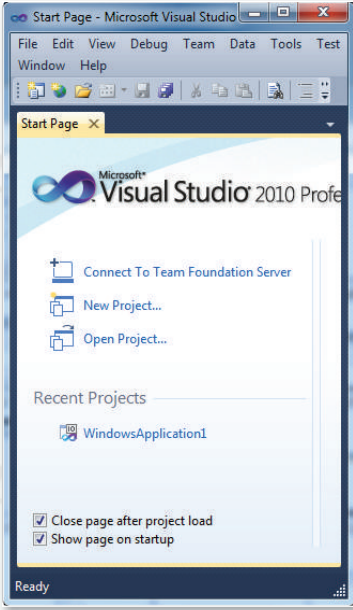


ثانيًا

افتح ملف مشروع (حسابات ومساحات) :

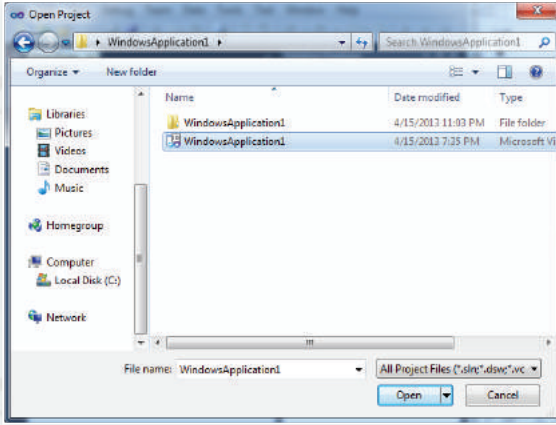
يُفتح المشروع باستخدام إحدى الطرق التالية:

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



شكل ٢-٣-٦: الشاشة الرئيسية

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



شكل ٣-٣-٦: نافذة اختيار ملف مشروع

ثالثًا

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	الغرض 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:

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

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

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 زر الأمر ( Button ab ) وغيّر 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 abl ) وغيّر 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


خامساً حفظ النموذج الثالث (مساحات) :

اختر الأمر (Save) من قائمة (File).

سادساً حفظ المشروع وإغلاقه :

احفظ المشروع وأغلقه.

إثراء علمي

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



## تمرينات



ما الفرق بين المشروع والنموذج والخاصية؟



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.

ما خطوات إنشاء نموذج جديد في (فيجول بيسك ستوديو)؟



ما الفرق بين أداة (Label) وأداة (Textbox)؟





## .....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

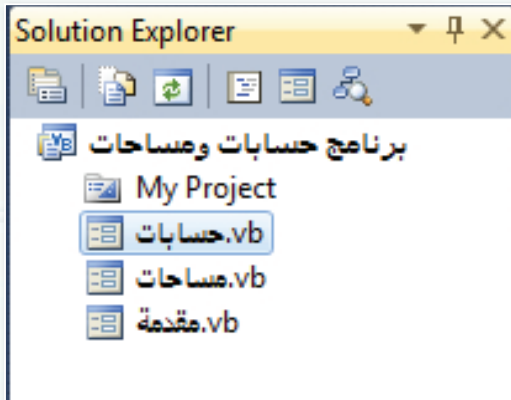
- برنامج (فيجول بيسك ستوديو ٢٠١٠).
- مشروع (windows application) الذي تم إنشاؤه في التدريب السابق.

## 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

- افتح مشروع (حسابات ومساحات) الذي أنشئ في التدريبات السابقة وحفظ باسم (مشروع).
- اختر النموذج الثاني (حسابات)، بالنقر المزدوج عليه من محتوى المشروع كما في الشكل (٦-٤-١).
- تظهر شاشة نموذج (حسابات)، التي تتكون من أدوات أنشئت في التدريبات السابقة.
- 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.
- قبل الانتقال لكتابة الأوامر لابد من معرفة المكان الذي سوف يحتوي هذه الأوامر وهي شاشة البرمجة، فكيف يتم الدخول إلى هذه الشاشة والتعامل معها؟

الدخول إلى شاشة البرمجة :

ثانياً

إثراء علمي

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 ستوديو): حيث يمكن تحديد الإجراء المناسب للأداة التي تم اختيارها في مربع سرد الأدوات.

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

Public Class : التعريف العام لطبقة النموذج.

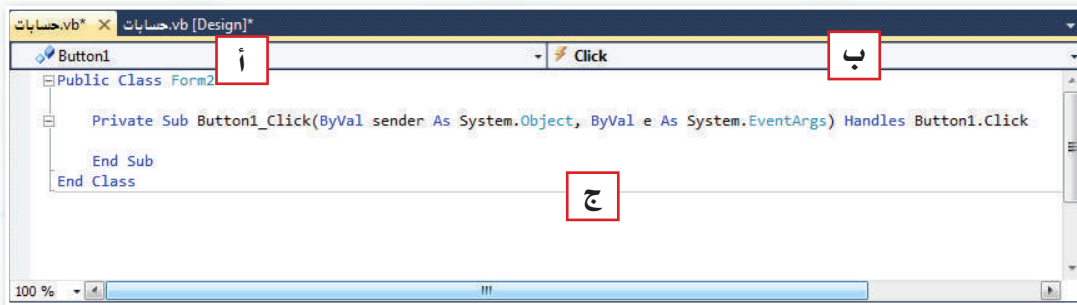
Private Sub : الإجراء الفرعي داخل البرنامج، وهو ثابت لجميع الإجراءات.

الإجراء \_\_ اسم الأداة : اسم الأداة المراد التعامل معها مثل : (Button1)، والإجراء الخاص بالأداة مثل : (Click)، والشرطة السفلية ( \_\_ ) وتربط 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



المعالجة التي تجري على العمليات لإظهار المخرجات.

الحصول على مخرجات البرنامج من خلال أدوات مربعات النصوص

(Text3 , Text4 , Text5 Text6 ) .

## Public Class Form2

Private Sub Button1\_\_Click() Handles Button1.Click

Dim A , B As Single

A = TextBox1.Text

B = TextBox2.Text

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

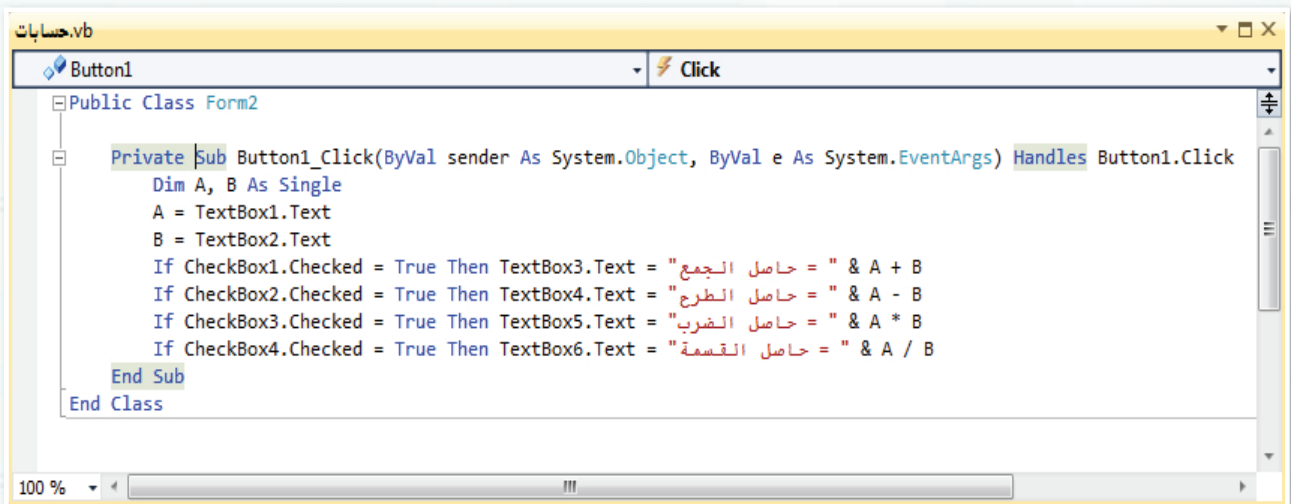
If CheckBox2.Checked = True Then TextBox4.Text = "حاصل الطرح = " & A - B

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.

### تنفيذ برنامج (حسابات) :

#### رابعاً

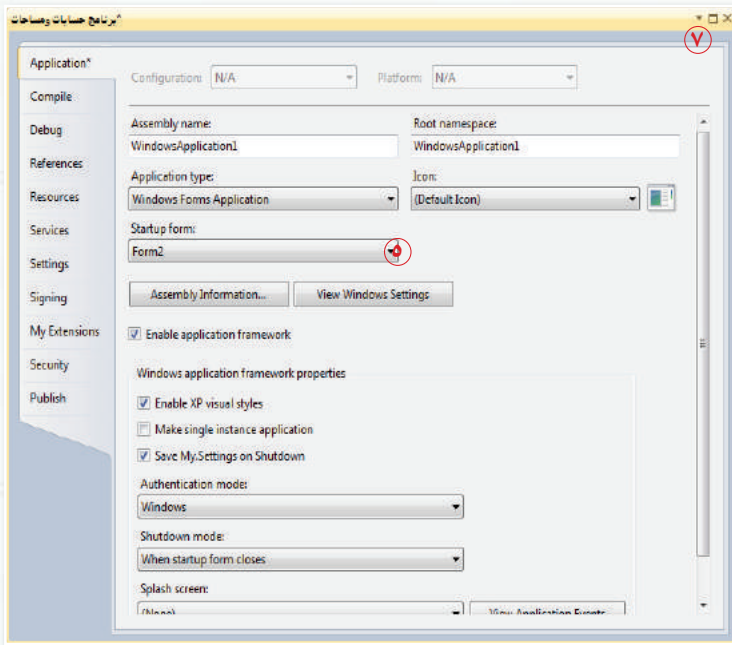
#### إثراء علمي

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 ستوديو) ينفذ أول شاشة تم إنشائها).

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

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



شكل ٦-٤-٤ : نافذة اختيار النموذج للتنفيذ

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

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).

#### إيقاف تنفيذ برنامج (حسابات) :

خامساً

انقر على زر التوقف ( ) في شريط الأدوات القياسي.

#### حفظ برنامج (حسابات) :

سادساً

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

#### إثراء علمي

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



## تمارين



ما المقصود بالأوامر التالية :



● End Sub    ● Dim grade As Integer    ● Private Sub Button5\_Click ( )

ما الفرق بين الأمرين التاليين :



```
M = TextBox1.text  
TextBox1.text = M
```

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
```

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





## .....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

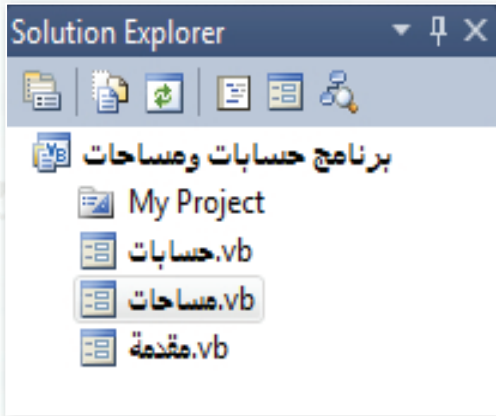
- ▶ برنامج (فيجول بيسك ستوديو ٢٠١٠).
- ▶ مشروع (windows application) الذي تم إنشاؤه في التدريب السابق.

## 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 خلوه من الأخطاء، بالإضافة إلى تحويل المشروع إلى ملف ذاتي التنفيذ بصيغة (.exe)، أي سيتم تنفيذ البرنامج مباشرة دون الحاجة إلى تشغيله بواسطة (فيجول بيسك ستوديو).

## خطوات التدريب

### أولاً : فتح النموذج الثالث (مساحات) :



شكل ٦-٥-١ : نافذة محتوى المشروع

- ١ ▶ افتح مشروع (حسابات ومساحات).
- ٢ ▶ اختر النموذج الثالث (مساحات)، بالنقر المزدوج عليه من محتوى المشروع كما في الشكل (٦-٥-١).
- ٣ ▶ تظهر شاشة نموذج (مساحات)، التي تتكون من أدوات أنشئت في التدريبات السابقة.
- ٤ ▶ انقر نقرًا 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.



### Public Class Form3

Private Sub Button1\_\_Click() Handles Button1.Click

Dim A, B As Single

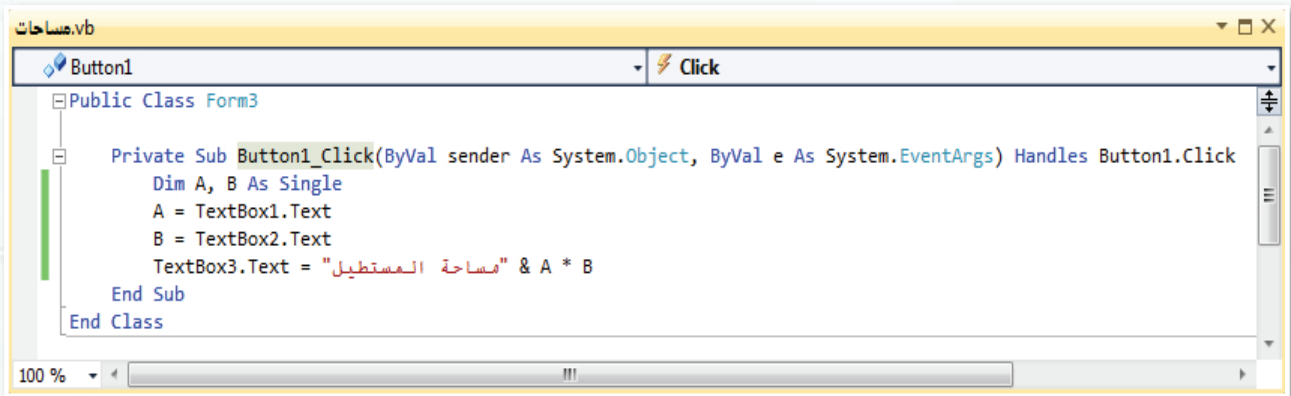
A = TextBox1.Text

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.


## ثانياً

### تنفيذ برنامج (مساحات) :

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

## ثالثاً

### إيقاف تنفيذ برنامج (مساحات) :

انقر على زر التوقف (  ) في شريط الأدوات القياسي.

## رابعاً

### حفظ برنامج (مساحات) :

احفظ برنامج (مساحات).

## خامساً

### 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 نقرًا 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
```

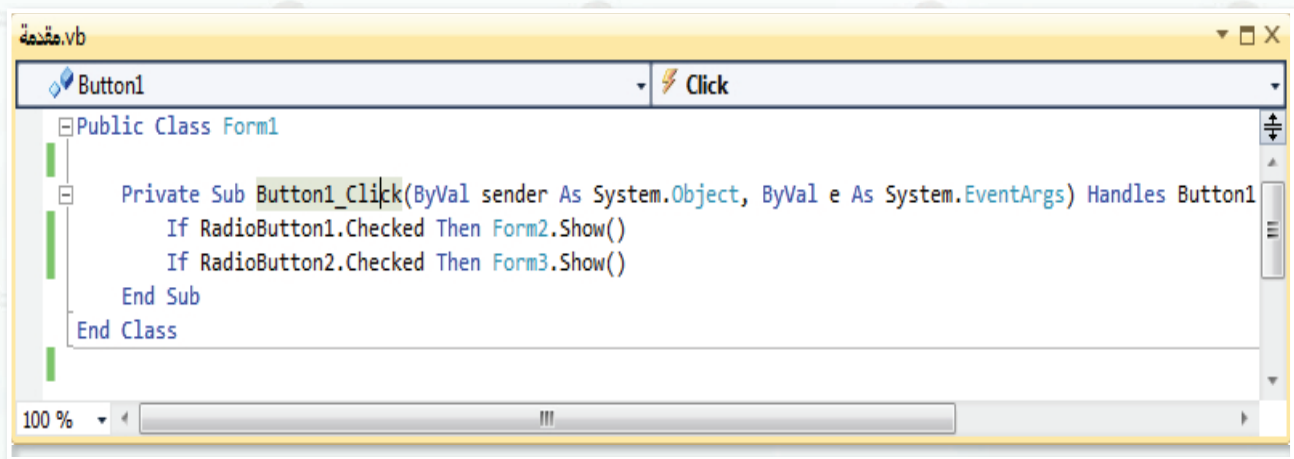
```
Private Sub Button1__Click() Handles Button1.Click
```

```
    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 الثالث (مساحات) المسمى (Form3).

**السطر الخامس:** نهاية الإجراء.

سادساً حفظ برنامج المقدمة (مقدمة) :

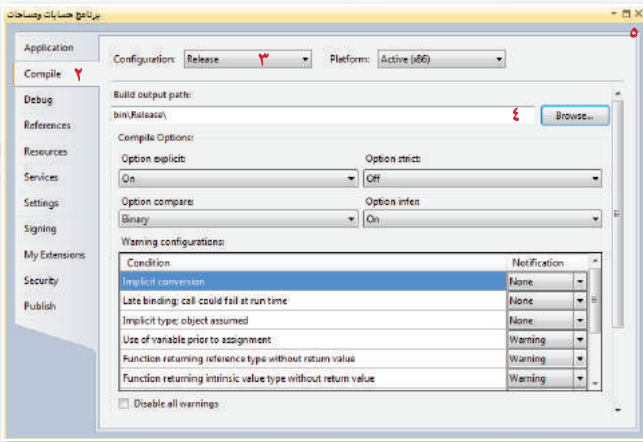
احفظ برنامج (مقدمة).

سابعاً تنفيذ مشروع (حسابات ومساحات) :

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

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

ثامناً تحويل مشروع (حسابات ومساحات) إلى برنامج ذاتي التنفيذ :



١ ▶ اختر الأمر (Properties) من قائمة

(Project)، فتظهر نافذة خصائص المشروع

كما في الشكل (٤-٥-٦).

٢ ▶ اختر التبويب (Compile).

٣ ▶ من القائمة المنسدلة (Configuration)

اختر (Release)، وهذا يعني أن النسخة

التنفيذية للبرنامج ستكون نسخة إصدار.

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

٤ ▶ حدّد مكان تخزين ملف التشغيل بالنقر على زر (Browse...)

الخاص بالخيار (Build output path)، واختر 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.

إثراء علمي

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



- 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 خُزن فيه ملف تشغيل برنامج (حسابات ومساحات) عن طريق مستكشف الملفات والمجلدات في نظام التشغيل.
- ١٠ انقر نقرًا مزدوجاً على الملف (WindowsApplication1.exe) ليتم تشغيل برنامج (حسابات ومساحات).
- ١١ أغلق برنامج (حسابات ومساحات).
- وبهذا تكون قد انتهيت من مشروع برنامج (حسابات ومساحات).

## تمرينات



- 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 تتخذها لكي يقوم (فيجول بيسك) بذلك؟
- ٢ اكتب أوامر برنامج بلغة (فيجول بيسك) لإدخال وقت بصيغة (٢٤) ساعة وتحويلها إلى صيغة (١٢) ساعة وإحاقها بعباراة "صباحاً" أو "مساءً".
- ٣ ما الخطوات اللازمة لتحويل مشروع (فيجول بيسك) إلى برنامج ذاتي التشغيل؟



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

في هذا التدريب ستتعلم:

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



## متطلبات التدريب

برنامج (فيجول بيسك ستوديو 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 ننتقل بصورة تدريجية إلى تطبيقات أكثر عمقاً وتتطلب فهماً أكثر.

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

$$\text{درجة الحرارة بالفهرنهايتي} = (9 \div 5 \times \text{درجة الحرارة بالمتوي}) + 32$$

$$\text{درجة الحرارة بالمتوي} = (9 \div 5 \times (\text{درجة الحرارة بالفهرنهايتي} - 32))$$

## خطوات التدريب

التصميم الورقي :

أولاً

صيغة حل المسألة :

١

حلّ عناصر المسألة، وذلك بتحديد التالي:

● **مخرجات البرنامج:** درجة الحرارة بالفهرنهايت أو بالمتوي حسب اختيار المستخدم.

● **مدخلات البرنامج:** درجة الحرارة بالمتوي، أو بالفهرنهايت ولنرمز لها بـ (D).

● **عمليات المعالجة:** قانون التحويل:

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

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



كتابة الخطوات الخوارزمية للمسألة :

٢

اكتب الخطوات الخوارزمية كما يلي:

أ) اختر عملية التحويل بالمتوي أو بالفهرنهايت.

ب) أدخل درجة الحرارة (D).

ج) احسب عملية التحويل حسب اختيار المستخدم:

• اجعل درجة الحرارة بالفهرنهايت  $= 32 + (D \times 5 \div 9)$

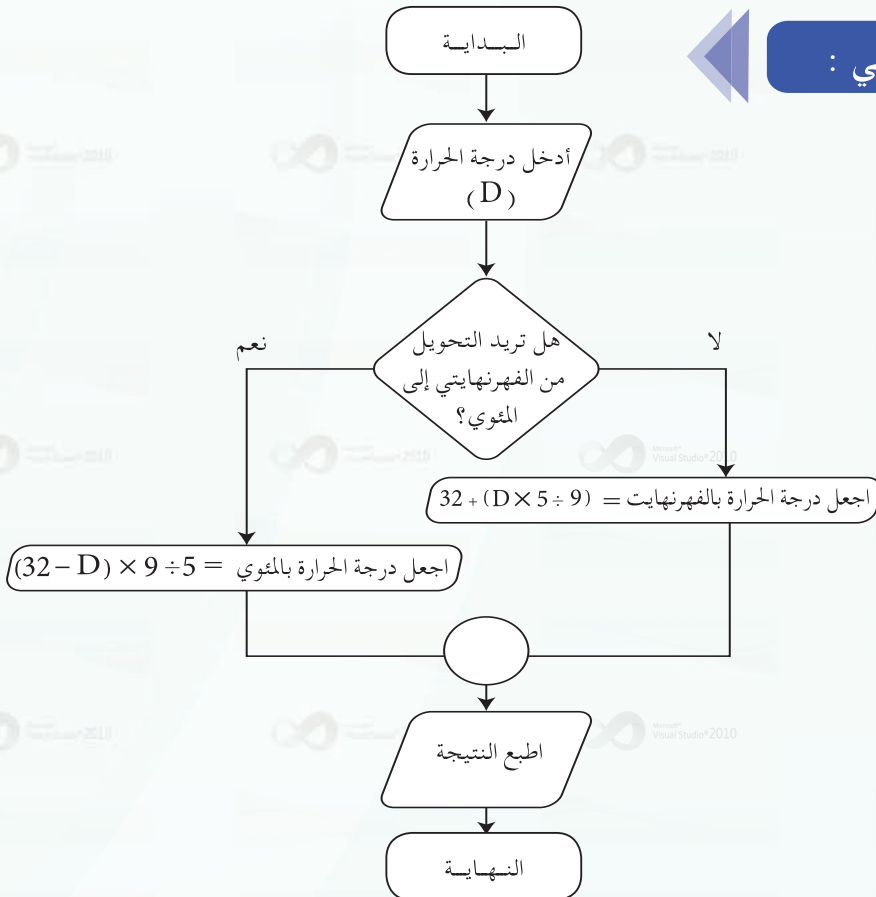
• أو اجعل درجة الحرارة بالمتوي  $= (32 - D) \times 9 \div 5$

د) اطبع النتيجة.

هـ) النهاية.

المخطط الإنسيابي :

٣





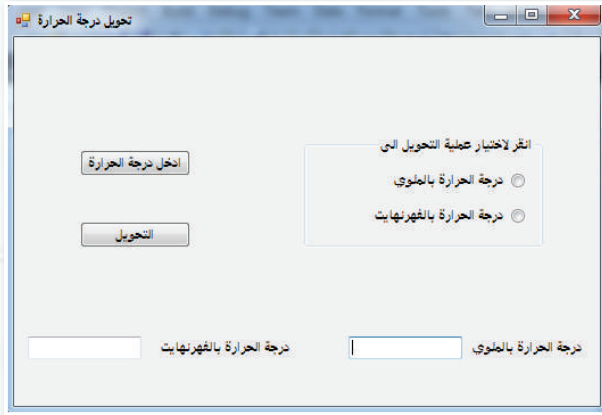
أدوات وخصائص النموذج :

ثانياً

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

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

القيمة	الغرض 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 (١-٦-٦)، واضبط خصائص

كل أداة كما في الجدول التالي:

شكل ١-٦-٦ : نموذج تحويل درجة الحرارة

القيمة	الغرض من الخاصية	اسم الخاصية	اسم الأداة
Label1	تحديد اسم الأداة	Name	Label
درجة الحرارة بالمئوي	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	
Label2	تحديد اسم الأداة	Name	Label
درجة الحرارة بالفهرنهايت	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	

اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
TextBox	Name	تحديد اسم الأداة	Textbox2
	Text	إظهار العنوان داخل الأداة على الشاشة	( فارغ ) احذف الموجود
	RightToLeft	اتجاه النص لليمين	Yes
GroupBox	Name	تحديد اسم الأداة	GroupBox1
	Text	إظهار العنوان داخل الأداة على الشاشة	انقر لاختيار عملية التحويل إلى
	RightToLeft	اتجاه النص لليمين	Yes
RadioButton	Name	تحديد اسم الأداة	RadioButton1
	Text	إظهار العنوان داخل الأداة على الشاشة	درجة الحرارة بالمئوي
	RightToLeft	اتجاه النص لليمين	Yes
RadioButton	Name	تحديد اسم الأداة	RadioButton2
	Text	إظهار العنوان داخل الأداة على الشاشة	درجة الحرارة بالفهرنهايت
	RightToLeft	اتجاه النص لليمين	Yes
Button	Name	تحديد اسم الأداة	Button1
	Text	إظهار العنوان داخل الأداة على الشاشة	أدخل درجة الحرارة
	Name	تحديد اسم الأداة	Button2
Button	Text	إظهار العنوان داخل الأداة على الشاشة	التحويل

### كتابة أوامر البرنامج :

### ثالثا

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

حفظ النموذج :

رابعاً

يُحفظ النموذج بإتباع الخطوات التالية:

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

٢ كتابة اسم البرنامج (تحويل درجة الحرارة).

تنفيذ النموذج :

خامساً

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

تمارين



١٥ اكتب برنامجاً باستخدام (فيجول بيسك ستوديو) لإدخال وقت بصيغة (٢٤) ساعة وتحويلها إلى صيغة

(١٢) ساعة وإحاقها بعبارـة "صباحاً" أو "مساءً"، وذلك عن طريق:

تصميم نموذج واجهة البرنامج.

تحديد الخصائص لكل أداة وضعتها على النموذج.

كتابة أوامر البرنامج.

تنفيذ البرنامج.





## .....التدريب السابع : تطبيق حساب الزكاة

في هذا التدريب ستتعلم:

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

## متطلبات التدريب

برنامج (فيجول بيسك ستوديو ٢٠١٠).

## مقدمة التدريب

في هذا التدريب ستقوم بكتابة برنامج باستخدام (فيجول بيسك ستوديو 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 ١٠٠٠ ريال)، علماً أن الزكاة =  $0,025 \times \text{المبلغ}$ .

## خطوات التدريب

التصميم الورقي :

أولاً

## صياغة حل المسألة :

١

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) =  $M \times 0.025$ .



## كتابة الخطوات الخوارزمية للمسألة :



اكتب الخطوات الخوارزمية كما يلي:

(أ) إدخال المبلغ  $M$ .

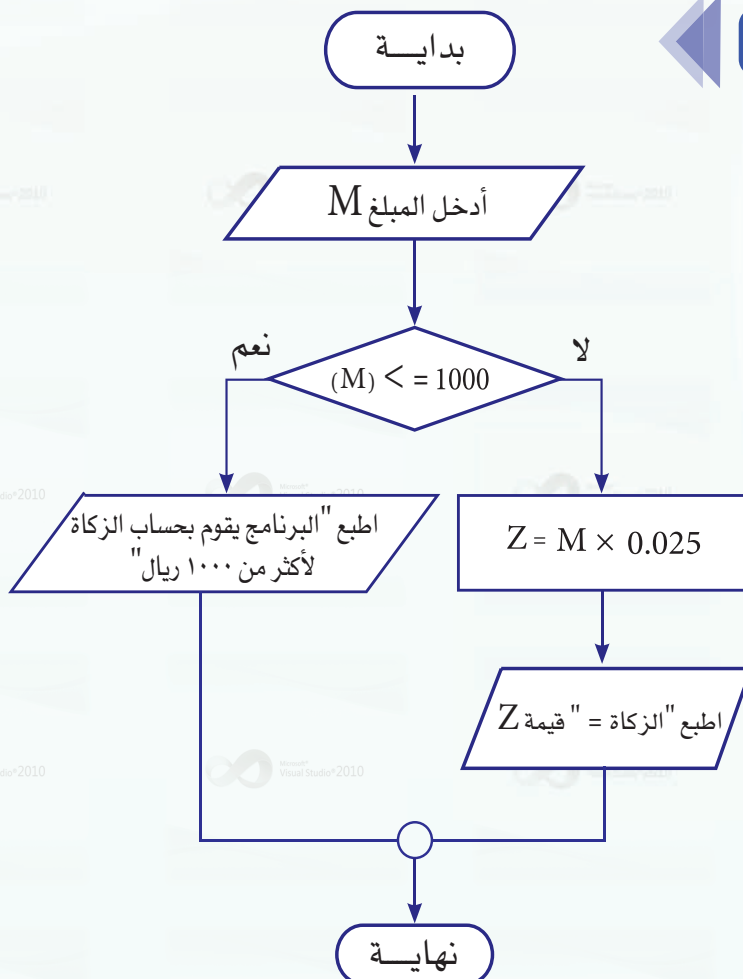
(ب) هل المبلغ  $M \geq 1000$  ، اطبع عبارة " البرنامج يقوم بحساب الزكاة لأكثر من ١٠٠٠ ريال " ، وتوقف.

(ج) احسب  $Z = M \times 0.025$ .

(د) اطبع  $Z$ .

(هـ) النهاية.

## المخطط الإنسيابي :



تصميم نافذة النموذج :

ثانياً

١ افتح برنامج (فيجول بيسك ستوديو) 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	



اسم الأداة	اسم الخاصية	الغرض من الخاصية	القيمة
TextBox	Name	تحديد اسم الأداة	Textbox2
	Text	إظهار العنوان داخل الأداة على الشاشة	( فارغ ) احذف الموجود
	RightToLeft	اتجاه النص لليمين	Yes
Button	Name	تحديد اسم الأداة	Button1
	Text	إظهار العنوان داخل الأداة على الشاشة	الزكاة

## كتابة أوامر البرنامج :

ثالثاً

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

## Public Class Form1

Private Sub Button1\_\_Click()

Dim M As Single. Z As Single

M = Textbox1.Text

If M &lt;= 1000 Then

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

Else

Z = 0.025 \* M

Textbox2.Text = " = الزكاة " &amp; Z

End If

End Sub

End Class



### 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.

### حفظ النموذج :

### رابعاً

يُحفظ النموذج بإتباع الخطوات التالية:

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

### تنفيذ النموذج :

### خامساً

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

## تمارين



١. اكتب برنامجاً باستخدام (فيجول بيسك ستوديو) يقوم لإدخال (10) أعداد ثم طباعة العدد الأكبر منها، وذلك عن طريق :

- 1 تصميم نموذج واجهة البرنامج.
- 2 تحديد الخصائص لكل أداة وضعتها على النموذج.
- 3 كتابة أوامر البرنامج.
- 4 تنفيذ البرنامج.



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

في هذا التدريب ستتعلم:

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

## متطلبات التدريب

برنامج (فيجول بيسك ستوديو 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	80-89	70-79	50-69	أقل من 50
التقدير	ممتاز	جيد جداً	جيد	مقبول	غير مجتاز

## خطوات التدريب

التصميم الورقي :

أولاً

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.



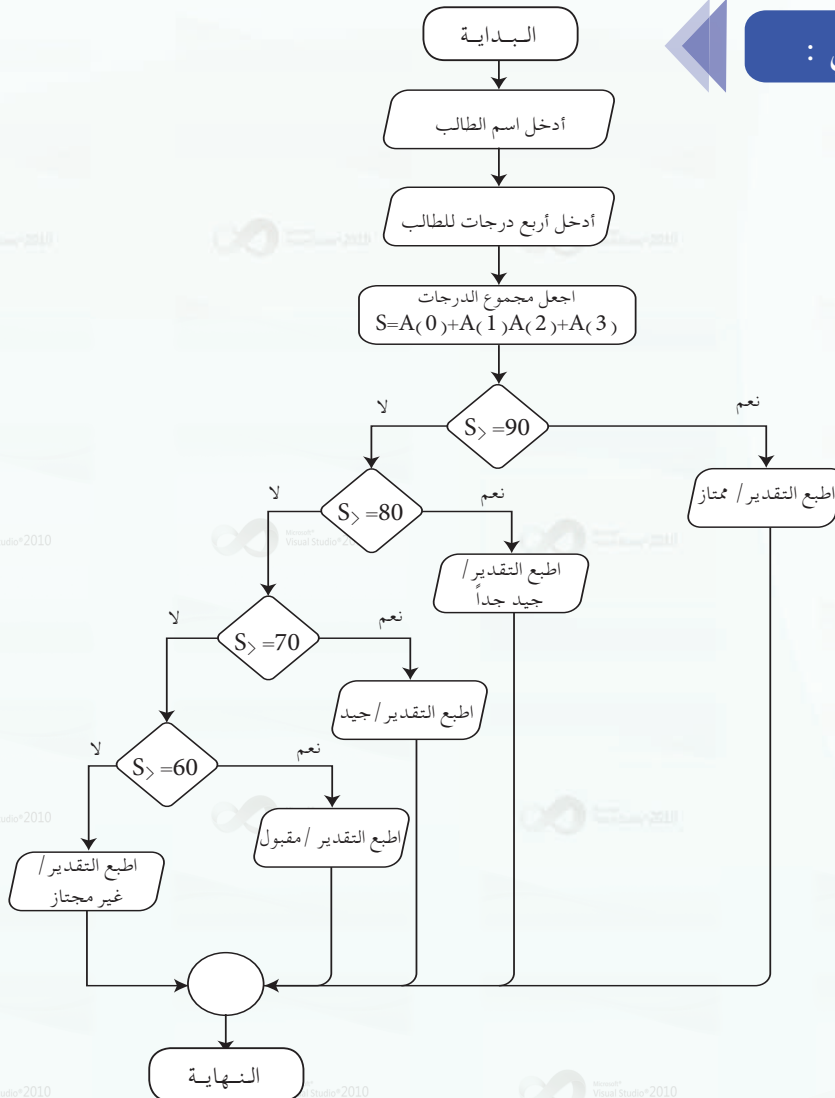
## كتابة الخطوات الخوارزمية للمسألة :



اكتب الخطوات الخوارزمية كما يلي:

- أدخل اسم الطالب.
- أدخل درجات الطالب الأربع على التوالي.
- خزن الدرجات في المصفوفة.
- جمع الدرجات الأربع:  $S = A(0) + A(1) + A(2) + A(3)$
- قارن بين مجموع الدرجات ( $S$ ) والتقدير المناسب.
- اطبع النتيجة مع بيانات الطالب.
- النهاية.

## المخطط الإنسيابي :



تصميم نافذة النموذج :

ثانياً

- ١ افتح برنامج (فيجول بيسك ستوديو) وأنشئ مشروعاً جديداً.
- ٢ اضبط خصائص نموذج برنامج حساب التقدير وفق الجدول التالي:

القيمة	الغرض 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 (٦-٨-١)، واضبط خصائص كل أداة كما في الجدول التالي:

شكل ٦-٨-١ : نموذج برنامج حساب التقدير

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





القيمة	الغرض من الخاصية	اسم الخاصية	اسم الأداة
Label2	تحديد اسم الأداة	Name	Label
احذف الموجود (فارغ)	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	
Label3	تحديد اسم الأداة	Name	Label
درجات الطالب	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	
Label4	تحديد اسم الأداة	Name	Label
احذف الموجود (فارغ)	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	
Label5	تحديد اسم الأداة	Name	Label
احذف الموجود (فارغ)	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	
Label6	تحديد اسم الأداة	Name	Label
احذف الموجود (فارغ)	إظهار العنوان داخل الأداة على الشاشة	Text	
Yes	اتجاه النص لليمين	RightToLeft	

## كتابة أوامر البرنامج :

ثالثاً

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 على التوالي ( أعمال السنة ، الحضور والغياب ، الاختبار الفصلي ، الاختبار النهائي ) "

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

End Class

### شرح البرنامج :

السطر الثاني إلى الثالث: تعريف المتغيرات العامة التي تستخدم في أكثر من إجراء. وهي:

العداد (I) المستخدم مع المصفوفة، المصفوفة ( A ) وعدد عناصرها (٤-١) ، 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 العداد (تكرار) FOR بعدد مساوٍ 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.

### حفظ النموذج :

رابعاً

يُحفظ النموذج بإتباع الخطوات التالية:

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

### تنفيذ النموذج :

خامساً

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

## تمارين



اكتب برنامجاً باستخدام (فيجول بيسك ستوديو) 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	80-89	70-79	50-69	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 ستوديو ٢٠١٠) لعرض أسعار أجهزة الحاسب.



## متطلبات التدريب

برنامج (فيجول بيسك ستوديو ٢٠١٠).

## مقدمة التدريب

في هذا التدريب ستقوم بكتابة برنامج باستخدام (فيجول بيسك ستوديو) 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
	EPSON	1631
طابعة متعددة المهام	HP	1871
	LG	2850
	DELL	4050
جهاز عرض	LG	7687
	HP	8625

## خطوات التدريب

التصميم الورقي :

أولاً

صياغة حل المسألة :

١

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.



كتابة الخطوات الخوارزمية للمسألة :

اكتب الخطوات الخوارزمية كما يلي:

أ ( اعرض أسماء أجهزة الحاسب وأنواعها.

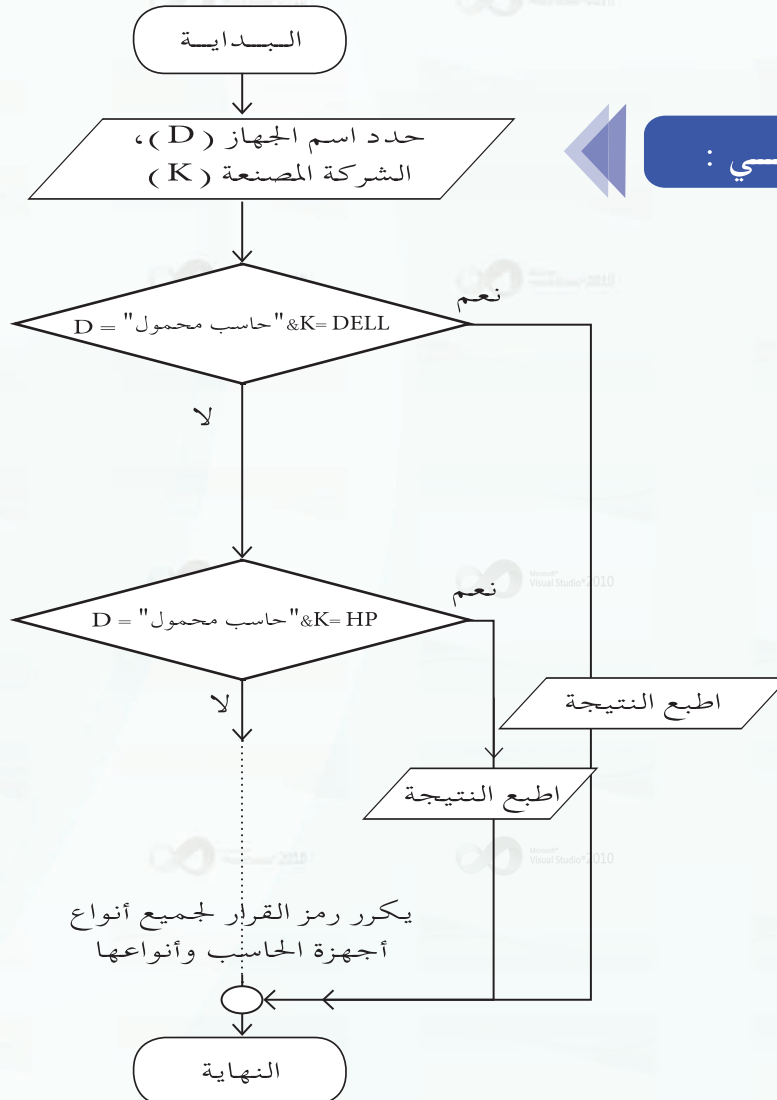
ب ( حدد اسم الجهاز (D) ، واسم الشركة المصنعة للجهاز (K).

ت ( قارن بين اسم الجهاز (D) ، واسم الشركة المصنعة للجهاز (K) اللذين تم اختيارهما لتحديد السعر.

ث ( اطبع النتيجة.

ج ( النهاية.

المخطط الإنسيابي :





ثانياً

تصميم نافذة النموذج :

- ١ افتح برنامج ( فيجول بيسك ستوديو ) وأنشئ مشروعاً جديداً.
- ٢ اضبط خصائص نموذج برنامج أجهزة الحاسب وفق الجدول التالي:

القيمة	الغرض 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	الغرض 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	

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

### كتابة أوامر البرنامج :

### ثالثاً

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
End Class

```

### 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.

## حفظ النموذج :

## رابعاً

يُحفظ النموذج باتباع الخطوات التالية:

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

٢ كتابة اسم النموذج (درجات وتقدير الطالب).

## تنفيذ النموذج :

## خامساً

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 ستوديو) 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





## مصطلحات الكتاب



### مصطلحات الوحدة الرابعة :

المصطلح باللغة العربية	المصطلح باللغة الإنجليزية
الإدخال	Input
الإخراج	output
وحدة المعالجة المركزية	CPU
لغة الآلة	Machine Language
لغة التجميع	Assembly Language
لغة البرمجة الإجرائية	Procedural Language
لغة البرمجة بالكائنات	Object Oriented Language
لغة البيسك	Basic Language
لغة الفيچول بييسك	Visual Basic Language
لغة السي	C Language
لغة السي بلس بلس	C++ Language
لغة الجافا	Java Language

## مصطلحات الوحدة السادسة :

المصطلح باللغة الإنجليزية	الترجمة باللغة العربية
High level language	لغة مستوى عالي
Visual Basic	لغة فيجول بيسك
Properties	الخصائص
Tools	الأدوات
Visible	مرئي
True	صحيح
False	خاطئ
Font	خط
Form	نموذج أو واجهة
TextBox	صندوق النص
RadioButton	زر الخيار
CheckBox	صندوق اختيار
ListBox	صندوق قائمة اختيار
ComboBox	الخانة المركبة
Condition	شرط
Expression	تعبير
Statement	جملة أمر

