الطبعة الأولى 2016

للمبتدأ والمتفوق والمتميز سلسلة صدى

بسم الله الرحمن الرحيم

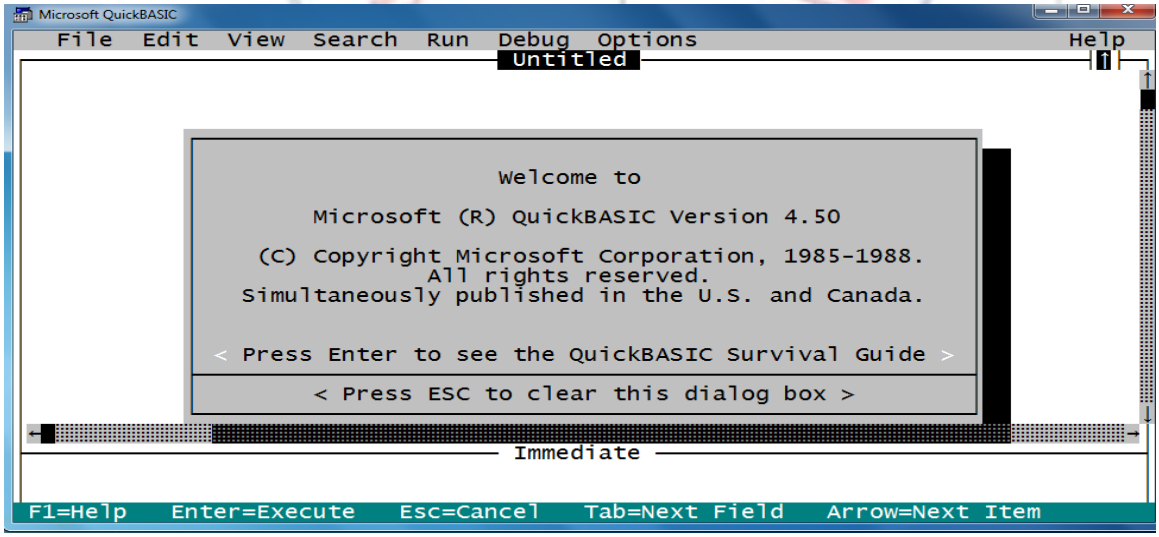
سلسلة مطبوعات صدى لطلاب المرحلة الثانوية

الميسر 2 في علوم الحاسوب

البرمجة بلغة بيسك

لطلاب الصف الثاني الثانوي

بإسلوب مُيسر وشرح عملى بلغة بيسك 100٪



إعداد وتصميم الأستاذ/

عثمان عوض حسن (صدى)

00249918084991

ما يميزنا تجده بمطبوعاتنا

طموحاتى الكبرى بعيدة جداً مثل الشمس ، قد لا أستطيع الوصول إليها ولكن يمكننى النظر لأعلى لرؤية جمالها ، والإيمان بها ، ومحاولة السعى وراءها حيث تقودنى

مقولة أعجبتنى

جميع حقوق الطبع والتصميم محفوظة يمنع النسخ أو التصوير أو التعديل

سلسلة مطبوعات صدى ، ، ، متوفرة لجميع الصفوف (الأول+ الثانى+الثالث)

لضمان التفوق إطلب سلسلة مطبوعات صدى ،،، متوفرة لجميع الصفوف للحصول عليها إتصل بالرقم الموضح

الوحدة الرابعة

البرمجة بلغة بيسك Basic Language Programming

تاريخ لغة بيسك Beginner All- purpose Symbolic Instruction Code

- هى لغة تعليمية سهلة ومبسطة تم تصميمها عام 1964 لتفى بغرض المعالجات العلمية غير المعقدة والحرفية البسيطة .
- صممت لغة بيسك كلغة رئيسية فى أقسام المؤسسات التعليمية التى تحتاج للحاسوب فى مشاريعها وتطبيقاتها ولكنها تعتبر كلغة إبتدائية فى أقسام علوم الحاسوب نفسها .
- ظلت لغة بيسك مغمورة حتى بداية الثمانينات أى بداية ثورة الحواسيب الدقيقة والتى إعتبرت لغة بيسك الرئيسية لها لسببين هما :

الأول والرئيسى : لأن الحواسيب الدقيقة كانت بها مشكلة صغر الذاكرة وقصر الكلمة لذلك كان مفسر بيسك هو الأنسب لصغر حجمه ومرونته .

الثانى : سهولة وبساطة لغة بيسك ولأنها مناسبة للمستخدمين الجدد والمستفيدين لأنه ليس لديهم أدنى خبرة فى مجال الحاسوب ولغات البرمجة المعقدة .

س/ **عرف لغة بيسك ؟** هى لغة تعليمية سهلة ومبسطة تم تصميمها عام 1964 لتفى بغرض المعالجات العلمية غير المعقدة والحرفية البسيطة .

س/ **صممت لغة بيسك فى المؤسسات التعليمية كلغة رئيسية وإبتدائية . لماذا ؟**

ج/ صممت فى أقسام علوم الحاسوب لأنها تُعلم البرمجة وأما فى الأقسام الأخرى لأنه تتم الحوجة لها فى تطبيقات الحاسوب فقط والمشاريع .

س/ **ظلت لغة بيسك مغمورة حتى بداية الثمانينات أى بداية ثورة الحواسيب الدقيقة . علل ؟**

ج/ لأن الحواسيب الدقيقة إتخذت لغة بيسك كلغة رئيسية لها .

س/ **لماذا إتخذت الحواسيب الدقيقة من لغة بيسك كلغة رئيسية لها ؟ لسببين هما :**

ج/ (١) السبب الأساسى مشكلة صغر الذاكر وقصر الكلمة .

(٢) لمناسبتها وسهولتها وبساطتها.

مفسر ومترجم بيسك

لا يتم التعامل مع تعليمات اللغات الراقية التى تكتب إلا بعد ترجمتها وتفسيرها إلى اللغة الثنائية أو لغة الماكينة وهذا ما يعرف بعملية التفسير أو الترجمة ويوجد نوعان من البرامج لترجمة اللغات من المستوى الأعلى إلى لغة الماكينة وهما المفسر والمترجم . كمثال لا يتعامل مع تعليمات لغة بيسك إلا بعد ترجمتها .

المفسر Interpreter :

هو برنامج يقوم بترجمة الأوامر والتوجيهات من لغات المستوى الأعلى إلى لغة الماكينة سطر بعد سطر

المترجم Compiler :

هو برنامج يقوم بترجمة الأوامر والتوجيهات من لغات المستوى الأعلى إلى لغة الماكينة دفعة واحدة .

خطوات عملية التفسير :

(١) يقوم المفسر بتحليل الأمر للتاكد من صحته بلغة الماكينة (أو وجود ما يقابله) .

(٢) ينفذ الأمر بعد ذلك .

المفسرات interpreters : تتلخص فكرة عمل المفسرات فيما يلى :

* عند تنفيذ أمر لعدة مرات يقوم المفسر بتكرار عملية الترجمة لأنه ينساها بعد التنفيذ مباشرة ولا يخزنها
* تعامل البرنامج كمجموعة أمر واحد متصلة ويعرف كل أمر بالذى يليه .
* عند الرجوع لأمر سابق تم تنفيذه يتم البحث بالتوالى عن الأوامر السابقة حتى تجده .
* تخزن المتغيرات المستعملة فى شكل جدول تبحث فيه عند الرجوع لأى منها .

المترجم compilers : تتلخص فكرة عمل المترجمات فيما يلى :

* تقوم بترجمة البرنامج دفعة واحدة ثم تكون الملف الهدف أو البرنامج الهدف .
* تقوم بوصل المتغيرات والقفزات (التفرع) فى البرامج بعناوين الذاكرة مباشرة .
* لا تحتاج البحث المتتالى أو الرجوع للأوامر عند إعادة تشغيل البرنامج .

مقارنة بين المفسرات والمترجمات :

المفسرات	المترجمات
تكرر الترجمة لأنها تنسى الأمر بعد التنفيذ	لا تكررها لأنها تقوم بوصل البرامج بعناوين الذاكرة
تعامل البرنامج كمجموعة أوامر متصلة	تعامله كامر ر واحد متصل لا يحتاج للرجوع للسابق
تحتاج البحث المتتالى عن إعادة التشغيل	لا تحتاج لأنها تقوم بوصل البرامج بعناوين الذاكرة
تخزن المتغيرات فى شكل جدول	لا تخزنها لأنها تقوم بوصل البرامج بعناوين الذاكرة

س/ **ما هو الفرق بين البرنامج الهدف والبرنامج المصدر ؟**

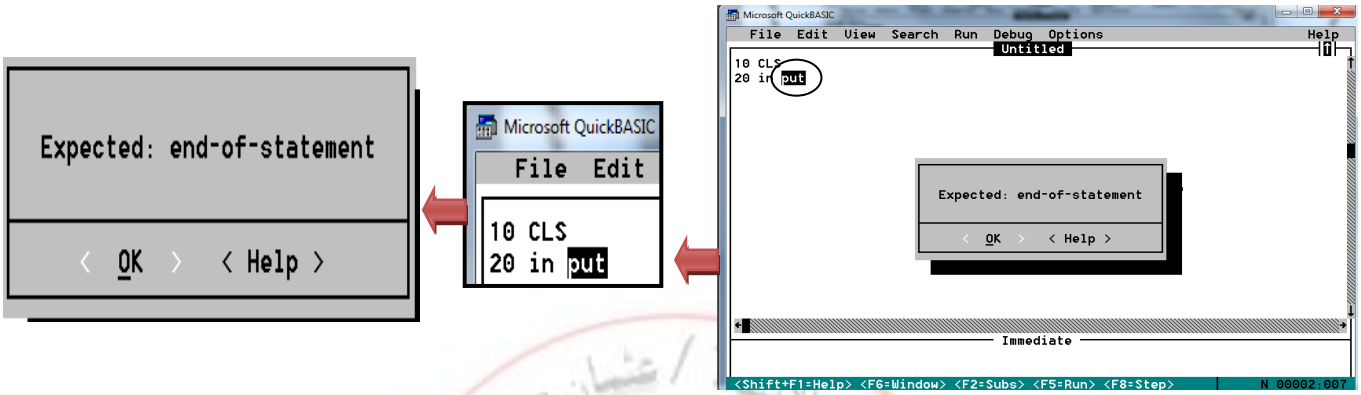
البرنامج الهدف (Object program) أو الملف الهدف (Object file) :

تعريف : هو البرنامج أو الملف الذى تمت ترجمته إلى لغة الماكينة (اللغة الثنائية).

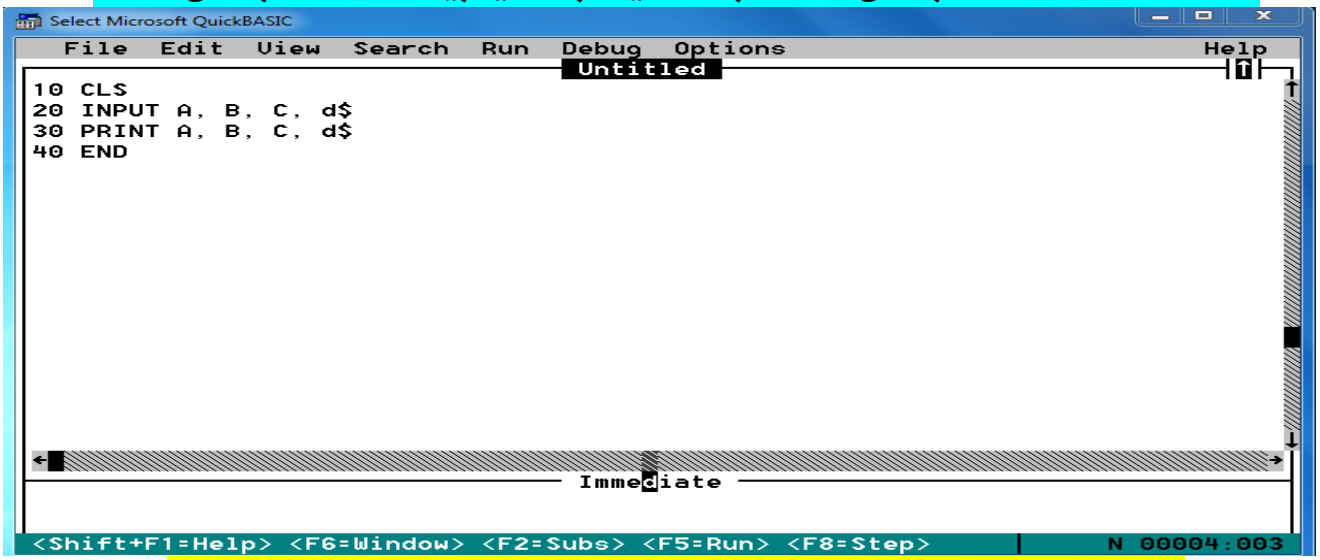
البرنامج المصدر (Source program) :

تعريف : هو البرنامج الأساسى المكتوب باللغة الراقية كمثال لغة بيسك .

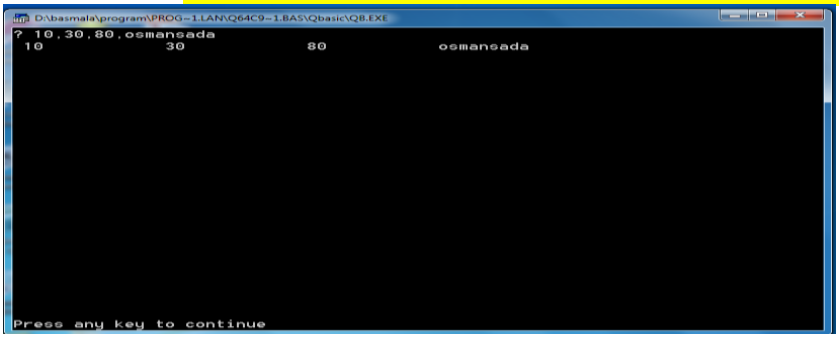
الشكل التالى هو لبرنامج تم تفسيره فى لغة كيو بيسك لاحظ الخطأ فى عبارة (Input) لذلك تم التوقف بعد السطر مباشرة



الشكل التالى هو نفس البرنامج تمت كتابته صحيحاً بلغة كيو بيسك ويمثل البرنامج المصدر



الشكل التالى يمثل البرنامج الهدف وهى المخرجات أو التنفيذ للبرنامج المصدر



فى التنفيذ تظهر العلامة ? وتم إدخال البيانات التالية من اليسار لليمين على التوالى 10,30,80, osmansada وكان الإخراج نفس البيانات

وفى الأسفل كتبت عبارة أضغط أى مفتاح للإستمرار أى للرجوع للبرنامج المصدر

المترجمات المثالية أو الممتازة :

تعريف : هى مترجمات خاصة تقوم بتنظيم الأوامر بطريقة إقتصادية فى السرعة والتخزين قبل التنفيذ

تتميز المفسرات على المترجمات بالآتى : (١) التخزين الإقتصادى (٢) التخاطب المباشر مع الحاسوب

و تتميز المترجمات على المفسرات بالآتى :

(١) السرعة (٢) التامين (٣) التركيب البنائى (٤) التكامل مع اللغات والبرامج

تمرين : ما هى عيوب المفسرات ؟ وما هى عيوب المترجمات ؟

القياسية فى لغة بيسك

س/ **ماذا نعنى بمصطلح القياسية ؟**

ج/ القياسية تعنى أن يتم تشغيل البرنامج المكتوب باى نوع من لغات بيسك فى أى حاسوب .

لا يعتبر مترجم بيسك قياسياً وبالتالى لا تعتبر لغة بيسك لغة قياسية (**Standard Language**) لأنها تختلف من حاسوب لآخر .

توضيح :

لن يعمل برنامج كتب بلغة بيسك على أجهزة وانق (**Wang**) فى أجهزة آى بى إم (IBM) لأنه توجد العديد من إصدارات (أنواع) لغة بيسك حتى فى أجهزة آى بى إم نفسها ولكن أوامر اللغة نفسها لا تختلف من نسخة لأخرى بينما تختلف البيئة فقط (أى نوع الإصدارة وشكلها وكيفيتها) .

أنواع أو نسخ أو إصدارات (Versions) لغة بيسك : هنالك العديد من الإصدارات منها مثلاً :

(١) بيسكا Basica (٢) جى دبليو بيسك GW BASIC (٣) كيو بيسك Q BASIC

(٤) فيجوال بيسك ٦ Visual Basic 6 (٥) فيجوال إستديو Visual Studio

س/ **لماذا لا تعتبر لغة بيسك قياسية ؟** ج/ لإختلاف بيئة التشغيل أو لأنها تختلف من حاسوب لآخر.ز

الترقيم فى لغة بيسك

يتم التنفيذ للأوامر بعد كتابة كل أمر فى سطر منفرد وإعطاء الأسطر أرقاماً بالتتالى (أنظر البرنامج فى الصفحة السابقة) ولكن هنالك بعض الإصدارات لا تشترط ترقيم الأسطر مثال (كيو بيسك) .

ملحوظة: البرنامج السابق كتب بلغة (كيو بيسك) إذا يمكن ترقيمه ويمكن عدم ترقيمه أيضاً .

شروط ترقيم الأسطر :

(١) يجب الترقيم بالترتيب تصاعدياً حسب الأولوية لأن التنفيذ يتم حسب أولوية الترقيم مثلاً ينفذ السطر بالرقم (**50**) قبل السطر بالرقم (**100**) حتى وإن كان السطر (**50**)هو أخر سطر فى البرنامج .

(٢) يجب إعطاء فوارق كافية بين ترقيم الأسطر مثلاً (**100**) ، (**200**) وذلك لإدخال أوامر أخرى بينها عند الحوجة إليها وإعطاءها الترقيم (**101- 199**) مثلاً .

س/ **ما هى شروط الترقيم فى بيسك ؟** (١) ترتيب الترقيم تصاعدياً (٢) إعطاء فوارق كافية بين الترقيم

س/ **وضح ما ذا نعنى بشروط الترقيم فى بيسك (أولوية الترقيم) ؟** راجع الصفحة السابقة .

الثوابت والمتغيرات فى بيسك

الثابت Constant : عبارة عن القيمة الحقيقية التى يتعامل معها البرنامج أثناء التشغيل .

تعريف آخر : موقع تخزين لا يمكن تغيير قيمته أثناء تشغيل البرنامج .

المتغير Variable : عبارة عن أسماء تستعمل لتمثيل القيم أو البيانات أثناء تشغيل البرنامج .

تعريف آخر : موقع تخزين يمكن تغيير قيمته (يقبل قيم جديدة) أثناء تشغيل البرنامج .

أنواع الثوابت :

(١) الثابت الرمزى (Character) أو السلسلى (String) (٢) الثابت العددى (Number)

الثابت السلسلى أو النصى أو الرمزى :

هو الذى يتكون من سلسلة رموز لا تتجاوز ٢٥٥ رمز وتكتب بين علامتى تنصيص **Double quotations** .

الثابت العددى (Number) :

هو الذى يكون فى شكل سالب أو موجب ولكن توضع علامة السالب قبل العدد مباشرة والموجب إختيارى .

أمثلة للثوابت العددية : " $250.000.00 " ، " -935.33 "

أمثلة للثوابت الرمزية أو السلسلية : " What is this " ، " Hello "

أنواع المتغيرات : (١) المتغير النصى (٢) المتغير العددى

المتغير النصى : هو الذى يمثل قيمة نصية أو حرفية وينتهى بعلامة $.

المتغير العددى : هو الذى يمثل قيمة رقمية أو عددية .

تتكون أسماء المتغيرات من حرف أو حرفين أو حرف ورقم أو أربعين رمزاً تبدأ بحرف ، ومن شروط تسميتها لا يمكن إستخدام الكلمات المحجوزة لأنها قد خصصت لأغراض معينة .

الكلمة المحجوزة : هى كلمة تم تعريفها فى اللغة لغرض معين لا تصلح لغيره وتكتب بصيغة معينة .

أمثلة للكلمات المحجوزة : فى لغة بيسك مثلا : (***CLS***) ، (***DIM***) ، (***EXP***) وغيرها من الكلمات .

فإن إستخدمنا كلمة (***EXP***) **بهذا الشكل** (***EXP = 5***) لا يقبلها الحاسوب لأنها محجوزة للدالة (***E***)

ولكن يمكننا التلاعب بالأحرف وتغييرها مثلاً ***EXPPP*** = 5 أو ***EXPONENT =5***

أمثلة للمتغيرات العددية : ***H =3*** ، ***EXPPP*** = 5 ، ***RD = 35***

أمثلة للمتغيرات النصية : H$ = " What is this " ، ***RHD$***= " Hello "

ملحوظة :

* عند إضافة علامة الهاش # فى المتغير تعنى الحوجة له بدرجة عالية من الدقة (***Double Precision***)
* وعند إضافة علامة فى المائة ٪ تعنى أن هذا المتغير العددى رقماً فقط (***Integer***) لتوفير التخزين .

المصفوفات البيانية *Arrays*

المصفوفة Array : هى مجموعة من القيم تخزن تحت إسم واحد أو متغير واحد .

المصفوفة فى الحاسوب مثلها مثل بناء الأبراج يجب وضع الأساس لها أولاً والأساس للمصفوفة يعنى تعريفها والإعلان عن حجمها عند بداية البرنامج بواسطة الكلمة المحجوزة (Dim) والتى تعنى البعد أو الأبعاد (Dimensions) وهى الطول والعرض أو عدد الصفوف والأعمدة .

أمثلة :

* تم الإعلان عن مصفوفة نصية عمودية تسمى B بها عمود واحد وطوله ٥ `10 dim B$(5)`
* تم الإعلان عن مصفوفة رقمية تسمى A بها ثلاثة أعمدة وطول كل عمود إثنان (صفان)

`20 dim A (2,3)`

* المصفوفة النصية السابقة تسمى B وقيمة الوحدة الثانية هى Title `30 B$ (2) = "Title "`
* المصفوفة الرقمية السابقة تسمى A وقيمة الوحدة فى الصف الثانى والعمود الثانى هى 10

`40 A (2,2) = 10`

تمرين : (١) عرف مصفوفة تتكون من ٣ أعمدة و ٥ صفوف ؟ (٢) ما نوع هذه المصفوفة ؟

(٣) قم بالإعلان عن متغير يسمى name فى شكل مصفوفة ؟ (٤) ما نوع هذا المتغير ؟

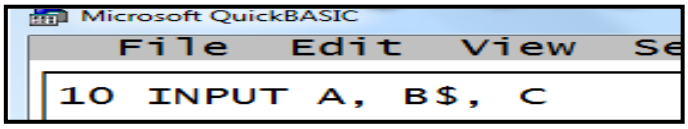
الإدخال والإخراج

توجد طريقتان لإدخال البيانات فى لغة بيسك وهما الإدخال المباشر من لوحة المفاتيح بواسطة الأمر Input أو الإدخال من البرنامج نفسه بواسطة الأمر Read

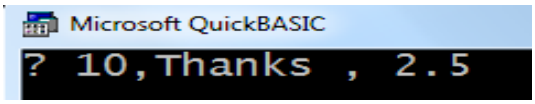
الإدخال المباشر :

* يتم الإدخال بالتخاطب المباشر عن طريق الشاشة بالأمر Input والتى تعنى أدخل .
* يمكن إدخال أكثر من متغير بالأمر Input (إذا كتبت فى سطر واحد يتم إدخالها فى نفس السطر).
* عند تشغيل البرنامج بالضغط على مفتاح F5 فى لوحة المفاتيح أو من قائمة Run فإن البرنامج يستجيب للإدخال بإظهار العلامة ? وينتظر الإدخال ثم بعد ذلك يعطى الإخراج . (راجع ص ٣) .
* يمكن إستخدام الأمر Input للتخاطب مع الحاسوب فى شكل أسئلة وأجوبة .

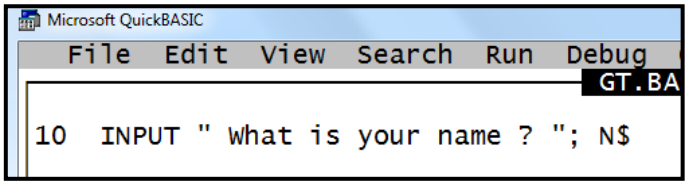
مثال لإدخال أكثر من متغير بواسطة الأمر **Input**

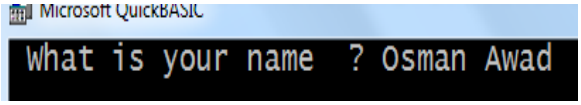


التنفيذ للبرنامج



مثال للتخاطب مع الحاسوب فى شكل إستجواب : (ملحوظة : إذا لم ندخل علامة ; يعتبر خطا فى الترجمة)

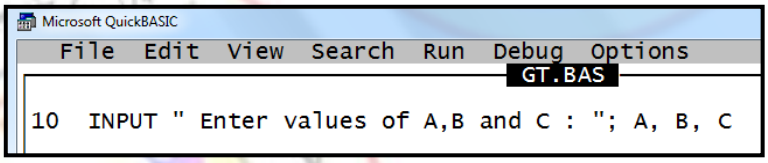


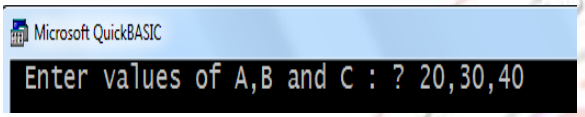


الشرح

قام البرنامج بطباعة عبارة **What is your name** وأعطى إشارة الإستجابة للإدخال وهى ? وبعدها قمنا بإدخال المتغير الذى تم تعريفه فى البرنامج بإسم N وهو سلسلى وأدخلنا **Osman Awad**

مثال آخر للتخاطب مع الحاسوب فى شكل إستجواب : إدخال القيم العددية **20,30,40**

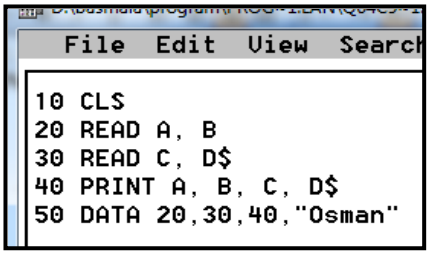


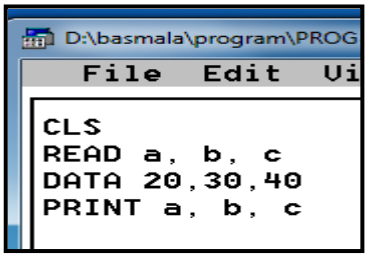


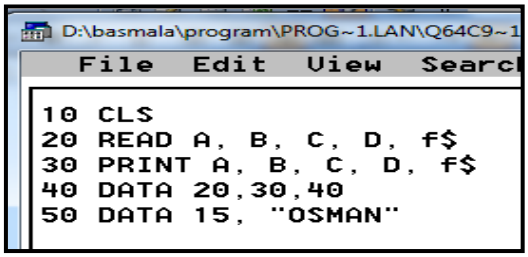
الإدخال عن طريق البرنامج :

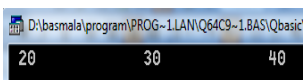
* يتم الإدخال عن طريق البرنامج بالأمر **Read** ويعنى إقرأ والذى يتبعه الأمر **data** مباشرة وتُعطى للقيم.
* لا يشترط أن تكون عبارة data بعد الأمر read مباشرة فيمكن كتابتها فى أى مكان .
* يفضل أن تكون العبارة data فى نهاية البرنامج وذلك لتسهيل تغيير البيانات ما لزم ذلك .
* يمكن أن يكون هنالك أكثر من أمر للعبارة Read وعبارة Data واحدة والعكس أيضاً صحيح .

اليك الأمثلة التالية مع التنفيذ :





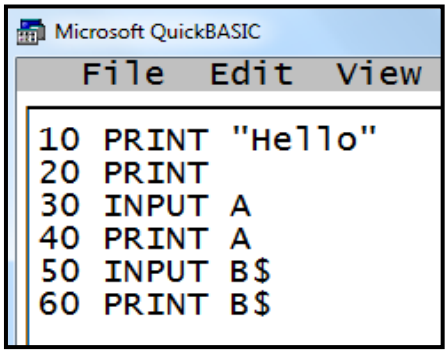


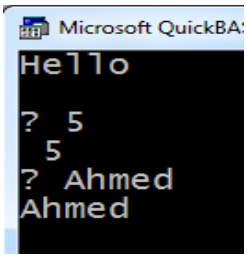


الإخراج فى لغة بيسك

يتم الإخراج فى لغة بيسك بالأمر **PRINT** أو **LPRINT** وهو أمر الطباعة على الشاشة أو الطابعة وإذا تم إستخدامه ولم يكتب بعده أى شىء يكون الناتج طباعة سطر فارغ .

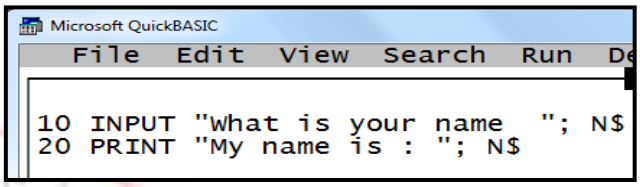
الشرح :

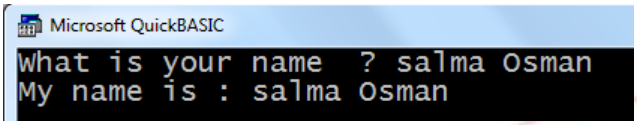




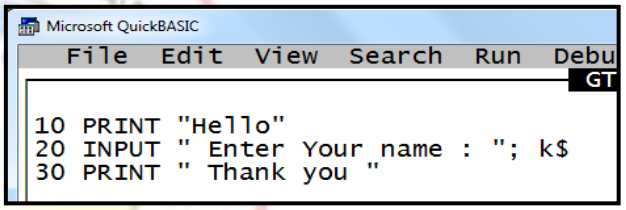
فى المثال تمت طريقة الطباعة مباشرة بالأمر **print** لعبارة **Hello** وبعدها تمت طباعة السطر الفارغ وبعدها إدخال المتغير **A** وطباعته وهو متغير عددى وتم إدخال 5 أما المتغير **B** فهو متغير حرفى سلسلى لذلك تبعته علامة $ حيث تم إدخال ***Ahmed*** إذاً بهذه الطريقة يمكن إستخدام عبارة الإخراج عموماً .

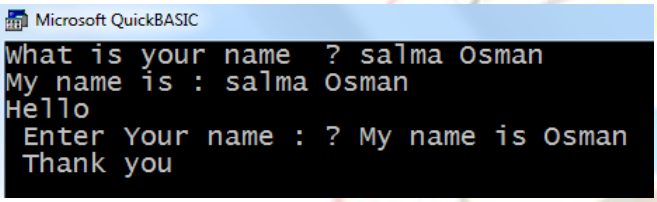
مثال : فى السطر الأول تمت طباعة ما هو إسمك وتم إدخاله وفى الثانى تمت طباعة إسمى هو كذا





مثال : أكتب برنامجاً يقوم بالترحيب بالمستخدم ويطلب منه إدخال متغيرات وفى النهاية يشكره ؟



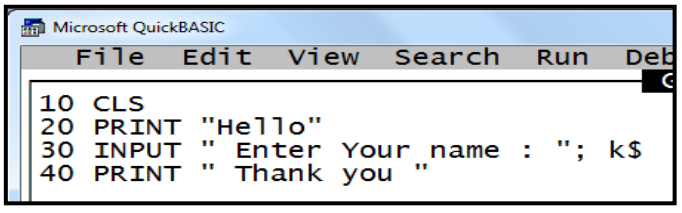


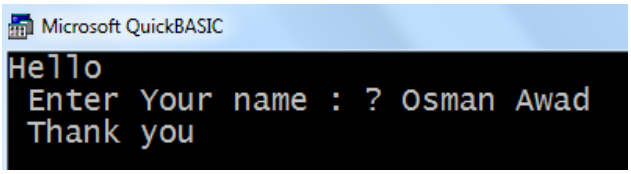
ملحوظة : قد تتساءل لماذا تم إظهار مخرجات البرنامج السابق هنا فى تنفيذ هذا البرنامج بالذات ؟

الإجابة لأننى لم أستخدم الأمر **CLS** ففى كل برنامج من البرامج السابقة كنت أستخدم هذا الأمر بكتابته فى السطر الأول قبل كتابة البرنامج ثم تنفيذه فيتم مسح كل الشاشة السابقة وبعده أكتب البرنامج (أنظر ص12)

س/ **ما هى الفائدة من أمر مسح الشاشة ؟** مسح كل مدخلات ومخرجات البرامج الأخيرة

لتوضيح ذلك لاحظ الفرق بين مخرجات نفس البرنامج السابق بإستخدام عبارة مسح الشاشة .





أولوية المعالجات الحسابية فى لغة بيسك

تتم العمليات الحسابية فى لغة بيسك أولاً بحساب القوى وثانياً حساب الضرب والقسمة وبعدها حساب الجمع والطرح وتكون من اليسار لليمين إذا تساوت العمليات فى الأولوية وإذا وجدت عملية حسابية داخل أقواس يتم فك الأقواس قبل كل ذلك ثم تنفذ كما سبق شرحه .

(١) حساب القوى (٢) حساب الضرب والقسمة (٣) حساب الجمع والطرح

قاعدة رياضية للتذكر :

الضرب والقسمة يسبقان الجمع والطرح والرفع إلى القوى أولى وفك الأقواس أولى منهما

ملحوظة :

فى لغة بيسك علامة الضرب هى النجمة (*) والقوى هى نجمتان (* *) أو علامة (↑) .

مثال (١) : ما ناتج العملية التالية (العملية من اليسار لليمين) : 15/3 - 25 + 6**

* أولاً يقوم الحاسوب بحساب القوى $2^5 = 32$ ، ثم يحسب ناتج القسمة (15 ÷ 3) = 5

* بعد ذلك يقوم بحساب ناتج القسمة ناقص القوى (32) - (+5) = -27

* بعد ذلك وأخيراً يقوم بإضافة 6 للناتج ويعطى النتيجة النهائية (6) + (-27) = -21

مثال (٢) : ما ناتج العملية التالية : 14/4 * 2↑3 - 13

* أولاً يقوم الحاسوب بحساب القوى $2^3 = 8$ ، ثم يحسب ناتج القسمة (14 ÷ 4) = 3.5

* بعد ذلك يقوم بحساب ناتج القوى مضروباً فى ناتج القسمة (8) × (3.5) = 28

* بعد ذلك وأخيراً يقوم بطرح 13 من الناتج ويعطى النتيجة النهائية (28) - (13) = 15

مثال (٣) ما ناتج العملية التالية : 15/3 - 2(5 + 6)**

* أولاً يقوم الحاسوب بفك الأقواس = 11 ، ثم يحسب القوى $2^{11} = 2048$

* بعد ذلك يقوم بحساب ناتج القسمة مطروحاً منه ناتج القوى (5) - (2048) = -2043

لاحظ الأولوية كانت لفك الأقواس فى هذا المثال بعكس المثالين السابقين .

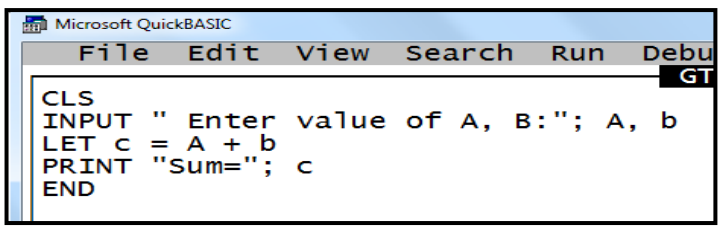
تدريب : (١) أحسب بطريقة الحاسوب : (2 * 4 + 8 / 5 ↑ 2) (4 * 2 – 3 ↑ 2 - 12)

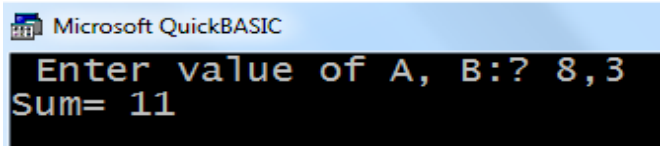
لابد أن يكون الناتج النهائى 144 هكذا من اليسار $2^3 = 8$ ثم (4 * 2) = 8 =(0-12)= 12

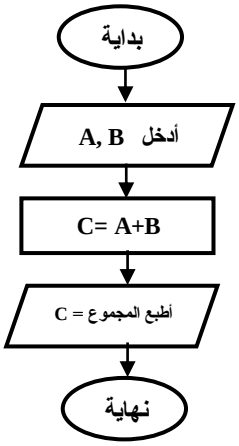
والقوس الثانى على اليمين $2^5 = 32$ ثم (32 ÷ 8) = 4 ثم (8 + 4) = ثم 12 ×12 = 144

(٢) أرسم مخططاً وأكتب برنامجاً يقوم بإدخال عددين وجمعهما وطباعة النتيجة ؟

ملحوظة : يمكن إستخدام الأمر LET والتى تعنى أحسب ويمكن تجاهله ، بعد التنفيذ (8 + 3 = 11)





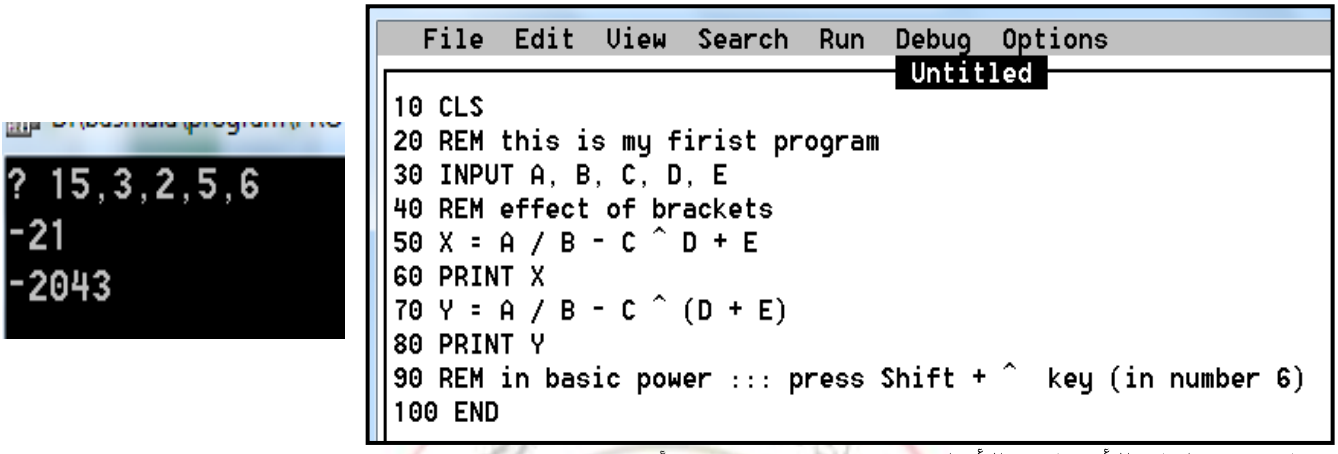


كتابة الملاحظات والتعليقات فى بيسك

تعريف : الملاحظات عبارة عن تعليقات أو تذكير يكتب داخل البرنامج بالعبارة ***REM*** ولا ينفذها الحاسوب .

ملحوظة : ***REM*** هى كلمة محجوزة من الكلمة ***REMARK*** .

س/ **ما هى فائدة كتابة الملاحظات والتعليقات ؟** لوجود حقائق مهمة ومعلومات مفيدة للمبرمج .



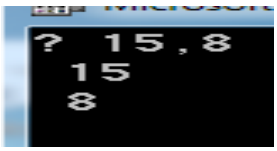
لاحظ تم تجاهل الأسطر بالأرقام (20 ، 40 ، 90) لأنها بدأت بعبارة **Rem** وهى الملاحظات أو التعليقات

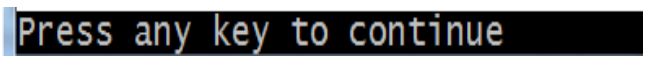
الإيقاف المؤقت وإنهاء البرنامج فى بيسك

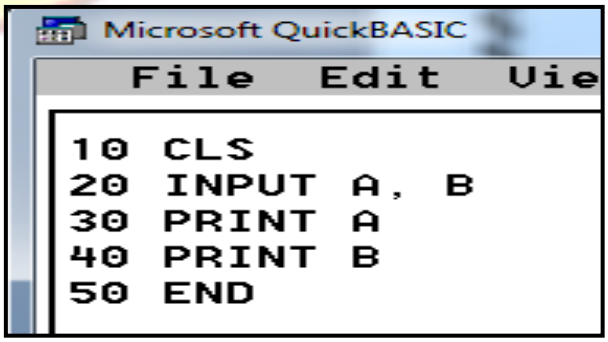
يستخدم الأمر **Stop** لإيقاف البرنامج مؤقتاً وللإستمرار مرة أخرى نكتب الأمر **Cont** أما لإنهاء البرنامج نكتب الأمر **End** .

ملحوظة : فى إصدارة لغة كيو بيسك هذه تظهر رسالة تلقائياً أسفل شاشة التنفيذ وذلك للرجوع للبرنامج بمجرد الإدخال وهى (أضغط أى مفتاح للإستمرار) ***Press any key to continue***

أنظر البرنامج التالى والتنفيذ يظهر الرسالة أسفله . (راجع ص **3** وأنظر الرسالة أسفل الشاشة)





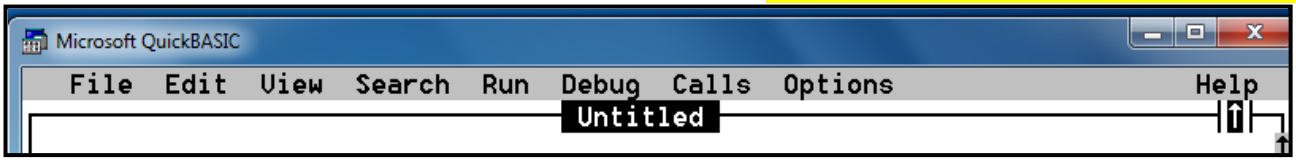


بعض أوامر نظام التشغيل فى التعامل مع لغة بيسك

توجد مجموعة من الأوامر للتعامل مع البرنامج عند كتابته كتخزينه أو مسحه من الذاكرة أو طباعته .

لتنفيذ البرنامج يتم الضغط على مفتاح **F5** أو إختيار الأمر **Run** من قائمة **Run** أو الضغط على مفتاحى **Shift** مع **F5** عموماً يتم إختيار الأمر المطلوب حسب القائمة المحددة من قوائم البرنامج فيتم تفسير وترجمة البرنامج وإن كانت هنالك أخطاء يتم تحديدها أو يتم التنفيذ مباشرة (راجع ص 3) .

أنظر القوائم الرئيسية التالية لبرنامج بيسك .



(١) أمر الحفظ Save أو Save As : هو أهم الأوامر الموجودة فى الحاسوب لحفظ الملفات والبيانات .

س/ **ما هو الفرق بين الأمرين Save و Save As ؟**

الأول للحفظ عموماً والثانى هو لحفظ نفس الملف ولكن بإسم مختلف .

س/ **لماذا يتم حفظ الملفات والبيانات فى الحاسوب ؟** للحصول عليها عند الإسترجاع أو بهدف الإسترجاع .

توضيح : مثلاً حفظ النغمات والصور والبرامج فى ذاكرة الهاتف والرجوع لها عند الحوجة .

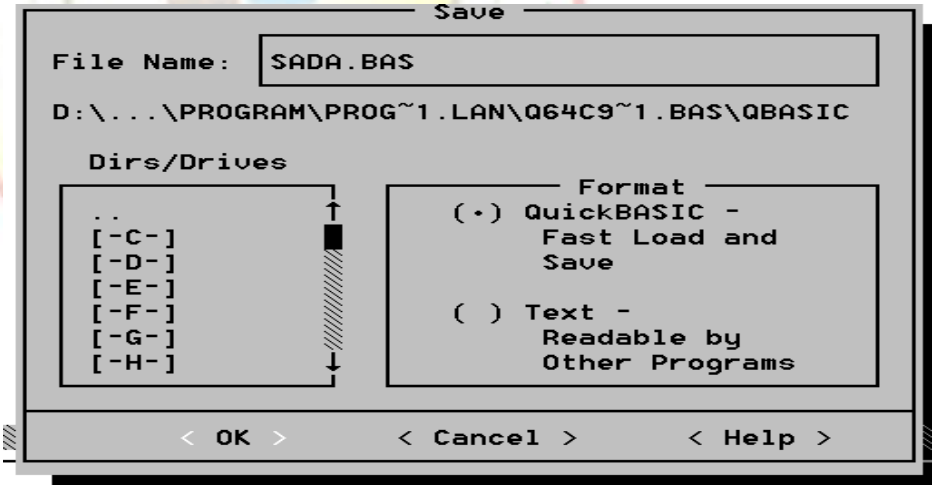
س/ **كيف يتم الحفظ ؟** بعد إختيار حفظ من قائمة **File** نكتب إسم الملف **File name**

أو بهذه الطريقة : **File name** . **bas**

(١) (٢) (٣)

(١) يكتب فيها إسم الملف (٢) للفصل بين الإسم ونوع الملف

(٣) نوع أو إمتداد الملف حيث bas يعنى (بيسك)



(٢) الأمر Load : وهو لتحميل البرنامج من القرص ويتم كتابة إسم البرنامج بعد إختيار الأمر .

إذاً يكون الأمر بهذه الصيغة load " file name "

(٣) الأمر New : وهو لفتح ملف جديد وحذف الملف الحالى من الذاكرة .

س/ **ماذا يحدث إذا تم الضغط على الأمر New أثناء التعامل مع برنامج آخر غير محفوظ ؟** يحذف نهائياً ويتم فقده

س/ **كيف يتم تفادى هذه المشكلة حتى لا يتم الضغط عن طريق الخطأ ؟** نستخدم الأمر حفظ دائماً فى كل مرة

س/ **عندما يتم الضغط على حفظ فى كل مرة هل يتم التعامل معه بنفس طريقة الحفظ أعلاه؟** لا إنما تحفظ التغيرات فقط

(٤) الأمر CLS : وهو يعنى **Clean Screen** أى مسح الشاشة نهائياً من أى كتابة .

س/ ما الفرق بين الأمر New والأمر CLS ؟ الأمر **CLS** يمسح الشاشة من أى مخرجات أو مدخلات أخرى للبرامج بينما الأمر **New** يمسح البرنامج نفسه . (راجع ص 8 شرح الأمر CLS) .

(٥) الأمر LIST أو LLIST : وهو لإظهار كل البرامج بالذاكرة على الشاشة أو الطابعة .

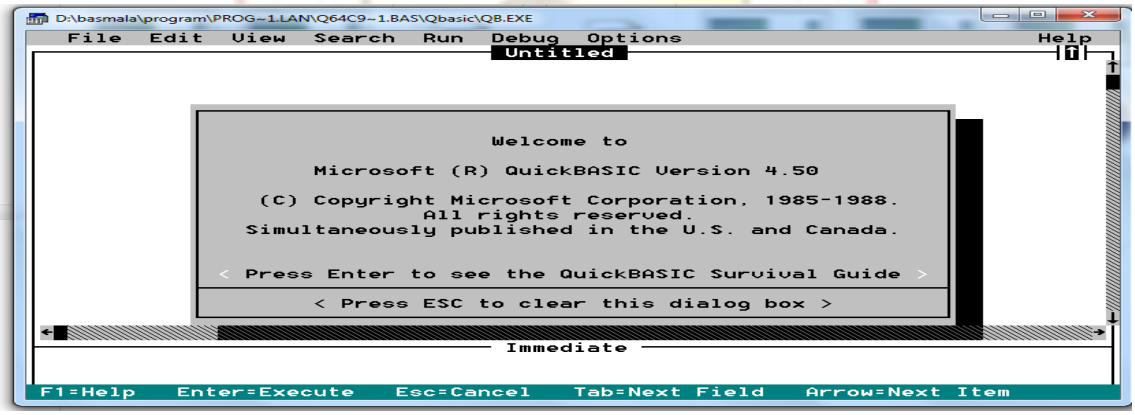
يمكن أيضاً إظهار بعض أسطر البرامج بكتابة الأمر يتبعه رقم السطر أو أرقام الأسطر بهذه الطريقة **LIST 30-50** ويعنى إظهار الأسطر من 30 إلى رقم 50 وإذا كتبت 30 فإنه يظهر هذا السطر فقط .

(٦) الأمر RENUM : وهو لإعادة ترقيم الأسطر فمثلاً عندما نكتب (**RENUM 1,2,3**) يكون الناتج **10 ,20, 30** أما إذا كتبت بمدى **RENUM 10,20,30** ينتج **100 ,200, 300** وهكذا .

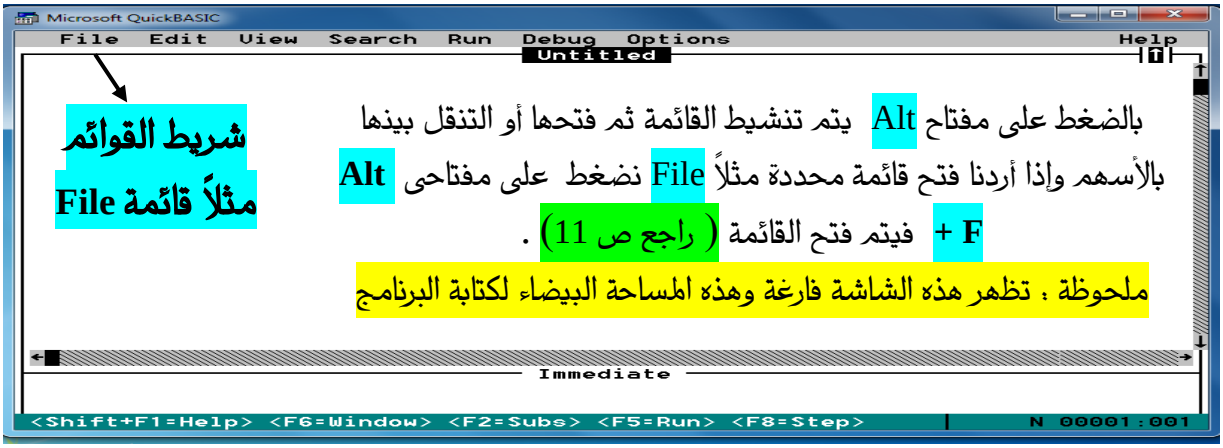
(٧) الأمر SYSTEM : وهو لإيقاف المعالجة بمفسر بيسك والإنتقال لتنفيذ أوامر نظام التشغيل .

(٨) الأمر BASIC : وهو للرجوع مرة أخرى لمفسر بيسك وتظهر علامة المفسر على الشاشة (OK)

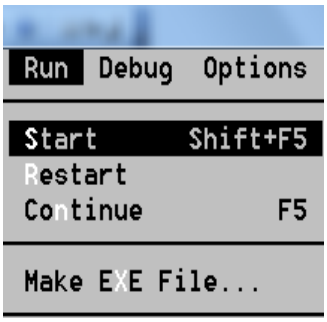
ملحوظة : أولاً : تعمل معظم الحواسيب الحالية بنسخة **Q.Basic** أى النسخة التى نفذنا عليها هذه البرامج ويتم تحميل الأوامر السابقة من نظام التشغيل دوس بطباعة **Q.Basic** لتظهر شاشة ترحيبية وهى الشاشة الأولى أسفل الصفحة وبها خيارين الأول **Enter** للدخول للتعليمات المساعدة ودليل الإستخدام والثانى **ESC** للدخول للشاشة الرئيسية للبرنامج أى الشاشة أسفل هذه الصفحة ويتم فيها كتابة البرنامج .



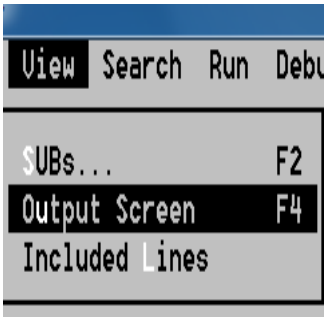
ثانياً : الأوامر السابقة جميعها لا يتم كتابتها فى هذه النسخة وإنما يتم إختيارها من شريط القوائم بالأعلى



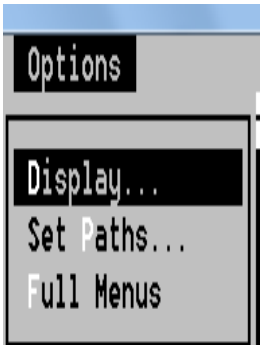
ومثلاً لفتح قائمة **Run** نضغط على **Alt + R** فتظهر الخيارات **START** لبدء تنفيذ البرنامج وإذا لم يوجد برنامج تظهر الشاشة فارغة وتطلب الضغط على أى مفتاح للرجوع ويكتب أسفل الشاشة ، **RESTART** لإعادة تنفيذ نفس البرنامج من الأول مرة أخرى ، **CONTINUE** لمتابعة تنفيذ البرنامج من آخر سطر توقف فيه وغيرها .

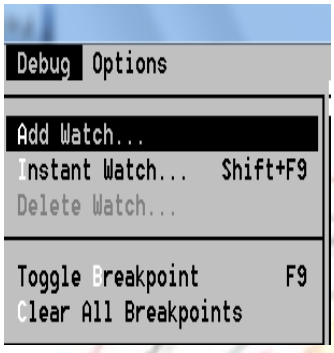


أما قائمة **View** تظهر قائمة منسدلة بها ثلاثة أوامر بالترتيب وهى الأول لكتابة برنامج فرعى والثانى لرؤية شاشة المخرجات والثالث لتضمين بعض الأسطر .



هنالك قوائم أخرى كما موضحة جانباً على اليسار





التفرع والتحكم فى بيسك

التفرع هو القفز لتنفيذ أوامر فرعية أما التحكم فهو السيطرة على البرنامج وإيقاف الحلقة اللانهائية .

س/ **ما هى الحلقة اللانهائية (الحلقة المغلقة) أو المصيدة ؟** تنفيذ أمر معين دون تحديد شرط التوقف (اللانهاية).

الصيغة العامة للتفرع أو القفز :

إذا تحقق الشرط إذاً نفذ السطر رقم كذا **IF (Condition) Then (Line Number)**

الحالات المنطقية التى يتحقق منها الحاسوب حتى ينفذ القفز والتحكم هى :

(١) **تحقق التساوى أو التكافؤ :** وهو أن يتساوى الطرفين مثلاً A=B أو A$ = B$

(٢) **تحقق الشرط الأكبر :** وهو أن يكون أحد الطرفين أكبر مثلاً A > B أو A$ > B$

(٣) **تحقق الشرط الأصغر :** وهو أن يكون أحد الطرفين أصغر مثلاً A< B أو A$ < B$

(٤) **تحقق التساوى أو الشرط الأكبر معاً :** وهو أن يتحقق أحد الشرطين لأحد للطرفين مثلاً A > =B

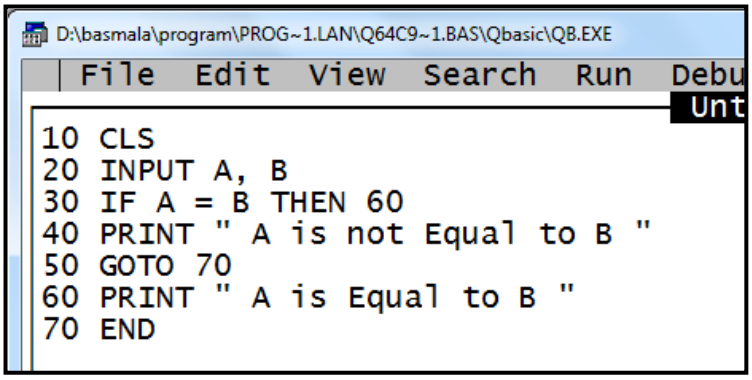
(٥) **تحقق التساوى أو الشرط الأصغر معاً :** وهو أن يتحقق أحد الشرطين لأحد للطرفين مثلاً A < =B

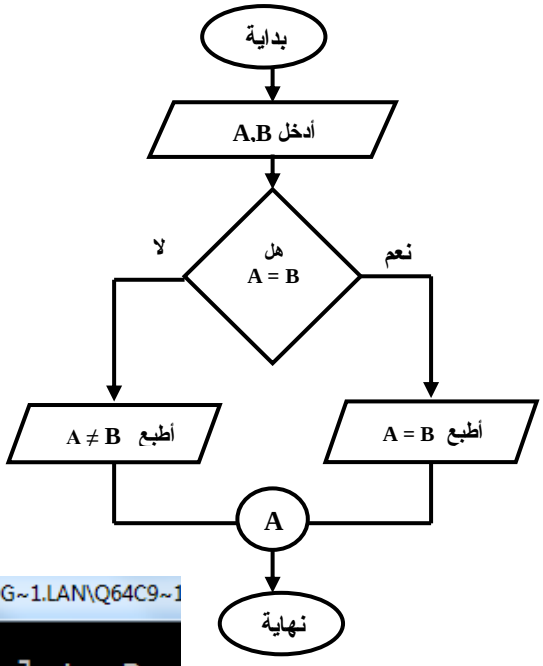
س/ **ما هى الشروط التى تحقق التحكم أو القفز ؟**

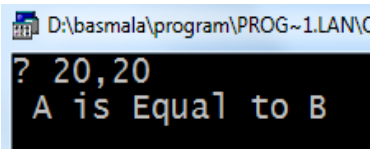
التساوى أو التكافؤ ، الأكبر ، الأصغر ، التساوى أو الأكبر ، التساوى أو الأصغر

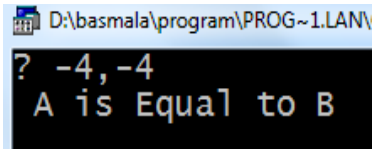
مثال : أرسم مخططاً وأكتب برنامجاً لإدخال عددين وإختبار هل هما متساويان أم أن أحدهما أكبر ؟

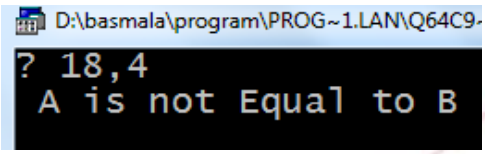
ملحوظة : عند تحقق أحد الحالات السابقة يقفز البرنامج إلى السطر المحدد بعد كلمة **Then** لتنفيذه وتنفيذ الأوامر التى تليه مباشرة بالترتيب أو يستمر فى تنفيذ البرنامج بدون قفز إمن لم تتحقق الحالة .

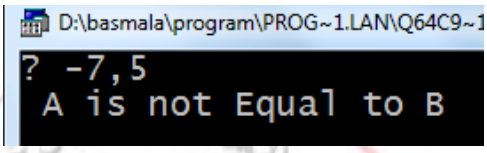












تم تنفيذ هذا البرنامج أربع مرات كما يلى :

الأولى 20,20 وهما عددين موجبين متساويين وكانت النتيجة أن الأول يساوى الثانى

الثانية 4-, 4- وهما عددين سالبين متساويين وكانت النتيجة أن الأول يساوى الثانى

والثالثة موجب أصغر وسالب أكبر والرابعة موجب أكبر وموجب أصغر وهما عددين غير متساويين وكان النتيجة أن الأول لا يساوى الثانى وعند تنفيذ البرنامج وإدخال الحالة الأولى والثانية تم القفز مباشرة إلى السطر رقم 60 وتم تجاهل ما قبله لعدم تحققه ، وفى الحالة الثالثة والرابعة تم تنفيذ البرنامج بالتوالى مباشرة من أول سطر إلى آخره بتجاهل السطر رقم 30 .

التفرع المقيد والغير مقيد :

التفرع المقيد : هو التفرع المشروط تبعاً لعبارة **IF - Then** أى يتقيد بوجودها بالبرنامج .

التفرع غير المقيد : هو التفرع الذى لا يتقيد بعبارة **IF - Then** أى لا يشترط وجودها بالبرنامج .

ملحوظة :

(١) إذا كان التفرع مقيداً أم غير مقيداً يكون بالعبارة **GoTo** يتبعها رقم السطر أنظر البرنامج أعلاه .

(٢) ليعمل التفرع بطريقة مفيدة فى البرنامج لا بد أن يكون مقيداً أو مسيطراً عليه .

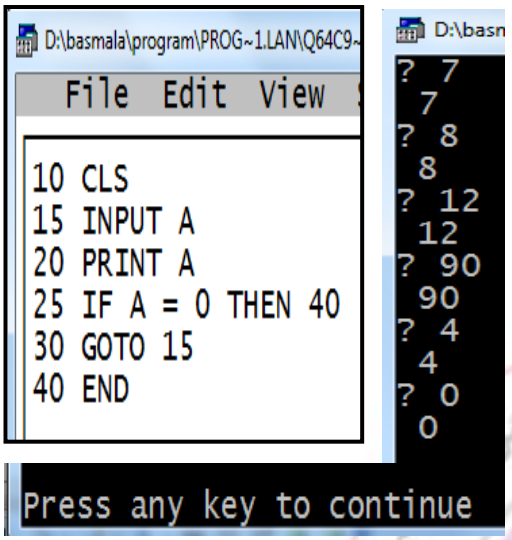
الحالات التى تحدث فى حالة عدم التقيد :

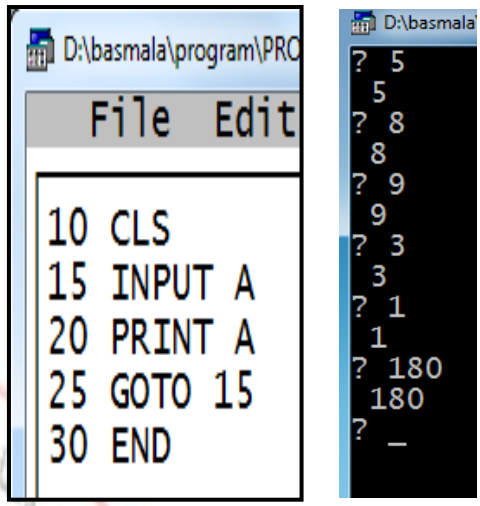
(١) إستمرار تنفيذ البرنامج فى حلقة غير منتهية أو حلقة مغلقة . لاحظ المثال بالصفحة التالية .

(٢) عدم تنفيذ الأوامر بين الأمر **GoTo** والأمر اللاحق له لذا يجب ضبطها إذا كانت ضرورية وعدم كتابتها إذا كانت غير ضرورية أصلاً . لاحظ المثال فى الصفحة التالية .

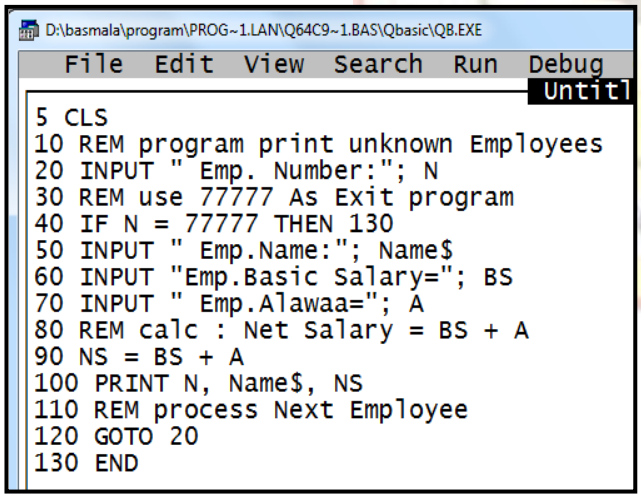
الشاشة على اليمين أدناه توضح دخول البرنامج فى مصيدة وسيظـل يعمل ويطلـب إدخال القيم إلى عدد لا نهائى من المرات لاحظ المتغيرات التى تم إدخالها وفى كل مـرة يطلب إدخال مـتغير ويطبعه والسـبب هـو عبارة التقيد غير المشروط (إذهب إلى السطر 15) التى أدخلته فى عدد لا نهائى من مرات التنفيذ .

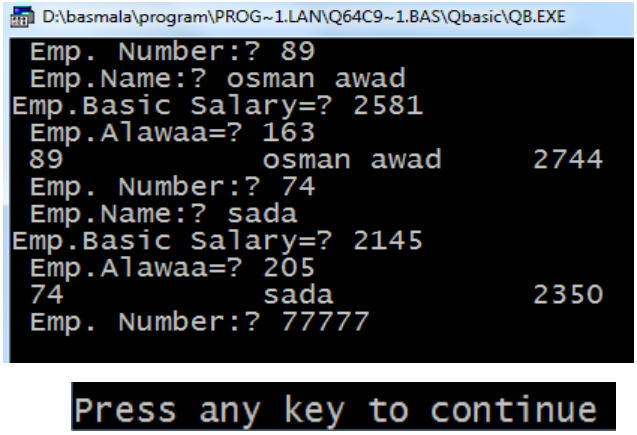
أما البرنامج باليسار فهو مشروط لأن الحلقة المغلقة سيتم إيقافها عند إدخال 0 ويعطى رسالة أضغط أى مفتاح للإستمرار أى للرجوع للبرنامج عدا ذلك سوف تستمر إلى ما لا نهاية .

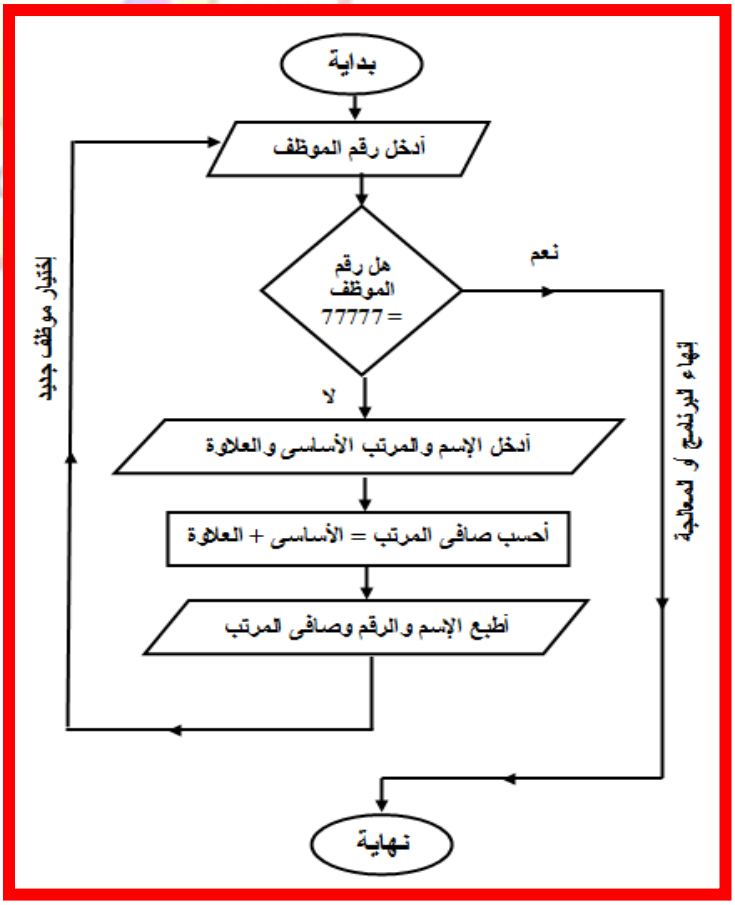




مثال : أرسم مخططاً وأكتب برنامجاً لإدخال إسم موظف ورقمه وراتبه الأساسى وعلاوته ثم يطبع تقريراً على الشاشة فيه الإسم والرقم وصافى المرتب ؟





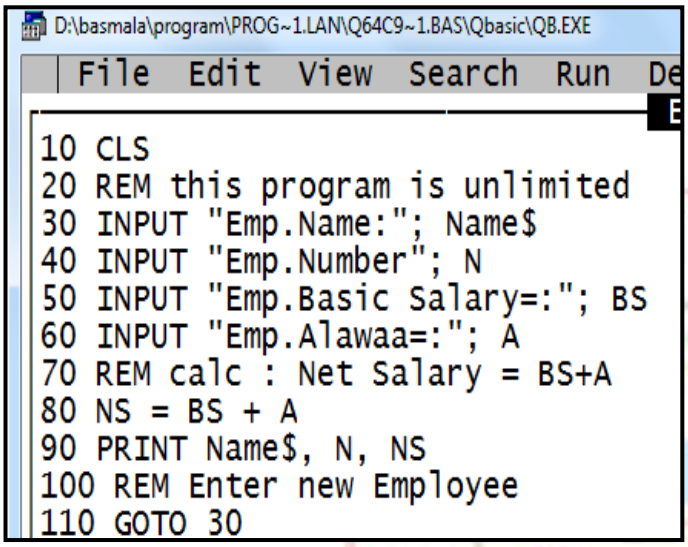


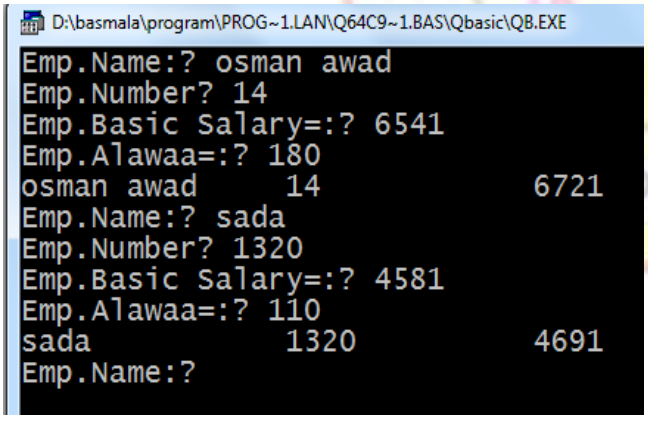
المخطط السابق والبرنامج لعدد غير معلوم ويلاحظ فيه :

أولاً : تم إستخدام التعليقات فى الأسطر التالية : 10 ليفيد أن البرنامج يطبع عدد لا نهائى من الموظفين وكذلك السطر رقم 30 ليفيد أن الرقم 77777 هو شرط توقف الحلقة والسطر رقم 80 ليوضح طريقة حساب صافى المرتب وأخيراً الرقم 110 ليوضح إكتمال المعالجة والإنتقال للموظف التالى .

ثانياً : تم إستخدام التقيد المشروط وهو إدخال رقم الموظف الوهمى وهو شرط توقف الحلقة ، لاحظ التنفيذ تم تنفيذ عدد 2 موظف وفى الثالثة تم إدخال شرط التوقف فظهرت عبارة أضغط أى مفتاح للإستمرار دلالة على نهاية البرنامج وبهذه الطريقة أيضاً للبرنامج التالى مع العلم بعدد الموظفين .

مثال : المخطط التالى لعدد لا نهائى من الموظفين .





السطر الأول لمسح البرنامج السابق والسطر الثانى تعليق بان هذا البرنامج لعدد لا نهائى من الموظفين والسطر الثالث إدخال الإسم والرابع الرقم والخامس المرتب الأساسى والسادس العلاوة والسابع تعليق لطريقة حساب صافى المرتب والثامن حساب صافى المرتب والتاسع طباعة الصافى والاسم والرقم والعاشر تعليق بإدخال موظف جديد والسطر قبل الأخير عودة للإدخال مرة أخرى وهكذا سيظل يعمل إلى مالا نهاية.

مثال : المخطط والبرنامج بالصفحة التالية لعدد لا نهائى من الموظفين .

لاحظ فى السطر رقم 20 تعليق بان هذا البرنامج لعدد N من الموظفين وفى السطر رقم 30 يطلب إدخال عدد الموظفين المطلوب مثلاً فى البرنامج قمنا بإدخال 3 أى يجب أن يحسب البرنامج لعدد 3 موظفين فقط وفى السطر رقم 40 تعليق بان القيمة الإبتدائية للعداد Counter هى 0 وفى السطر رقم 50 إعطاء القيمة الإبتدائية للعداد والأسطر من 60 إلى 90 لإدخال الإسم والرقم والمرتب الأساسى والعلاوة والسطر رقم 130 لزيادة العداد بواحد والشرط إذا كان العداد أصغر من N إستمر وأذهب للموظف التالى

Microsoft QuickBASIC

File Edit View Search Run Debug Options

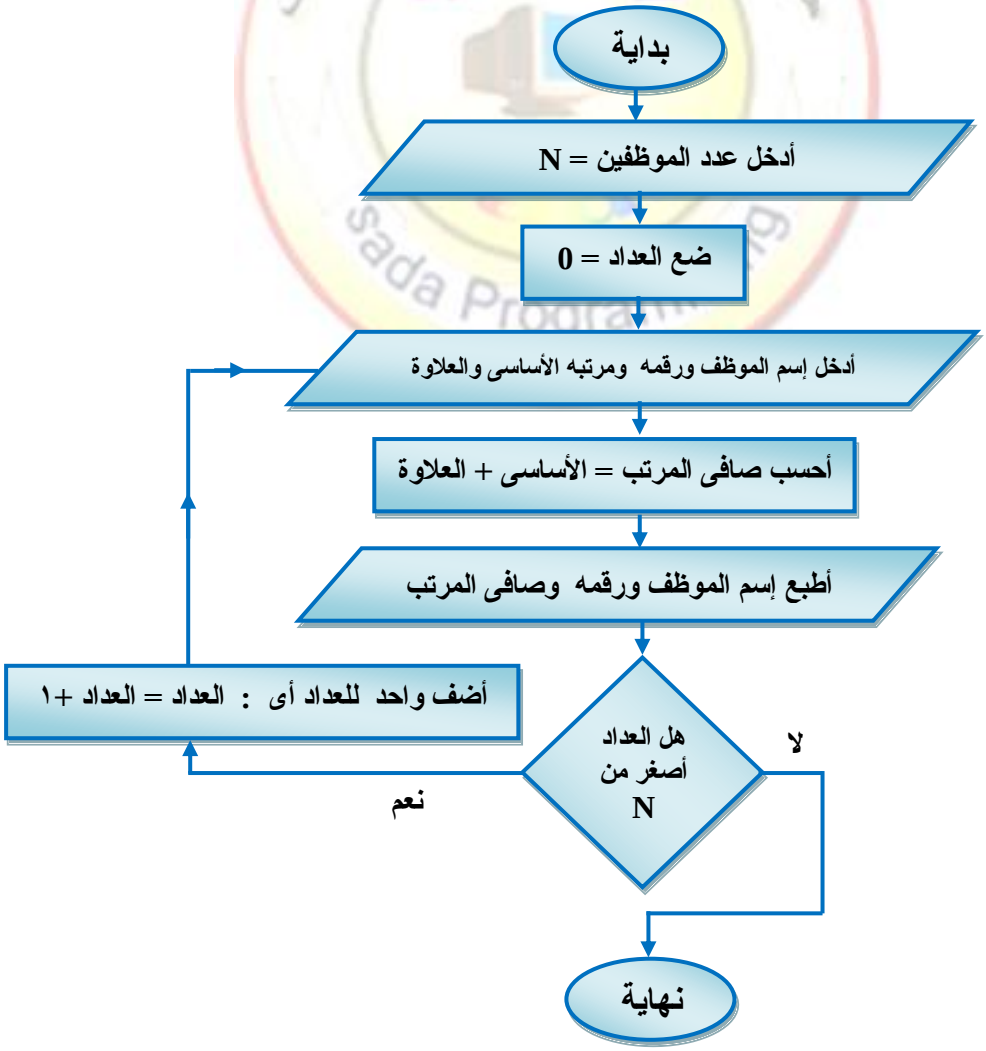
EMP_N2.BAS

```
10 CLS
20 REM this program is N Employees
30 INPUT "total No of Employees:"; N
40 REM initial value of Zero
50 counter = 0
60 INPUT "Emp.Name:"; Name$
70 INPUT "Emp.Number"; Nu
80 INPUT "Emp.Basic Salary=:"; BS
90 INPUT "Emp.Alawaa=:"; A
100 REM calc : Net Salary = BS+A
110 NS = BS + A
120 PRINT Name$, N, NS
130 counter = counter + 1
140 IF counter < N THEN 60
150 PRINT "-------- End of process --------"
160 PRINT " *** By osman Awad Hassan *** "
170 END
```

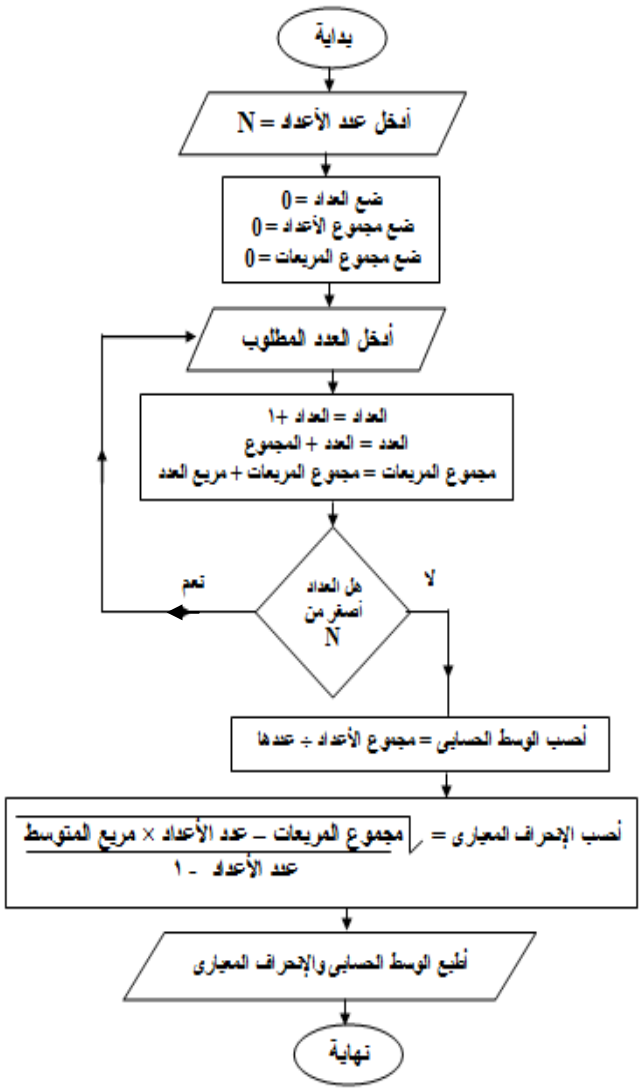
Microsoft QuickBASIC

```
total No of Employees:? 3
Emp.Name:? salma
Emp.Number? 14
Emp.Basic Salary=:? 2681
Emp.Alawaa=:? 120
salma          3              2801
Emp.Name:? sada
Emp.Number? 19
Emp.Basic Salary=:? 2319
Emp.Alawaa=:? 180
sada           3              2499
Emp.Name:? osman awad
Emp.Number? 8
Emp.Basic Salary=:? 3917
Emp.Alawaa=:? 170
osman awad     3              4087
-------- End of process --------
 *** By osman Awad Hassan ***
```

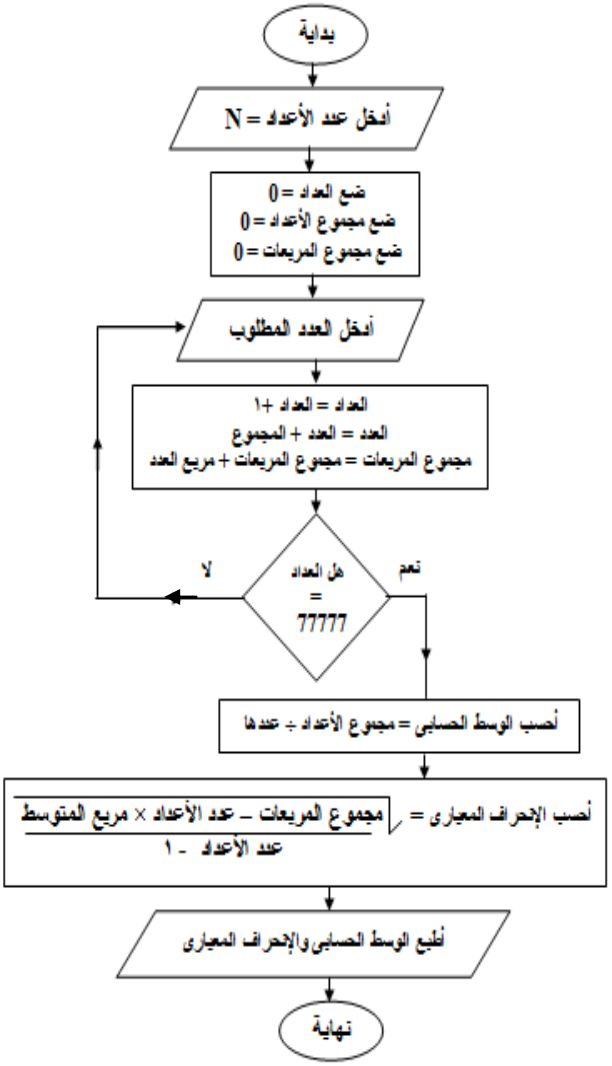
لاحظ أن الشرط طالما العداد أصغر من عدد الموظفين إختر موظف جديد وفى البرنامج يعادل ذلك السطر رقم 140 أما الأسطر 150، 160 فهى طباعة عادية ويمكن عدم كتابتها لذلك لم نكتبها بالمخطط وهذا الرسم التدفقى بالأسفل هو نفسه بالكتاب صفحة 54 ولكن هنا بطريقة مشابهة تماماً للبرنامج .



مثال : أكتب برنامجاً يحسب المتوسط
ومربع الإنحراف المعيارى لأعداد عددها N



مثال : إذا كان عدد الأعداد غير معلوم
نستخدم الرقم الوهمى




```
Microsoft QuickBASIC
File  Edit  View  Search  Run  Debug  Option
                                VARIANCE.BAS
10 REM average and variance of numbers
20 INPUT "Enter data sizeof N:"; n
30 REM initial sum,sumsq,counter=0
40 sum = 0
50 counter = 0
60 sumsq = 0
70 INPUT "Enter data of X= "; X
80 REM make sum and sumsq of squares
90 sum = sum + X
100 sumsq = sumsq + X * X
110 counter = counter + 1
120 IF counter < n THEN 70
130 REM compute avg and variance
140 avg = sum / n
150 v = SQR(sumsq - n * avg * avg) / (n - 1)
160 PRINT " the average ="; avg
170 PRINT " the variance ="; v
180 END
```

```
Microsoft QuickBASIC
Enter data sizeof N:? 2
Enter data of X= ? 18
Enter data of X= ? 3
 the average = 10.5
 the variance = 10.6066
```

البرنامج لعدد معلوم من الأعداد

التفرع بطريقة العداد فى بيسك

*يقصد به تكرار تنفيذ الأمر لعدد من المرات فى وقت سريع جداً كمثال طباعة جدول ضرب لعدد 6 مرات .

*يعتبر الصبر على تكرار التنفيذ من أهم مميزات الحاسوب إذ أن الإنسان لا يطيق ذلك خصوصاً وإن تجاوز الحد.

*تتميز لغات البرمجة بخاصية التفرع بالعداد وذلك إذا كان العدد المراد تكراره معلوم مسبقاً .

الصيغة العامة للتفرع بالعداد أو التكرار :

* عبارة **For ... Next** : وهى توجيه الحاسوب بتنفيذ التكرار حتى عبارة **Next** وتعنى إبتداءً من كذا نفذ بعدد N من الخطوات والصيغة العامة لها هى :

For (Variable) = Constant 1 To Constant 2 Step N Increment

_____

_____

Next (Variable)

حيث : **For** هى الحلقة وهى للتكرار وتعنى إبتداءً من ، **Variable** هو المتغير (رمز الحلقة) **Constant 1** القيمة الإبتدائية للحلقة ، **Constant 2** القيمة النهائية ، **Step** الخطوات N عدد الخطوات ، **Increment** الزيادة فى طول الخطوة أو عددها ، ـــــ يعنى التنفيذ للمطلوب **Next** الخطوة التالية للحلقة ،،،،، إذا لم تعطى قيمة **Constant 1** وأيضاً **Increment** يعتبرهما الحاسوب = 1 .

مثال : أكتب برنامجاً لطباعة الأعداد الزوجية من 6 إلى 20 ؟

```
10 CLS
20 FOR j = 6 TO 20 STEP 2
30 PRINT j
40 NEXT j
50 END
```

```
6
8
10
12
14
16
18
20
```

وأما تنازلياً فيكون :

```
10 CLS
20 FOR j = 20 TO 6 STEP -2
30 PRINT j
40 NEXT j
50 END
```

```
20
18
16
14
12
10
8
6
```

مثال : أكتب برنامجاً لطباعة الأعداد الفردية من 7 إلى 19 ؟

```
10 CLS
20 FOR j = 7 TO 19 STEP 2
30 PRINT j
40 NEXT j
50 END
```

```
7
9
11
13
15
17
19
```

وأما تنازلياً فيكون :

```
10 CLS
20 FOR j = 19 TO 7 STEP -2
30 PRINT j
40 NEXT j
50 END
```

```
19
17
15
13
11
9
7
```

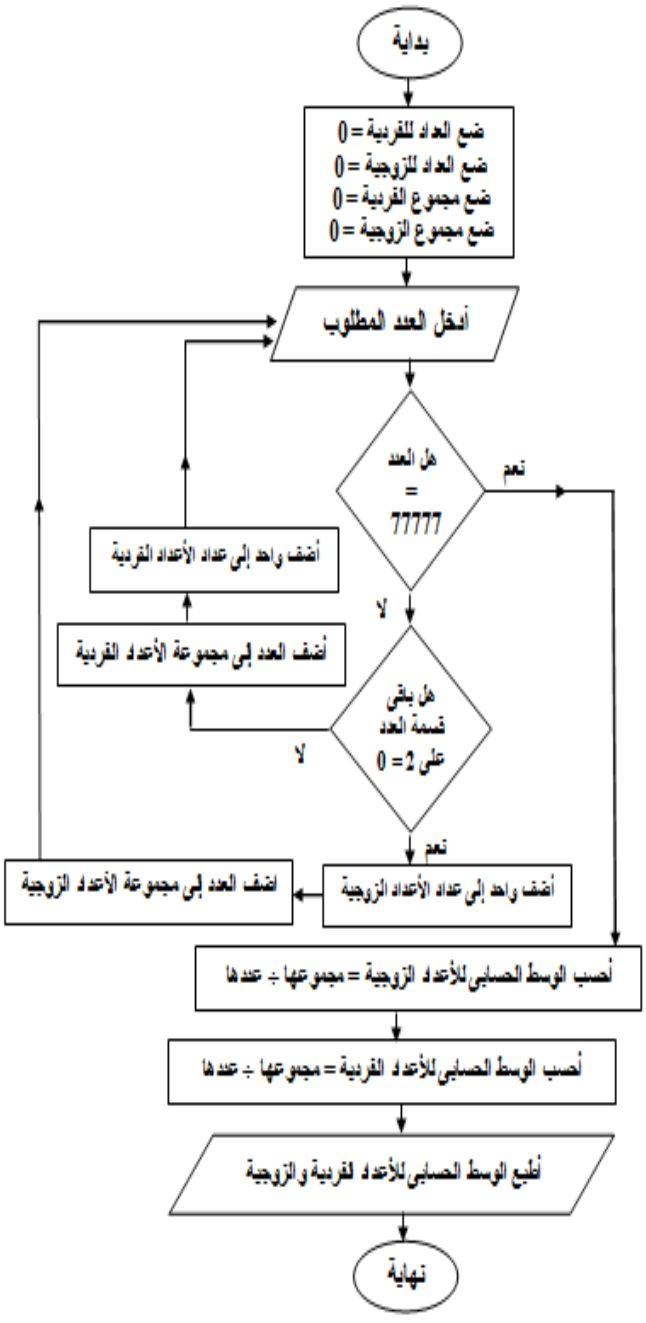
إذا أردنا تنفيذ برنامج الإنحراف المعيارى للأعداد التى عددها N نغير الأوامر من 50 إلى 120 وباقى الأسطر تكون كما هى .

راجع صفحة 18 للرجوع للأسطر الباقية

وبهذه الطريقة نكون قد إختصرنا ثلاث خطوات من خلال عبارة **For ... Next** الأولى هى إعطاء العداد قيمة إبتدائية والثانية إضافة واحد للعداد فى كل مرة والثالثة إختبار العداد مع العدد الكلى N .

```
D:\basmala\program\PROG~1.LAN\Q64C9~1.BAS\Qbasic\QB.EXE
File  Edit  View  Search  Run  De
10 CLS
50 FOR j = 1 TO N
60 INPUT " Enter data of X ="; X
70 sumsq = sum + X
80 sumsq = sumsq + X * X
90 NEXT j
```

مثال : أرسم مخطط لحساب متوسط الأعداد الفردية والزوجية لمجموعة من الأعداد عددها غير معلوم ؟



مثال : أكتب برنامجاً يطبع على الشاشة الأعداد الزوجية من 10 إلى 2 ؟

لاحظ أننا لم نستخدم أرقام الأسطر وتم تنفيذ البرنامج صحيحاً وذلك لأن نسخة كيو بيسك تقبل الترقيم وعدمه أيضاً .

```
Microsoft QuickBASIC
File  Edit  View  Searc
CLS
REM Evening numbers
FOR j = 10 TO 2 STEP -2
PRINT j
NEXT j
END
```

```
10
8
6
4
2
```

مثال : أكتب برنامجاً يطبع على الشاشة الأرقام التى تقبل القسمة على 5 ؟

```
Microsoft QuickBASIC
File  Edit  View  Search
CLS
FOR j = 10 TO 20 STEP 5
PRINT j
NEXT j
END
```

```
10
15
20
```

البريمجات فى لغة بيسك

Subroutines procedures

تعريف : البريمج هو جزء من البرنامج يؤدى دور كامل من الأدوار الفرعية .

تعريف آخر : هو برنامج فرعى يؤدى مهمة محددة داخل البرنامج الرئيسى .

يتم تقسيم البرنامج الرئيسى لعدة برامج فرعية تسمى بالبريمجات أو الروتينات وهى من صفات البرمجة الممتازة وتسمى بالبنائية .

أمثلة لبريمجات :

بريمج حساب فاتورة الكهرباء ، بريمج حساب الوسط الحسابى للأعداد، بريمج طباعة أسماء الموظفين ، غيرها.

من فوائد البريمجات :

(1) تسهيل البرنامج وتدقيق المخرجات (2) من صفات البرمجة الممتازة (3) سرعة إكتشاف الأخطاء وتصحيحها

س/ **عرف الروتينان** Subroutines **أو الإجراءات** ***procedures*** ؟

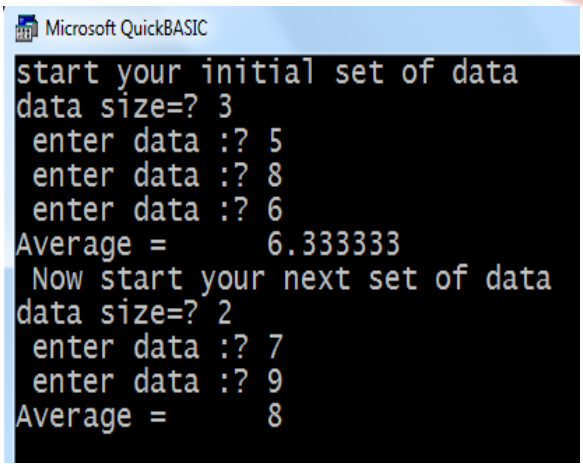
هى جزء من البرنامج الرئيسى تؤدى مهام محددة ،،،،،،، أو يمكنك كتابة أى من التعريفين أعلاه .

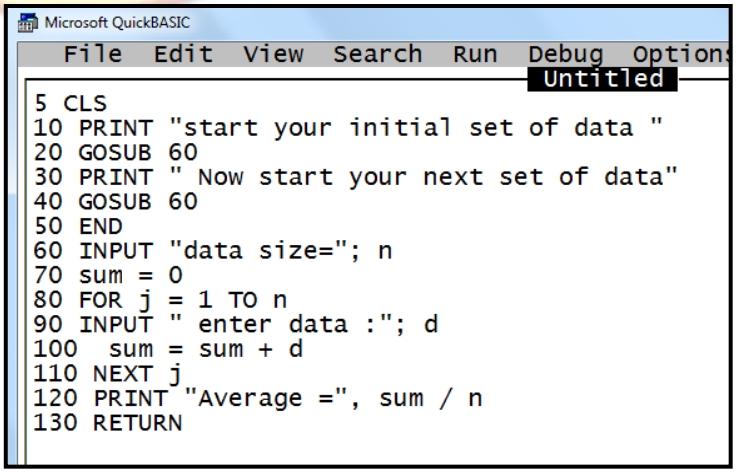
طريقة عمل البريمج أو الروتين :

يبدأ البريمج بالأمر **GoSub** إضافة لرقم السطر أى بهذه الطريقة **(GoSub 60)** ويعنى نفذ البريمج أو الأمر فى السطر رقم 60 وينتهى فى السطر السابق له مباشرة بالأمر **Return** أى أخرج من البريمج ونفذ أوامر البرنامج الرئيسى الآن . ،،،،، فى البريمجات لا نستخدم عبارة **Goto** لأنها غير محبذة فى البرمجة .

س/ **كيف ينفذ الحاسوب الروتين الفرعى** ؟ يبدأ بالتنفيذ إذا وجد الأمر **GoSub** والذى يحدد له السطر المراد القفز إليه مباشرة ويبدأ التنفيذ ويتوقف عندما يجد الأمر **Return** بعدها يعود للبرنامج الرئيسى .

مثال : حساب الوسط الحسابى لمجموعتين من الأعداد عددها N ؟





لاحظ المرة الأولى أدخلنا حجم الـ data size = 3 وأدخلنا 6,8,5 وكان الوسط الحسابى = 6.333

لان الوسط الحسابى = مجموعها وهو 19 ÷ عددها وهو 3 وفى الثانية أدخلنا 9,7 والناتج = 8

الدوال المحجوزة فى لغة بيسك

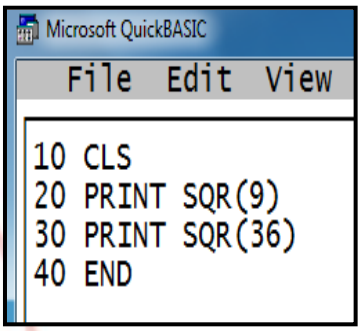
تعريف : هى دوال تستخدم لغرض معين ولا تصلح لغيره وتكتب بطريقة محددة .

تستخدم الدوال المحجوزة فى كثير من البرامج والتطبيقات البرمجية ويتم تخزين هذه الدوال فى لغة البرمجة وتعطى قيم وتستخدم حين الحوجة لها كمثال دالة الجزر التربيعى ودالة اللوغريثم ودالة التربيع وغيرها .

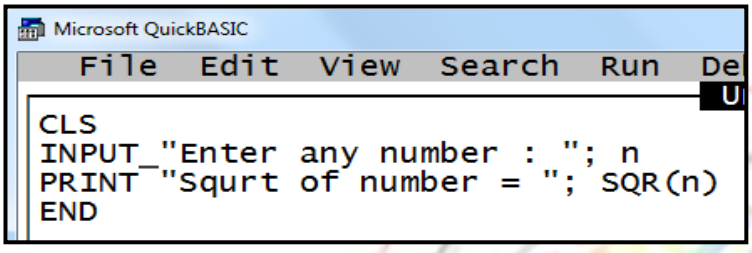
مثال : ما هى دالة الجزر التربيعى فى لغة بيسك للعدد 30 ؟

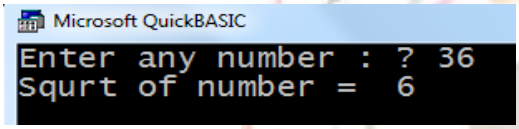
الحل : **Sqr (30)** وتعنى أحسب الجزر التربيعى للعدد 30 ؟

مثال : البرنامج التالى يقوم بحساب الجزر التربيعى للعدد 36 ,9 ؟



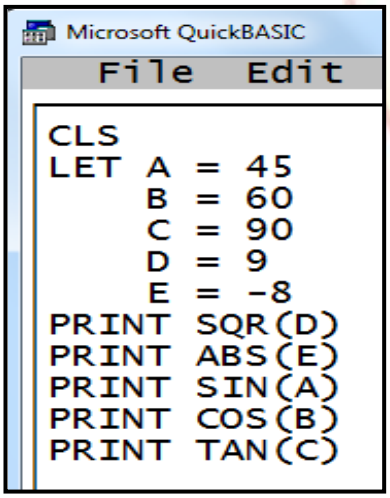
مثال : البرنامج التالى يقوم بإدخال عدد يختاره المستخدم ثم يقوم البرنامج بحساب الجزر التربيعى له ؟





الدوال الرياضية فى لغة بيسك

الطريقة التى كتبت بها الدالة باللغة العربية تسمى الصيغة الرياضية كمثال **جتا (س)** ، **ظا (س)** وهكذا أما فى بيسك تكتب :



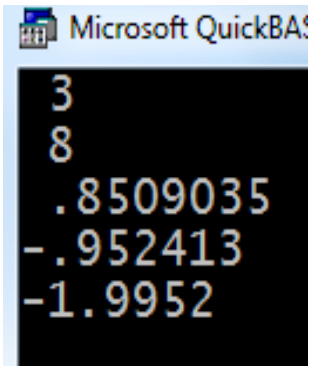
(1) جيب الزاوية **جا (س)** وتكتب فى بيسك هكذا **Sin (x)**

(2) جيب تمام الزاوية **جتا (س)** وتكتب فى بيسك هكذا **Cos (x)**

(3) ظل الزاوية **ظا (س)** **Tan (x)**

(4) لوغريثم العدد الطبيعى **لـــو (س)** **Log (x)**

(5) ظل تمام الزاوية وهى دالة عكس ظل الزاوية **ظتا (س)** $^{-1}$ **ATan (x)**



(6) الجزر التربيعى للعدد **(س)** $\sqrt{س}$ **SQR (x)**

(7) القيمة المطلقة للعدد $|س|$ **Abs (x)**

(8) الدالة الأسية وهى لحساب الأس لعدد معين e^x **Exp (x)**

(9) دالة توليد رقم عشوائى **Ran (x)**

الدوال الحرفية فى لغة بيسك

م	الدالة الحرفية	وظيفتها
1	**CHR$ (m)**	إيجاد الحرف المقابل للرقم m فى جدول ترميز آسكى
2	**ASC (x$)**	إيجاد ترميز آسكى لأول حرف فى سلسلة الحروف x$
3	**LEFT$ (x$, m)**	إستخلاص عدد m من الحروف من الجملة x$ بدءً من اليسار
4	**RIGHT$ (x$, m)**	إستخلاص عدد m من الحروف من الجملة x$ بدءً من اليمين
5	**MID$ (x$, m,n)**	إستخلاص جزء طوله n بدءً من الحرف رقم m من الجملة x$
6	**LEN$ (x$)**	لحساب طول الجملة الحرفية x$

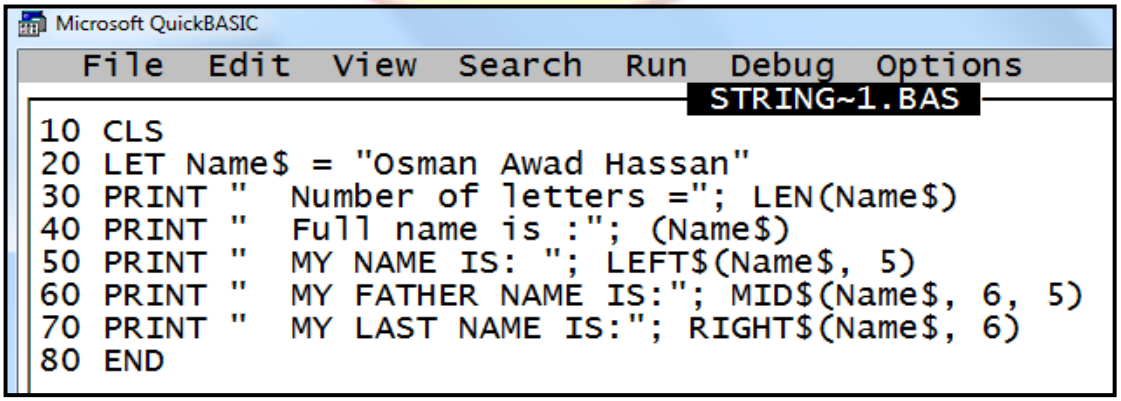
ملحوظة : لا يهم كتابة الدوال بالحروف الكبيرة أو الصغيرة لأن البرنامج تلقائياً يحولها للحروف الكبيرة .

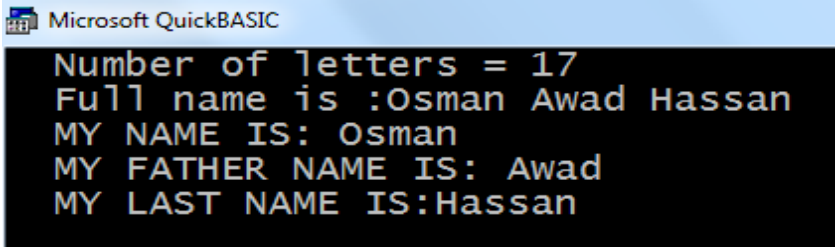
مثال :

البرنامج التالى به دالة لحساب سلسلة عدد الحروف فى إسم بما فيها المسافة بين الكلمات ولإستخلاص إسم معين من الأسماء من اليمين واليسار والوسط .

ملحوظة: البرنامج الموجود بالكتاب به أخطاء برمجية فى كتابة الدوال وتم تعديله وهذا هو البرنامج الصحيح

الشرح : الدالة التى تحسب عدد الحروف هى **Len** وحسب الإسم فإن عدد الحروف هى 15+ 2 = 17 أى أن المسافة أخذت حرفين والباقى لحروف الإسم والسطر رقم **50** لإستخلاص عدد حروف من اليسار **Left** فقط وطوله 5 أى حروف إسم **Osman** أما الدالة **mid** للوسط وهو **Awad** والدالة **Right** من اليمين وهو **Hassan**



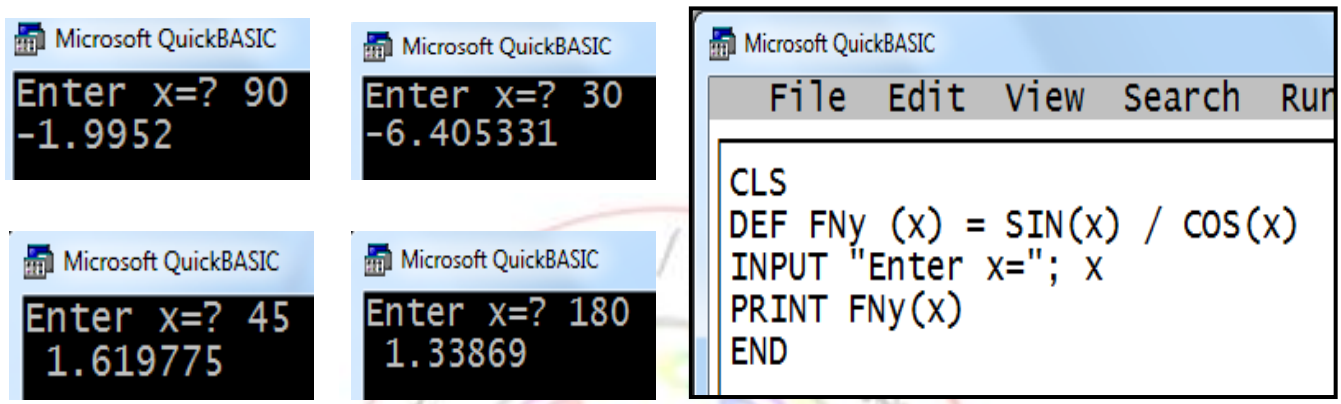


الدوال المبرمجة فى لغة بيسك

تعريف : هى الدوال التى يقوم المبرمج بتصميمها حسب رغبته ويتم تنفيذها كأنها دوال محجوزة .

يمكن فى لغة بيسك تصميم دوال خاصة وذلك عن طريق الأمر DEF FN أو def fn وتعنى التعريف أى definition والدالة هى fn وهو الأمر الذى يتم به تعريف الدالة التى يراد برمجتها وعند إستدعاء الدالة نكتب فقط الأمر fn بحيث يقوم بمناداة الدالة وتنفيذها لاحظ البرنامج التالى .

مثال : تصميم دالة الظل من خلال دوال الجيب وجيب التمام حيث يقوم المستخدم بإدخال الزاوية .



لتاكيد الدالة تم تنفيذ البرنامج أربع مرات وفى كل مرة يتم إدخال زاوية مختلفة ويتم حساب الظل لها

تنبيه : راجع تمرين الكتاب صفحة **65 -66**

(**تم بحمد الله وتوفيقه**) ، ، ،

أسالكم خالص الدعاء لى ولوالداى

اللهم صل وسلم وبارك على حبيبك النبى محمد الأمين عدد خلقك ورضا نفسك وزنة عرشك ومداد كلماتك

(اللهم لا تؤاخذنا إن نسينا أو أخطانا) ،،،، الأستاذ / عثمان صدى **+249918084991**

عبارة أعجبتنى ولكنى لا أتذكر قائلها :

كن شديد التسامح مع من يخالفك فى رأيك فإن لم يكن رأيه كل الصواب فلا تكن أنت كل الخطأ بتشبثك وأقل ما فى إطلاق حرية الفكر والقول تربية الطبع على الشجاعة والصدق وبئس الناس إذا قسروا على الجبن والكذب

أخى الطالب ،،،،

أختى الطالبة ،،،،

قم وإنتفض وأطرد اليأس فأنا بداخلى متيقن أنك لست مهملاً كما يظنون أو كما يصفك البعض ، إلا أنك مُقصر فى بعض واجباتك ، فأنت أقوى مما يظنون وأحسبك رانٍ إلى مستقبل يزهو بالفرح والإزدهار ؛ بيد أن بعض العقبات التى تصنعها لنفسك تجعلك فى محك الظنون لدى الآخرين وهذه العقبات هى التى تجعلك فى بوتقة اللامبالاة ويدفعها إستهزاء الآخرين من حولك بك وبعض تصرفات أولئك الحمقى أمثال (. . و . .) دعك منهم .

قف بينك وبين ذاتك وتأمل نفسك بعيداً عن الشرنقة وخارج حلقة اليأس التى تسجنك من الداخل وإتزن بين عواطفك وفكرك وكن سديد الرأى ولا تجامل نفسك ستجدها كما وصفتك فى بادئ حديثى .

ولأنى أعلم جيداً أنك تستطيع عاهدنى وعاهد أبويك وأشقاؤك ومعلميك ومن ينتظرك من أحباؤك وعاهد الله سبحانه وتعالى ونفسك قبلنا جميعاً .

عاهدنا أن ترمي بكل هذه المتلازمات المحبطة خلف ظهرك وتنطلق فى خط التفوق والتوجه نحو هدفك ولك حبٌ خالص يدعو لك من على البعد ولسان لا يفتأ يذكُرُك فى كل عشية وصباح ولك التوفيق والسداد .

وأنا أنتظر أى إقتراح منك أو لمن لا يجد فى كلامي مثقال خردل من صواب.

محبكم / عثمان عوض حسن

وأنا عن نفسي مؤمن بهذه العبارات وأجعلها زادي دوماً :

طموحاتى الكبرى بعيدة جداً مثل الشمس قد لا أستطيع الوصول إليها ولكن يمكننى النظر لأعلى لرؤية جمالها ، والإيمان بها ، ومحاولة السعى وراءها حيث تقودنى ،،،،،،،،،،،،،،،(الكوت)

اللهم علمنا ما ينفعنا وأنفعنا بما علمتنا يا رب

اللهم صل على سيدنا محمد وعلى آله وصحبه وسلم

هدفى هو : التعليم حق للجميع :

بمعنى يمكنك الحصول على نسخة إلكترونية لأى من هذه المذكرات مجاناً دون أى مقابل مادى .

لطلب النسخة الأولى والتى تشمل باقى الأبواب إتصل أو أرسل رسالة واتساب فقط للرقم #٠٠٢٤٩٩١٨٠٨٤٩٩١#

ستصلك مجاناً بدون أى مقابل مادى

وكذلك جميع مذكرات الحاسوب للثانوى (الصف الأول والثالث)

ساهم معنا فى نشرها ليستفيد منها من لا يستطيع شراء الكتب التى أصبحت ترهق العاتق ولكى تساهم بصدقة جارية لك ولوالديك

اللهم أجعل فى عملنا هذا الإخلاص وجنبنا الرياء وأعمال الشيطان